

تشییه سازی

مؤلف

مهندس امین اله مه آبادی

انتشارات آذرخش

بخش اول

فصل اول - مقدمه



۵۲۵	۱۹.۷ - پروسیجر تخصیص منبع ترشه کارر
۵۲۵	(سایروتن های SEIZE و ALLOC)
۵۲۹	۱۹.۸ - تغییر وضعیت یک GATE
۵۲۹	۱۹.۸.۱ - بازگردان یک GATE (سایروتن OPEN)
۵۳۰	۱۹.۸.۲ - بستن یک GATE (سایروتن CLOSX)
۵۳۰	۱۹.۹ - انتخاب صف نگارش کارر (تابع NOS)
۵۳۱	۱۹.۱۰ - انتخاب سرویس دهنده نگارش کارر (تابع NSS)
۵۳۳	۱۹.۱۱ - توقف یک فعالیت (سایروتن STOPA)
۵۳۳	نمونهات
۵۳۳	بخش ۱
۵۳۴	بخش ۲
۵۳۸	بخش ۳
۵۵۰	بخش ۴
۵۵۹	بخش ۵
۵۶۱	دستورات زبان SLAM II
۵۷۵	جداول آماری
۵۷۶	منابع

بخش اول

فصل اول - مقدمه



فصل اول

مقدمه

شبیه سازی

شبیه سازی^(۱) یعنی "ساختن شبیه یک سیستم به هر روش یا صورت ممکن" که از بعضی جهات با سیستم مرجع می تواند متفاوت باشد. این تفاوتها از نظر ابعاد، حدود و ثغور و مانند اینها آغاز می گردد ولی با توجه به اینکه شبیه سازی رفتار یک سیستم را مدل می کند و مورد مطالعه قرار می دهد، از نظر رفتار نباید با سیستم موجود تفاوت داشته باشد. پس می توان گفت "شبیه سازی ساختن شبیه یک سیستم با تمامی رفتارش" است.

بدیهی است که تعاریف ارائه شده بیان کامل و آژه شبیه سازی نمی باشد و نیاز به مقدّماتی است که پس از ذکر آن، ماهیت هدف و روش های شبیه سازی شرح داده می شود.

هدف شبیه سازی "مطالعه و بررسی سیستم مرجع"^(۲) می باشد یعنی "مطالعه رفتار سیستم مرجع مورد بررسی"، مدنظر قرار می گیرد. برای مطالعه سیستم روشهای مختلفی وجود دارد که از جهت چگونگی مطالعه و شیوه عمل تقسیم بندی می شوند و فرض آن بر احتمال دسترسی به سیستم یا احتمال وجود موضوع مورد مطالعه است.

۱-۱- شیوه های مطالعه سیستم مرجع

برای مطالعه یک سیستم روشها و متدهای گوناگون مطرح است ولی مسئله مهم در مطالعه وجود خارجی موضوع یا پدیده مورد نظر می باشد. اگر برای مطالعه توان آن را یافت "سیستم مرجع حیات ندارد" لذا چگونه می توان سیستم را محور بررسی قرار داد؟ از حیث وجود یا عدم وجود مرجع مورد نظر، مطالعه و بررسی سیستم ها به دو دسته تقسیم می شود:

- ۱- مطالعه مستقیم^(۳)
- ۲- مطالعه غیرمستقیم^(۴)

1- Simulation

2-reference system

3-direct

4- indirect

۱.۱.۲ - مطالعه غیر مستقیم

در بسیاری از موارد نیاز به مطالعه غیر مستقیم رفتار یک سیستم داریم و آن زمانی است که بنا بر دلایل مطرحه مطالعه مستقیم، توان مستقیماً سیستم را مطالعه نمود. با روش غیر مستقیم بسیاری از معایب روش مستقیم قابل رفع می‌باشد و حتی می‌توان رفتار یک سیستم را قبل از خلق یا ایجاد ضمن رفع معایب، نتایج حاصل را بررسی و آن را تولید نمود. مثلاً برای طراحی یک CPU (پردازنده مرکزی) کامپیوتر با مدل سازی مطالعه و پس از اخذ نتایج حاصل، ضمن رفع عيوب طراحی، مطلق و فیزیکی، ساخت آن صورت می‌پذیرد.

لذا می‌توان گفت که "شییه سازی مطالعه غیر مستقیم سیستم مرجع" است و به دور از مشکلات به بررسی موضوع می‌پردازد. شییه سازی دارای مزایای بسیاری است که کاربرد آن را توسعه داده و بطور مختصر به شرح آن می‌پردازیم:

- امکان مطالعه یک سیستم مرجع قبل از ایجاد یا خلق یا پس از انهدام آن را فراهم می‌کند.
- پارامترهای زمانی سیستم مورد مطالعه قابل تغییر است.
- از نظر اقتصادی و زمانی بسیار مقرون به صرفه است.
- محدودیتها و خطرات ناشی از مطالعه مستقیم قابل رفع می‌باشد.
- مطالعه جزء به جزء سیستم و کنار هم نهادن اجزاء را امکان پذیر می‌سازد.
- حداکثر کارایی با این روش قابل دستیابی است.

۱.۲ - شییه سازی سیستم‌ها

شییه سازی را می‌توان با یک تعریف کاملتر بیان و آن را در قالب فرایندی ارائه کرد: "شییه سازی فرایندی از مدل سازی و اجرای مدل است".

در واقع رفتار سیستم مرجع را با استفاده از مدل مناسب، مدل سازی می‌کنیم و چنانچه مدل دارای قدرت دریافت داده‌ها و اجرای آن باشد یک مدل اجرایی است که با ارائه نتایج دو پر و سه مدل سازی و اجرای آن را امکان پذیر می‌سازد. اگر بتوان مدل را به تنهایی اجرا نمود، آن را به یک برنامه شییه سازی تبدیل و برنامه تولیدی مدل سیستم مرجع بر روی کامپیوتر اجرا می‌شود.

برنامه‌های شییه سازی، امر شییه سازی و برنامه نویسی، مدل سیستم را راحت تر انجام می‌دهند و کاربران با امکانات موجود آنها نیازهای خویش را برآورده می‌سازند. لذا لزوم وجود زبان‌های شییه سازی کامپیوتری احساس می‌گردد.

۱.۱.۱ - مطالعه مستقیم

مطالعه مستقیم یعنی بررسی و مطالعه سیستم مرجع مستقیم و از نزدیک صورت می‌گیرد. پس بهترین فرض مطالعه مستقیم "وجود سیستم مرجع" می‌باشد. این امر در کنترل کیفیت یک محصول تولیدی بسیار حیاتی است و فرض وجود نمونه محصول مورد آزمایش اساس آن را تشکیل می‌دهد. با وجود دقت در مطالعه مستقیم این روش همیشه جوابگویی ما نیست و دارای عیوبی به شرح زیر است:

- ممکن است "سیستم مرجع" در لحظه مطالعه حیات نداشته باشد یعنی وجود خارجی ندارد که حتی بتوان آنرا مستقیماً بررسی نمود. مثلاً "انتخابار هواپیما" را نمی‌توان از طریق مطالعه مستقیم، موضوع بررسی قرار داد زیرا سیستم مرجع مورد مطالعه (هواپیما) و تعارض جهت مطالعه مستقیم در دسترس نیست و از میان رفته است.

- چنانچه پارامتر زمانی پیش آمده‌ها یا حدوث پدیده‌ها بسیار طولانی باشد و توان در زمانهای کوتاه به آنها دسترسی داشت. مثلاً "تکرار وقوع زلزله" دیرارهای موارد حتی ۱۰ الی ۱۵ سال می‌کشد. برای مطالعه مستقیم این سیستم و مسائل آن انتظار مدت طولانی اجتناب ناپذیر می‌باشد و برهه‌های طولانی را طلب می‌کند.

- اگر پارامتر زمانی پیش آمده‌ها یا حدوث آنها بسیار کوتاه باشد و حتی توان از نظر مطالعه، حدوث آنرا مشاهده و برآورد نمود لذا کوتاهی زمان رخدادها بسیاری از نتایج مطالعه را از دید پنهان می‌کند. مثلاً در "مطالعات امنی" زمان رخدادها و واکنشهای درون اتم در برهه‌های بسیار کوتاه زمانی و بسیار گذار رخ می‌دهد.

- حتی ممکن است که مطالعه مستقیم یک سیستم از نظر اقتصادی و زمانی مقرون به صرفه نباشد و لذا دیرارهای از موارد نیز به علت خطرات موجود، مطالعه را محال می‌نمایاند. مانند بررسی "شلیک یک موشک به یک هواپیمای در حال پرواز" که مطالعه مستقیم آن از نظر اقتصادی دارای هزینه و بسیار خطرناک نیز می‌باشد.

- پیچیدگی، سیستم مرجع و محدودیت‌های آن، مطالعه و بررسی بخشی از سیستم را به روش مستقیم غیر ممکن می‌سازد. زیرا مطالعه آن جزء خاص در کل مجموعه به علت دریافت پارامترهای ورودی از اجزاء دیگر، نتایج مطلوب را به دست نمی‌دهد. مانند مطالعه بخشی از مدار یک سیستم کنترل که سگنالهای ورودی از بخشهای دیگر دریافت می‌کند و با اثرگذاری آنها نتایج صحیح از بررسی حاصل نمی‌گردد.

ایجاد یک شهر را می‌توان بیان کرد. البته نرم‌افزارهای شبیه‌سازی بسیاری در امور طراحی معماری، طراحی مکانیک، مطالعات فیزیکی و مانند آن به بازار ارائه گردیده و اوج آن را می‌توان در حقیقت مجازی^(۱) یافت.

معایب شبیه‌سازی

اغلب تحلیل‌گران در روبرو شدن با یک مساله تمایل به یافتن سریع پاسخ دارند. آیا می‌توان به بهترین صورت و با کمترین هزینه پاسخ مسائل مزبور را از طریق شبیه‌سازی یافت؟ در بسیاری از موارد پاسخ به چنین سؤالی بنابر دلایل زیر منفی است:

- طراحی یک مدل شبیه‌سازی مطلوب اغلب وقت‌گیر و پرهزینه می‌باشد و نیاز به تخصص‌های بالایی دارد. در پاره‌ای موارد این تلاش حتی تا چندین سال عادی تلفی می‌شود.
- شبیه‌سازی وضعیت یک مسئله را به دقت منعکس می‌کند. این مطلب در صورتی صحیح است که بسیاری از ظرایف در طراحی و اجرای مدل شبیه‌سازی دخالت داده شوند. لذا در غیر اینصورت نتایج نادرستی حاصل می‌گردد.
- شبیه‌سازی ممکن است در پاره‌ای موارد به عنوان یک روش دقیق شهرده‌نشود و میزان بی‌دقتی آن قابل اندازه‌گیری نباشد. البته امکان تحلیل دقت کار تا حدی از این امر می‌کاهد.
- از آنجا که نتایج شبیه‌سازی رقمی است، دقت اعشاری ارقام وابستگی‌ها و محدودیت‌های بسیاری دارد. به‌دادن زیاد به نتایج عددی حاصل صحیح نیست چون (انتشار خطا) در پاره‌ای موارد خروجی‌ها را بی‌اعتبار می‌سازد.
- با ارائه نرم‌افزارهای شبیه‌سازی و سیستم‌های کامپیوتری معایب انتشار خطا و هزینه‌های طراحی و اجراء تا حدود بسیاری کاهش یافته است.

مراحل اساسی یک مطالعه شبیه‌سازی

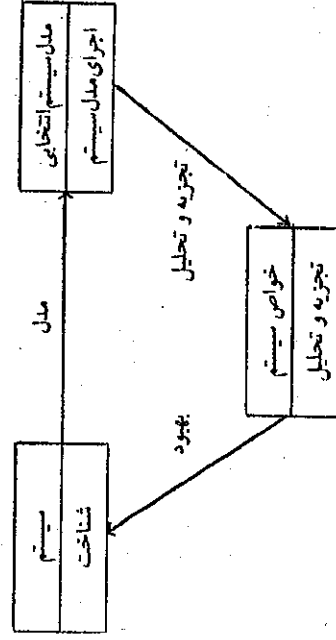
شکل ۱-۲ مراحل را نشان می‌دهد که اجرای آنها راهنمای مناسبی برای طراحی مدل در مطالعه صحیح شبیه‌سازی است.

معمولاً هر زبان شبیه‌سازی (که شرح داده خواهد شد) دارای مدل خاص خود می‌باشد. چنانچه زبان‌های فوق به ابزاری با قدرت ارائه گرافیکی، مدل‌های سیستم مرجع مجهز باشند، پدیده شبیه‌سازی بسیار آسان انجام می‌شود. زیرا مثلاً در زبان شبیه‌سازی SLAM و جهت سهولت برنامه‌سازی، ترسیم مدل SLAM از مسئله باید صورت گیرد و پس از آن به برنامه به زبان SLAM تبدیل و با کامپایلر مربوطه برنامه اجرا می‌شود. اکثر زبانهای شبیه‌سازی مدل گرافیکی و عناصر مربوطه را ارائه می‌کنند.

شکل ۱-۱ نشان دهنده ساختار شبیه‌سازی است که فرایند مدل سازی، اجرای مدل و در نهایت آنالیز نتایج ارائه می‌کند. سیستم مورد مطالعه طبق یک مدل مناسب انتخابی، مدل سازی خواهد شد و چنانچه مدل نتواند اجراء شود، به زبان شبیه‌سازی تبدیل و بر روی کامپیوتر اجراء می‌گردد. لذا باید بتوان به مدل سیستم مورد مطالعه داده‌ها را اعمال نمود و سپس نتایج را بررسی کرد. این امر در صورتی ممکن است که مدل انتخابی قدرت دریافت داده‌ها، چرخش عملیات و ارائه خروجی‌های لازم را داشته باشد.

با این نگرش دو گونه مدل داریم:

- ۱- مدل اجرایی. مدل با قدرت دریافت داده، اجراء و ارائه نتایج که نیاز به برنامه کامپیوتری ندارد.
- ۲- مدل غیر اجرایی. مدل بدون قدرت دریافت داده، چرخش عملیات و ارائه نتایج و برای اجراء به برنامه کامپیوتری تبدیل می‌شود.



شکل ۱-۱ روند شبیه‌سازی

اخیراً در شبیه‌سازی سیستم‌ها، پشرفت‌های بسیاری از جهت کاربرد و بکارگیری آن در علوم مختلف صورت پذیرفته و همچنین ابزارهای بسیاری ارائه شده است. به عنوان نمونه شبیه‌سازی

مراحل اساسی یک مطالعه شبیه‌سازی از دید شکل ۱-۲ به شرح زیر است:

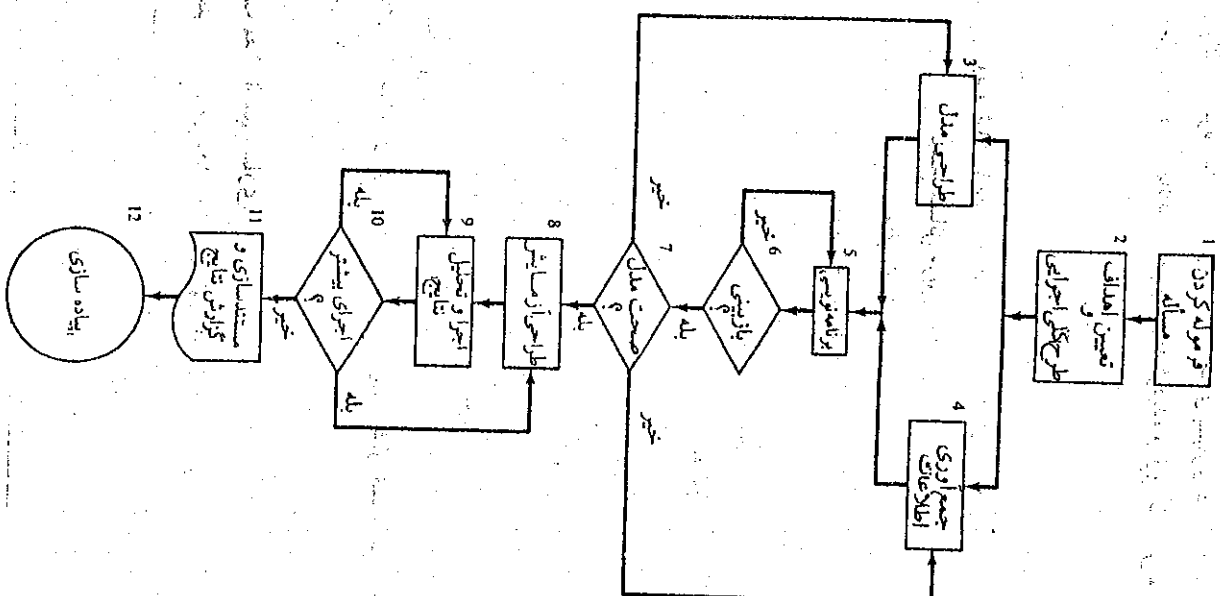
۱- فرموله کردن مسأله. هر مطالعه شبیه‌سازی با تعریف مسأله شروع می‌شود. اگر مسأله توسط مسئول آن، تعیین کنندگان خط مشی و سیاست‌گذاران ارائه شده تحلیل‌گر باید از درک صحیح آن اطمینان حاصل کند. در صورتی که تحلیل‌گر مسأله را تعریف کند درک صحیح آن توسط سیاست‌گذاران و توافق با نحوه تعریف توسط آنها از اهمیت فزاینده‌ای برخوردار است. در مواردی، ارائه تعریف متفاوتی از مسأله ضرورت پیدا می‌کند که البته شکل ۲-۱ چنین امکانی را نداده است. در واقع سیاست‌گذاران و تحلیل‌گران در بسیاری از موارد به وجود یک مسأله واقفند بدون اینکه از طبیعت آن آگاهی داشته باشند. در چنین مواردی، با پیشرفت مطالعه تعریف مسأله ممکن است تغییر کند.

۲- تعیین اهداف و طرح کلی اجرایی. اهداف تعیین‌کننده پرسش‌هایی هستند که باید با استفاده از شبیه‌سازی برای آنها پاسخ ارائه داد. در این مرحله باید تعیین کرد که آیا با توجه به تعریف مسأله و اهداف در نظر گرفته شده برای آن، شبیه‌سازی روش مناسبی برای تحلیل مسأله شمرده می‌شود یا خیر. طرح کلی اجرایی باید دربردارنده سیستم‌های مختلف تحت بررسی و روشی در زمینه ارزیابی میزان کارایی هر یک از آنها باشد. طرح کلی اجرایی همچنین باید متضمن انجام یک بررسی در زمینه تعداد افراد مورد نیاز، هزینه و مدت مربوط به برگزاری هر قسمت از کار باشد و پیش‌بینی‌های لازم در زمینه نتایج قابل حصول در انتهای هر قسمت را انجام دهد.

۳- طراحی مدل. ساخت مدل برای یک سیستم همانقدر کاری علمی شمرده می‌شود که کاری هنری است. آنچه به هنر طراح مدل کمک می‌کند عبارات است از استمداد مجرد کردن خصوصیات اصلی سیستم، انتخاب و اصلاح کردن فرضهای شکل دهنده سیستم و بالاخره کامل کردن مدل به نحوی که نتایج تقریبی مفیدی از آن به دست آید. بدین ترتیب، بهترین شیوه کار آغاز با مدلی ساده و کامل کردن تدریجی آن است. به هر حال، درجه پیچیدگی مدل نباید از حد تأمین مقاصد مطالعه شبیه‌سازی تجاوز کند. نقض چنین اصولی تنها منجر به افزایش هزینه‌های طراحی و اجرای مدل (به وسیله کامپیوتر) می‌شود. همسانی جزء به جزء بین سیستم واقعی و مدل ضروری نیست بلکه باید چکیده سیستم واقعی در مدل وارد شود.

توصیه می‌شود که استفاده‌کننده نهایی از مدل در مرحله طراحی و ساخت آن شرکت داده شود. دخالت دادن استفاده‌کننده، کیفیت مدل ساخته شده را بهبود بخشیده و اطمینان خاطر بیشتری به استفاده‌کننده از نظر ارزش کاربردی آن می‌دهد. گرچه چند الگوریتم شبیه‌سازی عرضه می‌شود ولی باید توجه داشت که هنر طراحی الگوریتمها از طریق تجزیه سیستمهای واقعی می‌توان آموخت.

۴- جمع‌آوری اطلاعات. ارتباط مداومی بین دو مرحله طراحی مدل و جمع‌آوری اطلاعات



شکل ۱-۲ مراحل یک مطالعه شبیه‌سازی

ورودی وجود دارد. برحسب اینکه درجه پیچیدگی مدل چه باشد، عناصر اطلاعاتی مورد نیاز نیز تغییر می کنند. به علاوه چون بخش قابل توجهی از زمان مورد نیاز برای اجرای یک مطالعه شبیه سازی به جمع آوری اطلاعات اختصاص می یابد، شروع زود هنگام این مرحله ضروری است. مرحله جمع آوری اطلاعات معمولاً همزمان با اجرای قسمتهای نخست مرحله طراحی مدل آغاز می گردد.

این امر که اطلاعاتی از چه قبیل باید گردآوری شود، تا حد زیادی توسط اهداف مطالعه شبیه سازی تعیین می شود. به عنوان مثال، در بررسی عملیات جاری یک بانک اگر قصد داشته باشیم براساس تغییر تعداد افراد سرورس دهنده طول صف انتظار را مورد مطالعه قرار دهیم اطلاعاتی از قبیل توزیع های آماری مربوط به زمانهای بین ورود متقاضیان در دوره های زمانی مختلف روز، زمانهای سرورس دهی برای هر سرورس دهنده و توزیع های آماری مربوط به طول صف تحت شرایط مختلف از عملکرد گذشته سیستم مورد نیاز خواهد بود. از اطلاعات مربوط به عملکرد سیستم در گذشته به منظور تعیین اعتبار مدل استفاده می شود.

۵- برنامه نویسی: نظر به اینکه تحلیل اکثر سیستم های واقعی ناظر به ساخت مدلی با نیاز فراوان به امکانات قابل توجه از نظر ذخیره سازی اطلاعات و پردازش آنهاست، ترجمه مدل به یک زبان برنامه نویسی و اجرای آن به وسیله کامپیوتر اجتناب ناپذیر است. طراح مدل باید تصمیم بگیرد که آیا به منظور برنامه نویسی باید از یک زبان عمومی مانند FORTRAN استفاده کرد یا زبان های شبیه سازی مانند SLAM، SIMSCRIPT یا GPSS مناسبترند. نوشتن برنامه به یک زبان عمومی نیازمند صرف وقت بیشتری است ولی یک چنین برنامه ای در مقایسه با برنامه ای که به یکی از زبان های شبیه سازی نوشته می شود به زمان اجرای کوتاهتری نیاز دارد. به طور کلی، چون زبانهای شبیه سازی به نحو قابل توجهی عمل برنامه نویسی را ساده می کنند در میان طراحان مدل تمایل روز افزونی در استفاده از آنها دیده می شود.

۶- صحت مدل: این مرحله ناظر به غلط گیری برنامه کامپیوتری است. در واقع سؤال اساسی در این مرحله آن است که آیا برنامه کامپیوتری به نحو مناسبی کار می کند؟ ایجاد برنامه مربوطه به یک مدل پیچیده بدون صرف وقت قابل توجهی در زمینه تعیین صحت آن امری بسیار مشکل است. در صورتی که پارامترهای ورودی و ساختار منطقی مدل به طرز صحیحی داخل برنامه در نظر گرفته شوند می توان گفت که مرحله تعیین صحت مدل کامل شده است.

۷- اعتبار مدل: در این مرحله باید تعیین کرد که آیا مدل معروف دقیق سیستم واقعی است یا نه. این مرحله معمولاً در قالب یک فرآیند تکراری صورت می گیرد که ناظر به مقایسه مدل با عملکرد سیستم واقعی و اصلاح مدل براساس اطلاعات و نتایج حال از مقایسه میزور است. این فرآیند آنقدر ادامه پیدا

می کند تا دقت مدل قابل قبول تشخیص داده شود.

۸- طراحی آزمایش: در این مرحله ابتدائیه حالت های مختلف را که باید شبیه سازی شوند تعیین می کنند. در اغلب موارد تصمیم گیری در این مورد براساس تجزیه و تحلیل نتایج حاصل از اجرای مقدماتی مدل انجام می گیرد. در مورد هر طراحی از سیستم که تصمیم به شبیه سازی گرفته شود باید طول زمان راه اندازی مدل، طول زمان اجرای شبیه سازی و تعداد بازسازی های مربوط به آن معین شود.

۹- اجرای مقدماتی و تجزیه و تحلیل نتایج: از اجرای مقدماتی مدل و متعاقباً از تجزیه و تحلیل نتایج به دست آمده از آنها به منظور ارائه تخمینهای برای معیارهای عملکرد هر یک از طراحی های مختلف از سیستم استفاده می شود.

۱۰- تعیین تعداد اجرای مدل: براساس نتایج بدست آمده از اجرای مقدماتی تحلیل گر باید تصمیم بگیرد که آیا به اجرای بیشتری نیاز است و در صورت نیاز، طرح آزمایش آنها چگونه باید باشد.

۱۱- مستندسازی و گزارش نتایج: ضرورت تدوین برنامه از طریق ارائه دلایل متعدد قابل توجهی است. اگر قرار باشد که برنامه مجدداً توسط تحلیل گر یا افراد دیگر مورد استفاده قرار بگیرد، ممکن است چگونگی عملکرد برنامه ضروری باشد. شناخت چگونگی عملکرد برنامه، میزان اطمینان به برنامه را تا حدی بالا می برد که سیاست گزاران و کاربران مدل براساس تجزیه و تحلیل نتایج حاصل به تصمیم گیری بپردازند. به علاوه در صورت وجود توضیحات کافی در برنامه، جهت اصلاح آن به سادگی می توان تغییرات لازم را در برنامه داد. دلیل دیگر در توجه ضرورت مستندات برنامه این است که استفاده کنندگان مدل بتوانند به طور اختیاری پارامترهای آن را تغییر دهند و به تعیین روابط حاکم بر پارامترهای ورودی و معیارهای عملکرد (خروجی) پرداخته و یا تعیین کنند که کدام پارامتر می تواند معیار عملکرد خاصی را بهینه نماید.

نتایج حاصل از تجزیه و تحلیل فوق باید به طور واضح و دقیق گزارش شود. این امر استفاده کنندگان مدل را قادر می سازد که تعریف نهائی مسأله را مرور کرده، به بررسی سیستم های آزمایش شده، معیارهای مقایسه آنها، نتایج حاصل از آزمایشها و راه حل ارائه شده برای انجام آنها بپردازند. مضافاً اگر قرار باشد تصمیمات فوق برای مقاماتی در سطح بالا توجه شود، گزارش مورد نظر می تواند مفید باشد زیرا توجه کننده مدل و فرآیند ساخت آن است.

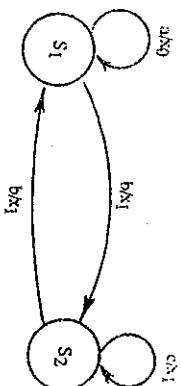
۱۲- پیاده سازی: موفقیت این مرحله بستگی به کیفیت اجرای یازده مرحله قبل دارد. به علاوه، موفقیت مرحله پیاده سازی منوط به این امر است که تحلیل گر تا چه میزان موفق به دخالت دادن استفاده کننده نهائی در سراسر فرآیند شبیه سازی شده باشد. در صورتی که استفاده کننده به طور

اشیاء دارای اثر متقابل که با یکدیگر در حال محاوره می باشند و هدف خاصی را دنبال می کنند. با این تعریف می توان گفت که ابتدا اجزاء سیستم یعنی اشیاء^(۱) با موضوعات^(۲) را می یابیم و رابط بین آنها یعنی پیامها را تعیین می کنیم. (روشهای آنالیز سیستم؛ شیوه و سند خاص خود در شناخت و تحلیل را دیده می کنند)

از دید شی گرا می توان سیستم را به صورت زیر در نظر گرفت:

- ۱- اشیاء objects
- ۲- پیش آمدها (وقایع) events
- ۳- وضعیتها states

از این نظر عناصر اساسی سیستم را اشیاء تشکیل می دهند و با استفاده از وقایعی که رخ می دهد وضع و حالت سیستم تغییر می کند. به عنوان مثال، شکل ۱-۳ مدل PTSM^(۳) یک سیستم می باشد.



شکل ۱-۳ یک دیگرام حالت

هر کدام از دایره ها، O، یک وضعیت را نشان می دهد. با هر پیش آمد (ورود یک ورودی) حالت سیستم تغییر می کند. در این مدل وضعیت شروع S1 می باشد که با ورودی a (پیش آمد) حالت سیستم تغییر نمی کند ولی چنانچه ورودی b باشد، سیستم از وضعیت S1 به وضعیت S2 تغییر می یابد.

با توجه به رخداد پیش آمدها، سیستم ها دستبندی می گردند و مهم است که پیش آمدها چگونه رخ می دهند. آیا وقایع متوالی رخ می دهند یا به صورت همزمان اتفاق می افتند؟ از دید رخ داد وقایع سیستم ها به دو دسته تقسیم می شوند:

- ۱- سیستم های همزمان^(۴)
- ۲- سیستم های غیرهمزمان (ترتیبی)^(۵)

1- objects	2- subjects	3- Finite State Machine
4- Concurrent System	5- sequential system	

فعال در فرآیند ساخت مدل شرکت کرده و در نتیجه، از طبیعت مدل و خروجی های آن درک صحیحی داشته باشد احتمال کاربرد سریع مدل افزایش می یابد. بالکمک، اگر خصوصیات مدل و فرضیات شکل دهنده آن در پرده ابهام باقی بمانند، امر کاربرد آنکو می تواند با دشواری روبرو شود.

فرآیند ساخت مدل شبیه سازی (شکل ۱-۲) را می توان به چهار قسمت تقسیم کرد. قسمت اول مشتمل بر مراحل ۱ و ۲ (تعریف مسئله، تعیین اهداف و طرح کلی) بوده و ناظر به کسب شناخت از مسئله است. تعریف اولیه از مسئله معمولاً گنگ و مبهم است. اهداف اولیه در مراحل بعد دچار تغییر می شوند و اغلب چنین است که باید که طرح کلی تجدید نظرهایی صورت می گیرد.

قسمت دوم مربوط به طراحی مدل و جمع آوری اطلاعات است و مراحل ۳ تا ۷ (طراحی مدل، جمع آوری اطلاعات، برنامه نویسی، تصحیح برنامه و تعیین اعتبار مدل) را در بر می گیرد. ارتباط مداوم مراحل این قسمت با یکدیگر را نباید نادیده گرفت. دخالت ندادن استفاده کننده از مدل در این قسمت می تواند عواقب نامطلوبی در مرحله کاربرد آن در پی داشته باشد.

قسمت سوم مربوط به اجرای مدل است و شامل مراحل ۸ تا ۱۰ (طراحی آزمایش، اجرای مقدماتی و تحلیل نتایج و اجراهای اضافی) می شود. آنچه در این قسمت مهم است، در دست داشتن طرح حساب شده ای به منظور برگردان کردن آزمایش از طریق پیاده سازی مدل شبیه سازی است. در واقع شبیه سازی گسسته یک مسئله تصادفی جز یک تجربه آماری نیست. به عبارت دیگر، متغیرهای خروجی از تخمین طائی تشکیل می شوند که از خطای تصادفی جدا نیستند و در نتیجه انجام یک تجربه و تحلیل مناسب آماری ضروری است. در برابر این شیوه صحیح می توان از طرز کار غلط تحلیل گرانی یاد کرد که صرفاً بر اساس یک اجرا و حاصل به دست آمده از آن به نتیجه گیری می پردازند.

چهارمین و آخرین قسمت پیاده سازی است و از دو مرحله ۱۱ و ۱۲ (تدوین برنامه و گزارش نتایج و پیاده سازی) تشکیل می شود. موفقیت در زمینه کاربرد مدل شبیه سازی بستگی به شرکت دادن استفاده کننده نهائی در ساخت مدل و کیفیت اجرای مراحل دیگر فرآیند مدل سازی دارد. شاید مهمترین مرحله در فرآیند ساخت مدل مرحله ۷ (تعیین اعتبار مدل) باشد زیرا یک مدل غیر معتبر نتایج نادرستی تولید می کند که کاربرد آنها می تواند خطرناک و یا پرهزینه باشد.

۱-۳- سیستم

سیستم را می توان با استفاده از آنالیز شی گرا^(۱) به صورت زیر تعریف نمود: مجموعه ای از

چنانچه پیش آمدهای سیستمی به صورت همزمان^(۱) رخ دهند این سیستم همزمان می‌نامند. زیرا پیش آمدها، امکان اتفاق و انجام همزمان را پتانیه‌اند. سیستمی که پیش آمدهای آن به صورت ترتیبی (پی در پی) یا غیرهمزمان رخ دهد، سیستم غیرهمزمان می‌نامند.

در انتخاب مدل مناسب برای مدل سازی، شناخت سیستم و انطباق مدل با سیستم بسیار مهم است. چنانچه از فلوچارت‌های (متداول) مانند شکل ۱-۲ برای مدل سازی یک سیستم همزمان یا استفاده گردد با این مدل، مدلسازی سرازای امکان پذیر نیست. زیرا امکان مدلسازی وقایع یا پروسه‌های همزمان را ندارد. پس باید سیستم و وقایع آن را شناخت و قدرت مدل را برآورد نمود.

رخ داد پیش آمدها اساس حیات یک سیستم هستند و چنانچه رخ داده‌ها بتوانند همزمان پیش آیند، مدلسازی و انتخاب مدل مناسب مشکل می‌باشد. مدلسازی سیستم‌های همزمان و اجرای مدل (در واقع شبیه‌سازی آنها) بسیار پیچیده است و چنانچه نیاز به شبیه سازی واقعی داشته باشیم، باید هر کدام از اشیاء در یک پروسه دارای حیات جریان یابد.

می‌توان رفتار یک سیستم (رخ داد وقایع و نتایج حاصل) را پیش‌بینی نمود؟ پاسخ بله یا خیر به این مسئله در تقسیم‌بندی شبیه‌سازی سیستم‌ها از دیدی دیگر نقش اساسی دارد.

رفتار سیستم: معمولاً رفتار یک سیستم از دو حالت تصادفی و غیر تصادفی خارج نیست و بنابر تعریف عبارتند از:

۱- قطعیت: چنانچه بتوان رفتار یک سیستم را بطور قطع و تعیین پیش بینی کرد، آن رفتار را رفتار قطعی یا غیر تصادفی می‌نامند. (مانند دریافت روزنامه از دستگاه مربوطه در قبال انداختن سکه).

۲- عدم قطعیت: چنانچه نتوان رفتار یک سیستم را بطور قطع و تعیین پیش‌بینی کرد، آن رفتار را رفتار تصادفی یا غیر قطعی گویند (مانند پيامی که از گرات دیگر می‌آید).

پیش آمدهای قطعی^(۳) و غیر قطعی^(۴) از دید آن که تابع تصادف هستند یا تصادف در آن هیچ نقشی ندارد دسته بندی می‌شوند. درواقع سیستمی که نتوان رفتار آن را پیش بینی کرد یک سیستم تصادفی یا غیر قطعی است و اگر بتوان رفتارش را پیش‌بینی کرد یک سیستم قطعی است. برای روشن شدن این مطلب و پیش بینی رفتار یک سیستم به بیان مثالی می‌پردازیم. در این مثال به بررسی یک زبان ترتیبی (شبه گلد) برای اجرای یک مرتب سازی^(۵) غیر قطعی پرداخته و رفتار ۲ برنامه و نتایج اجرای آنها را بررسی می‌کنیم.

1- concurrent	2- parallel	3- determinism
4- nondeterminism	5- sort	

مثال ۱-۱ - مرتب سازی ۴ عدد بروش قطعی - مرتب سازی بابل^(۱)

{*a[1].a[2].a[3].a[4] = 1, 4, 3, 2*}

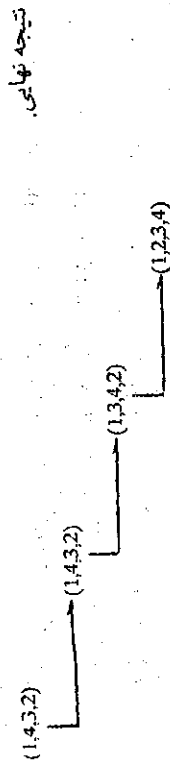
for i = 1 To 3

for j = i + 1 To 4

if a[i]>a[j] Then

swap a[i].a[j];

پس از اجراء با فرض داده‌های نمونه (1,4,3,2) نتیجه عبارتست از:



در این روش یک سلسله مشخص تا رسیدن به اعداد نهائی (مرتب شده) اتفاق می‌افتد و همیشه می‌توان این سلسله را پیش‌بینی کرد و رفتارش مشخص است. تصادف در این مسئله هیچ نقشی ندارد و سلسله حاصل همیشه یکسان و واحد است، با هر تکراری پیش‌بینی سلسله رخ داده‌ها ممکن است (قطعیت).

مثال ۱-۲ - مرتب سازی ۴ عدد بروش غیر قطعی^(۲) (با زبان DILAG). شرح مختصر چند دستور زبان DILAG:

● ستاره "*" به مفهوم حلقه تکرار (loop) است و تا وقتی دستورات آن اجرا می‌گردند که حداقل یک شرط داخل آن برقرار باشد.

● دستور شرطی با if شروع و با fi خاتمه می‌یابد. هر شرط داخل آن با علامت Condition [توأم است که در صورت درستی شرط، دستورات مقابل آن () اجرا می‌شوند.

● شروط داخل if به صورت یکسان بررسی می‌شوند و ترتیب خاصی برای تست شرط (در صورت صحت) و اجرای دستورات آنها وجود ندارد. هر کدام به صورت غیر قطعی (احتمالی) بررسی و اجراء می‌شوند.

در رویتن ترتیبی مثال ۱-۱ همیشه یک سلسله و یک راه قطعی برای جواب وجود دارد. در حالی که رویتن فعلی مسیرهای مختلف و متفاوتی ارائه می‌دهد که هیچکدام را نمی‌توان پیش‌بینی کرد. مثالی برای ارائه مفاهیم قطعیت و عدم قطعیت در برنامه‌نویسی ارائه گردید که نگرشی از دنیای برنامه‌نویسی ترتیبی^(۱) به دنیای برنامه‌نویسی موازی^(۲) است. ارائه این مفهوم که در دنیای مدل‌سازی سیستمهای همزمان و تبدیل آن به برنامه‌نویسی موازی جهت اجراء و توزیع داده‌ها، مفاهیم و روند متفاوت با مدل‌های ترتیبی و برنامه‌نویسی ترتیبی وجود دارد.

در جهان کنونی، رخداد پدیده‌ها اکثراً به صورت همزمان یا غیرقطعی می‌باشد و تصادف نقش اساسی در این سناریو ایفا می‌کند. چون در شیبه سازی سیستمهای واقعی هستیم لذا پدیده‌های تصادفی را بررسی می‌کنیم و نیاز به مولد^(۳) تصادفی پدیده‌ها داریم. مثلاً برای تولید رخ دادهای یک تاس، به مولد تصادفی با محدوده تولید اعداد ۱ الی ۶ نیاز است تا پرتاب یک تاس را تداعی کند.

انواع عدم قطعیت^(۳)

تصادفها سطوح خاصی دارند و بعضی رخ دادها از حیطه اختیارات و قدرت بشر خارج هستند. لذا بشر بعضی را خوشایند، بعضی را ناخوشایند و با احتمالاتی نامگذاری می‌کند. پس عدم قطعیت رخ دادها را می‌توان به صورت زیر نامگذاری کرد (این بررسی با توجه به مثال فرضی (رخداد خرابی)، در یک برنامه صورت می‌گیرد).

۱- عدم قطعیت پلید (demoniac): رخ دادی که وقوع آن منجر به عمل ناخوشایند قطع برنامه شود (ناخوشایند).

۲- عدم قطعیت فرشته‌ای (angelic): رخ دادی که منجر به قطع برنامه می‌شود ولی وقوع آن این امر را محقق نکند (خوشایند).

۳- عدم قطعیت ملون (Erratic): رخ دادی که وقوع آن ممکن است به قطع برنامه منجر شود یا نشود (رفتار نامشخص).

۴- عدم قطعیت احتمالاتی (probabilistic): رخ دادی که با احتمال %X ممکن است وقوع آن تحقق یابد.

۵- عدم قطعیت الهی (Oracle): رخ دادی که وقوع آن از اختیار بشر بیرون می‌باشد.

1- sequential programming

2- parallel programming

3- Random Generator

4- nondeterminism

● دستور $(a1, a2) = (a1, a2)$ یعنی همزمان $a1$ و $a2$ مقداردهی می‌شوند و اگر داشته باشیم $(a1, a2) = (a2, a1)$ مقادیر $a1$ و $a2$ همزمان تعویض می‌شوند.

● یک بلاک برنامه با "A" شروع و با "B" خاتمه می‌یابد.

رویتن مرتب سازی:

شروع حلقه

```

*
|
|
if
|| a1 > a2 → (a1, a2) : = (a2, a1) ;
|| a2 > a3 → (a2, a3) : = (a3, a2) ;
|| a3 > a4 → (a3, a4) : = (a4, a3) ;
دستور شرطی ۱
دستور شرطی ۲
دستور شرطی ۳

```

پایان حلقه

در این رویتن احتمال رخداد و بررسی یکسان هر کدام از شروط داخل if وجود دارد و چنانچه دستورات را شماره گذاری نمائیم (دستور شرطی ۱ و ...) احتمال اجرای هر دستور داخل if بطور یکسان است.

اجرای اول:

با دستور شرطی ۲ $(1,2,3,4)$ $(1,3,4,2)$ $(1,4,3,2)$ با دستور شرطی ۳

اجرای دوم:

با دستور شرطی ۲ $(1,2,3,4)$ $(1,4,2,3)$ $(1,4,3,2)$ با دستور شرطی ۳

دو راه رسیدن به پاسخ یعنی این که قطعاً بتوان سلسله رخ دادهای این رویتن را پیش‌بینی کرد و همان مفهوم عدم قطعیت است. به علت آن که احتمال بررسی و اجرای دستورات ۱ و ۲ و ۳ یکسان است لذا روندهای متفاوتی در اجراء پیش می‌آید.

مدل، ساده سازی و ایجاد یکخواختی و یگانگی است. الگوهای متفاوتی وجود دارد که اختلاف آنها در شیوه بیان و ارائه سیستم مرجع است.

۱.۵.۱ - انواع مدلها

با توجه به چگونگی بیان سیستم مرجع، می توان مدلها را به دو دسته کلی تقسیم کرد:

- ۱- الگوی ذهنی: هر نوع برداشت از یک سیستم مرجع در ذهن انسان را مدل ذهنی نامند.
 - ۲- الگوی صوری: ارائه یا بیان الگوی ذهنی به یک صورتی را مدل صوری نامند.
- در الگوهای ذهنی مسئله مهم همان استنباط و ادراک یا برداشت از موضوع مورد مطالعه است. این امر به مجموعه بایدها و نبایدها، همچنین جهان بینی، شیوه مطالعه و ادراک مدل ساز بستگی مستقیم دارد. الگوی صوری بیان آنچه که در ذهن وجود دارد به صورتی قابل بیان برای دیگران است و به انواع مختلفی تقسیم بندی می شود:

- ۲.۱ - مدلهای شماتیک: روابط و اجزاء مدل به صورت نمودار ارائه می شود و بلاکهای آن دارای روابط خاصی هستند (مانند فلوچارت).
- ۲.۲ - مدلهای فیزیکی: مدل ارائه شده دارای خواص فیزیکی از قبیل وزن و اشغال فضا است (مانند ماکت هواپیما).
- ۲.۳ - مدلهای نشانه ای: مدل با نشانه های خاصی بیان می شود، به صورت:

- ۲.۳.۱ - الگوی لفظی: ارائه در قالب الفاظ یا متون است. (مانند "قلم وسیله نگارش است").

- ۲.۳.۲ - الگوی گرافیکی: مدل به صورت تصویر یا خطوط گرافیکی خاصی ارائه می گردد و گرافهای سه بُعدی و مانند آن مدل را به واقعیت نزدیک تر می کند. (مانند "گراف سه بُعدی").

- ۲.۳.۳ - الگوی ریاضی: مدل به صورت فرمولها و روابط ریاضی ارائه می شود. مدل ریاضی صریح و واضح است و سازگاری دورنی دارد (مانند "معادلات دیفرانسیل").

مثال مدل ذهنی را می توان با برداشت ذهنی هر انسانی از "خول" و "فرشته" ارائه کرد لذا مدلهای ذهنی وابسته به قدرت تصور و ادراک افراد هستند. اگر هر فردی بخواهد تصور ذهنی خویش را بیان دارد، مثلاً "فرشته سمبل پاکی است"، می توان آن را مدل لفظی تلقی کرد. مدلهای دیگر از قبیل مدل فیزیکی و غیره، مصداقهای بسیاری دارند.

در مدلسازی وجود مدل جهت ارائه و شناخت سیستم مرجع با اهمیت است. انتخاب مدل مناسب مستلزم شناخت شیوه بیان رفتار سیستم مورد مطالعه می باشد. شناخت سیستم و مدلسازی آن به درجه بالایی از شناخت نیاز دارد. تبدیل رفتار سیستم به قالب مدل پیچیده است. بسیاری از

- ۶- عدم قطعیت فانی (Ephemeral): رخ دادی که در صریک سیستم لحظه ای وقوع یابد و محو شود (مانند خطای گذرا).

در واقع وقایع را می توان از حیث خوشایندی یا ناخوشایندی، کنترل یا عدم کنترل بشر و رفتار معین یا نامعین تقسیم بندی کرد. لذا باید رخ دادها و انواع آن در یک سیستم شیوه سازی به دقت بررسی شود. با توجه به چگونگی رخ داد وقایع، دسته بندیهای گوناگونی صورت می پذیرد ولی این نکته باید مد نظر قرار گیرد که بعضی پدیده های تصادفی خارج از کنترل بشر می باشد لذا باید به پارامترهای خارج کنترل در شیوه سازی دقت شود.

۱.۴ - عناصر سیستم

با نگرش شش گرا، عناصر یک سیستم در مرحله آنالیز به صورت زیر ارائه می گردد:

۱- موجودیتهای Entities

۲- اشیاء Objects

۳- صفات Attributes

۴- فعالیتهای Activities

۵- وقایع Events

۶- وضعیتهای States

۷- موضوعات Subjects

هر موضوع یا شیء یا موجودیت به ازاء پیش آمدی از حالتی به حالت دیگر می رود و فعالیت با فعالیتهای انجام می گیرد. وقایع و پیش آمدها حالت سیستم را تغییر می دهند و در قبال رخ داد آنها فعالیتهایی انجام می پذیرد که صفات بعضی اشیاء را عوض می کند.

۱.۵ - مدلسازی^(۱)

اساس و رکن اصلی شیوه سازی، مدلسازی یا انتخاب مدل است. انتخاب مدل مناسب پارامتر تعیین کننده ای است لذا در ابتدا باید انواع مدل، نکات و ظرایف آن را بطور کامل شناخت.

مدل:

"هر نوع ارائه یا بیان یک سیستم را مدل گویند." مدل رفتار سیستم را بیان می کند و از خواص

۹- آنالیز داده‌های تزیینی و نتایج چگونه صورت می‌گیرد؟ تئوری ریاضیات و آمار / استقلال

خروجی

۱۰- نقش تصادف در مدل‌سازی و شبیه‌سازی آن چیست؟ تصادفی / غیر تصادفی

۱۱- اگر مدل تواند اجراء کرده رفتار سیستم به یک زبان شبیه‌سازی تبدیل و روی کامپیوتر اجراء می‌شود به چه زبانی؟ X یا Y.

۱-۱- انواع سیستم‌ها

با توجه به تنوع بسیار سیستم‌ها هیچ اقدامی در زمینه طبقه‌بندی آنها نمی‌تواند کاملاً رضایت بخش باشد. به هر حال، از دو دید به سیستم‌ها می‌توان نگاه کرد: سیستم هدف انجام چه کاری را دارد؟ و چه امری باعث تغییر در سیستم می‌گردد؟ از دید اول، سیستم‌ها را به سیستم‌های با حفظ وضع موجود، هدفمند و خودآگاه تقسیم‌بندی می‌کنند. هدف این سیستم‌ها آن است که به گریه‌ای به تغییرات شرایط عکس‌العمل نشان دهند تا منجر به حفظ وضعیت از پیش تعیین شده گردد. هدف سیستم تئوری یک ساختمان نگهداری و حفظ دامنه درجه حرارت در محدوده از پیش مشخص شده است. مثال دیگر آن نیروهای انتظامی است که با حفظ نظم سعی در برقراری امنیت و ایجاد تعادل می‌کنند و در این جهت حفظ وضعیت امن می‌باشد. وظیفه این سیستم نگهداری وضعیت موجود و واکنش نسبت به تغییرات محیطی جهت بقا و وضعیت مشخص شده است.

سیستم‌های هدفمند قصد دارند تا به وضعیتی دست یابند که در حالت فعلی حائز آن نمی‌باشند. مثلاً سیستم آموزش و پرورش سعی در آموزش و افزایش معلومات محصلین دارند نه حفظ و بقا سطح معلومات فعلی آنها. در اثر کسب تجربه می‌توان اغلب سیستم‌های هدفمند را اصلاح و هدفهای والائی بر آنها تعریف کرد. سیستم‌های خودآگاه، خود تعیین کننده اهداف خویش هستند. این سیستم‌ها تحت شرایط از قبل تعیین شده و یا غیر آن مبادرت به تعیین اهداف و روشهای بیل به آنها می‌کنند. بارزترین مثال این سیستم‌ها، انسان می‌باشد.

از دید دوم، سیستم‌ها به واکنشی، پاسخی و خودکار تقسیم‌بندی می‌شوند. واکنش در داخل یک سیستم عبارت است از پیش آمدی که به طور قطعی توسط پیش آمد دیگری روی می‌دهد. مثال آن زدن کلید برق و روشن شدن لامپ است. در اثر پیش آمد زدن کلید، لامپ واکنش نشان می‌دهد و پیش آمد روشن شدن آن رخ می‌دهد. رابطه مورد استفاده، رابطه‌ای قطعی و علت و معلولی است.

در سیستمی که براساس پاسخ دچار تحول می‌گردد، یک پیش آمد می‌تواند علتی را سبب گردد ولی برای ایجاد معلول کفایت نمی‌کند. به عنوان مثال با تارک شدن هوا لامپ را روشن می‌کنیم. در

فلوچارت‌های متداول مدلهایی هستند که فاقد قدرت ارائه پیش آمدهای موازی می‌باشند. یعنی دو کار (Task) موازی را نمی‌توانند مدل‌سازی کنند. لذا برای مدل‌سازی سیستم‌های همزمان (دارای پیش آمدهای موازی) مناسب نیستند و کاربرد مدل‌سازی سیستم‌های سریال می‌دهد را دارند. یک مدل مناسب سیستم‌های همزمان مدل شبکه پتری (Petri net) است که بطور مفصل بیان خواهد شد.

۱-۵-۲- اجرای مدل

بعضی از مدل‌ها دارای قدرت و قابلیت اجراء نمی‌باشند یعنی نمی‌توان به آنها داده توزیع نمود و نتایج مورد نظر را اتخاذ کرد. مثلاً فلوچارت‌های متداول برنامه "فاقد این قدرت می‌باشند و صرفاً روابط و جریان حرکت را بیان می‌دارند. "شکله پتری"، مدلی است که در شکل اجرایی با اعمال داده و جرخش آن می‌تواند نتایج میانی و نهایی را جهت آنالیز رفتاری سیستم مدل شده بدست دهد. حتی می‌توان با این مدل، فلوچارت‌های مطروحه را نیز مدل‌سازی و اجراء کرد (بدون برنامه‌نویسی).

اگر مدل انتخابی فاقد قدرت اجرایی باشد، باید مدل سیستم تحت مطالعه به یک برنامه شبیه‌سازی خاص تبدیل شود و بر روی بستر کامپیوتر اجراء گردد. از این جهت بحث "زبانهای شبیه‌سازی" طرح می‌گردد که به آن خواهیم پرداخت. چون بستر اجرای زبانهای شبیه‌سازی، کامپیوتر است لذا این زبانها به "زبانهای شبیه‌سازی کامپیوتری" موسومند و دارای جهت یا گرایش شبیه‌سازی هستند. شرکتهای مختلفی بر روی کامپیوترهای متفاوت، زبانهای ارائه کرده‌اند که این بحث در فصل ۱۴ بطور کامل بیان شده است.

با توجه به مطالب ارائه شده می‌توان روند مورد نظر برای شبیه‌سازی یک مسئله را به صورت زیر خلاصه نمود و پاسخهای مورد نظر برای تمامی مراحل را به صورت مستدل و منطقی یافت:

- ۱- آیا سیستم مورد نظر یک سیستم همزمان است؟ بله/خیر.
- ۲- مدل مناسب انتخابی جهت مدل‌سازی چیست؟ مدل X یا Y.
- ۳- آیا مدل انتخابی قدرت مدل‌سازی سیستم مرجع را دارد؟ بله/خیر.
- ۴- مامیت رفتار پدیده‌های سیستم مرجع چیست؟ قطعی/تصادفی.
- ۵- آیا مدل سیستم مرجع کامل است؟ بله/خیر.
- ۶- تعیین عمر یا دوره شبیه‌سازی چگونه است؟ T واحد/ واحد.
- ۷- آیا مدل انتخابی قدرت اجراء یا دریافت داده دارد؟ بله/خیر.
- ۸- دریافت نتایج از توزیع داده‌ها چگونه است؟ جدول/گراف.

زمانی خاص رخ دهد.

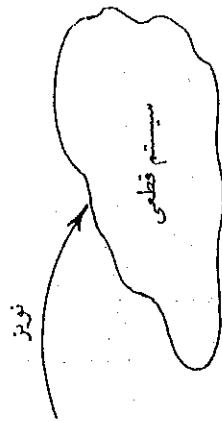
۱-۶-۳ - سیستم قطعی^(۱)

سیستم قطعی، سیستمی که رفتارش مشخص و شناخته شده است و هیچ پارامتر اتفاقی (تصادفی)^(۲) در آن نیست. رفتار این سیستمها بطور قطع و یقین معلوم است و احتمالات هیچ نقشی در تعیین پارامترهای سیستم ندارد. چنانچه ورودیهایش حاضر باشند خروجی مشخصی دارد و هیچ ورودی اتفاقی در آن رخ نمی دهد. مانند یک سیستم کنترلی است که رفتار آن مشخص و نویز ورودی ناشناخته بر آن هیچ اثری نمی گذارد.

۱-۶-۴ - سیستم تصادفی^(۳)

سیستم تصادفی، سیستمی که رفتارش وابسته به ورودیهای تصادفی است و دارای حداقل یک پارامتر ورودی تصادفی می باشد. یعنی حداقل رخ داد یک پدیده اتفاقی یا تصادفی در رفتار سیستم مؤثر است و نمی توان احتمالات را از معادله رفتاری آن خارج نمود. پس سیستمی که حداقل یک پارامتر ورودی تصادفی دارد یک سیستم تصادفی نامگذاری می شود.

مانند رفتار یک صف در بانک جهت سرویس مشتریان که ورود و خروج نفرات و زمان سرویس گرفتن آنها به صورت تصادفی می باشد. در سیستم تصادفی صف، اندازه گیری خروجی ها، میانگین نفرات، میانگین انتظار و غیره به عنوان تقریب آماری صحت عملکرد سیستم مدنظر است. زمانی که نویز در یک سیستم قطعی اثر می گذارد می توان آن را به یک سیستم تصادفی تشبیه کرد.



شکل ۱-۴ یک سیستم تصادفی

3- stochastic system

2- random

1- static system

واقع، نفس تاریکی هوا برای روشن شدن لامپ کفایت نمی کند مگر آنکه کلید خودکار باشد.

یک سیستم خودکار براساس تغییرات بطور خودکار متحول می شود. تغییرات خودکار پیش آمدهایی هستند که خود تعیین کننده خود هستند و برای آنها حفظ سوانح و پیش آمدهای قبلی ضرورت ندارد بخش عمده رفتار انسانی دارای چنین طبیعتی است.

البته در دسته بندی سیستمها نگرش و دیدهای مختلفی موجود و پارامترهای زیادی در این تقسیم بندی نقش دارند. مثلاً سیستم با زمان چه نسبتی دارد؟ زمان در آن چه اثری دارد؟ سیستم تصادفی یا قطعی است؟ تغییرات آن در زمان پیوسته است یا گسسته؟ تقسیم بندی سیستمها با نگرش های خاصی انجام می شود و یک تقسیم بندی دیگر را نیز می توان ارائه کرد:

۱- سیستم ثابت^(۱) یا سیستم پویا^(۲)

۲- سیستم تصادفی^(۳) یا سیستم قطعی^(۴)

۳- سیستم گسسته^(۵) یا سیستم پیوسته^(۶)

اکنون به شرح و بیان دسته بندی های می پردازیم.

۱-۶-۱ - سیستم ثابت

سیستم ثابت، سیستمی است که در آن هیچ پارامتری به زمان (t) بستگی ندارد. در واقع این سیستم وابسته به زمان نیست و با گذر زمان رفتار مشخص و شناخته شده ای دارد. ارتباطی بین هیچ عنصر این سیستم و زمان وجود ندارد و لذا در معادله رفتار سیستم متغیر یافت نمی شود. مانند تمامی سیستمهایی که در توالی زمان رفتار شناخته شده ای از خود بروز می دهند و سالیان سال رفتارشان نباید تغییر یابد.

۱-۶-۲ - سیستم پویا

سیستم پویا، سیستمی است که رفتارش با گذشت زمان متغیر می باشد. در واقع در معادله رفتاری سیستم یکی از پارامترهای مهم و حتمی (زمان) است. مانند رفتار یک صف در بانک جهت سرویس دهی به مشتریان که ورود و خروج صف در طول زمان متغیر می باشد یا معادله حرکت $x_0 + \dot{x} = v.t$ که به t وابسته است. ممکن است تغییرات آن سیستم در زمان به آرامی یا در برهه های

1- static system

2- Dynamic system

3- stochastic

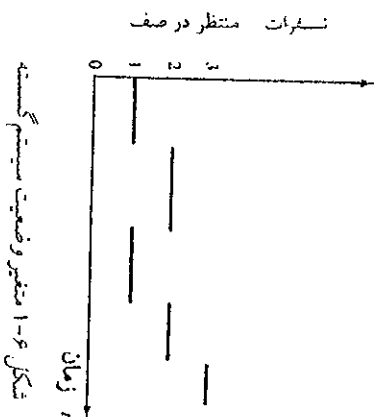
4- det rministic

5- discrete

6- continuous

۱.۶.۶- سیستم گسسته^(۱):

سیستم گسسته، سیستمی که در آن تغییر وضعیت‌ها در مقاطع زمانی (گسسته) صورت می‌گیرد و می‌توان جریان رخدادها را به صورت گسسته فرض نمود (مانند ورود مشتریان به صف که در مقاطع زمانی معین انجام می‌گیرد).

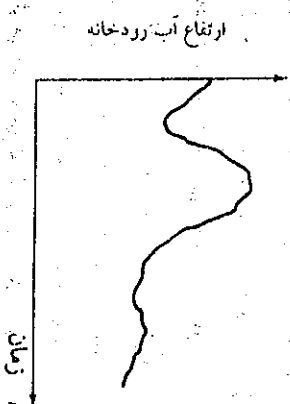


یعنی پیش‌آمدهایی که حالت سیستم را عوض می‌کنند در زمانهای گسسته اتفاق می‌افتند. لذا رخداد آنها را در مقاطع زمانی گسسته می‌توان در نظر گرفت و زمان را غیرپیوسته فرض کرد. پس متغیرهای تعیین وضعیت برای آنها، تنها در مجموعه‌ای از لحظات جدا از هم دچار تغییر می‌شوند.

نویز یک پارامتر تصادفی است که یک سیستم قطعی را به یک سیستم غیرقطعی تبدیل می‌کند. چون مامیت نویز همان تصادفی بودن است لذا در نهایت اگر سیستم قطعی با نویز تغییر وضعیت داشته باشد، یک سیستم تصادفی است. که حداقل یک پارامتر تصادفی (نویز) در سیستم دخالت می‌کند. شرایطی که یک سیستم قطعی را به یک سیستم تصادفی تبدیل کند به عنوان آشوب می‌شناسند.

۱.۶.۵- سیستم پیوسته^(۱)

سیستم پیوسته، سیستمی که در آن تغییر وضعیت‌ها^(۲) به آرامی یا پیوسته با زمان تغییر می‌کند (مانند جریان آب در یک رودخانه). تغییرات سطح آب در رودخانه به‌طور پیوسته با زمان تغییر داده می‌شود.



شکل ۱-۶-۵ متغیر وضعیت سیستم پیوسته

مدل ساخت رفتار سیستمهای پیوسته اغلب مدل سازی ریاضی است و معمولاً با معادلات دیفرانسیل بیان و حل می‌شوند. در این مثال متغیر تعیین وضعیت یعنی ارتفاع آب، در طی زمان بطور مداوم در حال تغییر است (شکل ۱-۶-۵).

فصل سوم

شیبه‌سازی سیستمهای گسسته

سیستمهای گسسته، سیستمهایی که می‌توان تغییرات آنها را ناپیوسته یا در زمانهای گسسته در نظر گرفت. عامل هر تغییر در وضعیت سیستم یک پیش آمد^(۱) نامیده می‌شود. مثلاً ورود یا خروج کالا به یک انبار یا اتمام سرویس یک کالا به عنوان پیش آمد تلقی می‌شود. شیبه‌سازی یک سیستم گسسته به عنوان شیبه‌سازی وقایع گسسته موسوم است و از سیستم پیوسته پیچیدگی کمتری دارد. گسستگی به فاصله رخ داد پیش آمدها متفاوت است و در هر سیستمی معنی و مفهوم خاصی دارد. اگر این فاصله طوری باشد که به سیستمهای پیوسته نزدیک گردد، سیستمهای پیوسته را به صورت پیش آمدهای گسسته می‌توان شیبه‌سازی کرد. پس جریان زمان یا گامهای افزایش برای پردازش رخ دادها در شیبه‌سازی پیش آمدها، عنصر اساسی است.

۳.۱- مدل‌های جریان زمان^(۲)

در شیبه‌سازی سیستم‌ها، مکانیزم جریان زمان وجود دارد که باید زمان سیستم را از مرحله آغازین، زمان صفر تا انتهای عملیات (پروژه شیبه‌سازی) حرکت داد و آن را با گامهایی سپری کرد. باید وضعیت سیستم را در لحظه‌هایی از زمان تغییر داد و وقتی که کل زمان سپری شد و پروژده شیبه‌سازی به اتمام رسیده، آن را قطع کرد. در سیستم‌های پیوسته زمان و تغییر وضعیت سیستم با یکدیگر پیوستگی دارند و زمان را با افزایش Δt حرکت می‌دهند و به انتهای شیبه‌سازی هدایت می‌کنند.

در شیبه‌سازی گسسته دو مدل جریان زمان داریم: "مدل گام ثابت"^(۳) و "مدل گام متغیر" و یا "واقعه بعدی"^(۴). در مدل گام ثابت با یک تایمر، ساعت سیستم، شیبه‌سازی می‌شود و این ساعت بوسیله ثابت زمانی τ به روز در می‌آید. وقایع رخ داده در انتهای این پروژدهای کوتاه زمانی (τ) بررسی و سرویس داده می‌شوند. تمامی پیش آمدهایی که در این پروژده رخ داده است مد نظر قرار می‌گیرند و

1- event

2- flow of time

3- fixed step

4- Next event

تعیین دقیق مقدار احتمال بسیار مشکل است و در بارهای موارد از طریق آتالیز ترکیبی می‌توان آن را محاسبه کرد. مثلاً در پرتاب n بار یک تاس k بار عدد ۴ بیاید. در مواردی که تعیین دقیق یک احتمال مؤثر نیست، می‌توان با داشتن فرکانس تکرار پیش آمدها احتمال تقریبی پیش آمد را بدست آورد. چنانچه یک آزمایش خاص n بار تکرار گردد و k بار یک پیش آمد E در n بار آزمایش رخ داده باشد، احتمال پیش آمد E به صورت زیر تفسیر می‌شود:

$$P(E) = \lim_{n \rightarrow \infty} \frac{k}{n}$$

پس با توجه به حد بالا، احتمال پیش آمد E در حالتی که تکرار آزمایش (n) بسیار زیاد است با عدد کسری $\frac{k}{n}$ تقریب زده می‌شود. در روشهای آزمایش امکان تکرار بسیار مهم است و روش‌ها از جهت خروجی‌های مورد آزمایش تقسیم‌بندی می‌شوند.

۳.۲.۲ - متغیرهای تصادفی^(۱) و فضای نمونه

در مدل‌های احتمالی، فرایندها یا متغیرهای تصادفی قابل پیش‌بینی نیستند. نتایج حاصل از یک فرایند یا متغیر تصادفی را فرایند نمونه، یک آزمایش، گویند. فضای نمونه براساس توزیع تعریف شده ایجاد گردیده است که توابع آن به توابع مولدهای تصادفی موسومند.

مثال ۱-۳- چنانچه نتیجه فرایند کنترل یک کامپیوتر "خراب" یا "سالم" باشد، می‌توان گفت:

{خراب}، $S = \{سالم\}$ فضای نمونه

متغیر تصادفی $X = \{0, 1\}$

$X = 0$: خراب و $X = 1$: سالم

و تعریف می‌شود که

ممکن است فضای نمونه زیاد باشد مثلاً

...، صحت کامل، خرابی نسبی، کاملاً خراب، $S = \{فضای نمونه\}$

متغیر تصادفی $X = \{0, 1, 2, \dots\}$

پس یک متغیر تصادفی X با یک نگاشت عناصر فضای نمونه را ایجاد می‌کند و عناصر نمونه می‌توانند گسسته، پیوسته یا ترکیبی باشند.

باعث تغییر حالت^(۱) سیستم می‌شوند. در مدل واقعه بعدی، زمان را "پیش آمد بعدی" تعیین و این پیش آمد ساعت شبیه‌سازی را معطم می‌کند که روشی بسیار سریع و کاراست لذا جریان زمان در شبیه‌سازی جهت اجرای سیستم و جمع‌آوری داده‌های عملیات، گام اساسی کار است.

۳.۲ - شبیه‌سازی تصادفی^(۲)

طبیعت سیستمهای پیرامون بشر و دنیای آن از شان (تصادف) برخوردار است و بخشهایی از آن توسط تصادف تعیین می‌شوند. احتمالات در این سیستمها نقش اساسی دارند و حداقل یکی از پارامترهای آن را تشکیل می‌دهند که به این سیستمها، سیستمهای تصادفی (Stochastic) گویند. این سیستمها رفتارشان ماهیتاً تصادفی و غیرتکراری است مانند سیستم دزدو پلیس، ورود مشتری به مغازه یا تقاضای خط تلفن و غیره. لذا سیستمهای گسسته را می‌توان به دو دسته تقسیم کرد:

۱- قطعی deterministic

۲- غیر قطعی Stochastic

پیاپی سازی سیستم تصادفی مشکل تر است زیرا مولدهای تصادفی خاص خود را نیاز دارد. برای حل سیستم‌های پویای پیچیده تصادفی گسسته، راه حل‌های تحلیلی می‌بایند. بنابراین تحت شبیه‌سازی مطالعه می‌شوند. لذا برای تصادفی‌ها نیاز به مولد تصادفی و شبیه‌سازی داریم که از توزیعهای مختلفی پیروی می‌کنند.

۳.۲.۱ - احتمال

احتمال معیاری است که درجه انتظار ما نسبت به وقوع یک پیش آمد را بیان می‌کند. در واقع تابعی است که پیش آمدها را در فضای اعداد حقیقی تصویر می‌کند و اندازه‌گیری یک احتمال با تابع $P()$ صورت زیر است:

● برای هر پیش آمد E احتمال آن چنین است $0 \leq P(E) \leq 1$

● احتمال فضای نمونه یا پیش آمدهای مطمئن $P(S) = 1$

● اگر E_1, E_2, \dots, E_n پیش آمدهای گسسته و جدا باشند، آنگاه

$$P(E_1 \cup E_2 \cup \dots \cup E_n) = P(E_1) + P(E_2) + \dots + P(E_n)$$

اگر X یک متغیر تصادفی گسسته باشد. با هر خروجی ممکن x_i در R_X یک عدد $p(x_i) = P(X=x_i)$ احتمالی را می‌دهد که متغیر تصادفی مساوی مقدار x_i باشد. تعداد $i=1,2,\dots$ باید دو شرط زیر را داشته باشد:

$$p(x_i) > 0 \quad \text{برای تمامی } i\text{ها}$$

$$\sum_{i=1}^{\infty} p(x_i) = 1$$

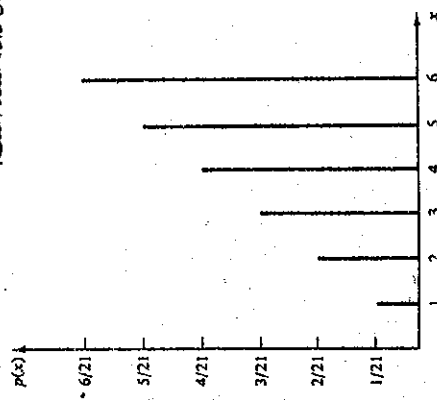
مجموع $i=1,2,\dots$ و $(x_i, p(x_i))$ تابع جرم احتمال یا توزیع احتمالی X است و $p(x_i)$ را تابع احتمال X گویند. (pmf)

مثال ۳-۳- آزمایش تالابم آب در یک فوت کردن به سطح آن را در نظر می‌گیریم. تعریف X به عنوان تعداد چینه‌های موج بعد از یک فوت کردن می‌باشد. پس $R_X = \{1,2,3,4,5,6\}$ فرض که موج با تعداد نقاط نشان داده شده است.

توزیع احتمال گسسته برای آزمایش تصادفی در ۲۱ آزمایش:

x_i	1	2	3	4	5	6
$p(x_i)$	1/21	2/21	3/21	4/21	5/21	6/21

که $i=1,\dots,6$ و $p(x_i) \geq 0$ و $\sum_{i=1}^6 p(x_i) = 1/21 + \dots + 6/21 = 1$ است. این توزیع به صورت گرافیکی در شکل ۳-۱ نشان داده شده است.



شکل ۳-۱ توزیع تجمعی احتمال برای مثال تالابم موج

فضای نمونه گسسته: چنانچه عناصر فضای نمونه قابل شمارش یا می‌شمارد یا بشمارد. فضای نمونه پیوسته: چنانچه عناصر فضای نمونه قابل شمارش نباشد یا نامشمار باشد. فضای نمونه ترکیبی: ترکیبی از فضای نمونه گسسته و پیوسته می‌باشد. در واقع در بسیاری از موارد مولدهای تصادفی دارای فضای نمونه گسسته یا پیوسته می‌باشند. در تولید تصادفی‌ها نیز می‌توان گفت متغیر تصادفی، نسبت است که از تابع توزیع یکپارخت یا غیر یکپارخت پیروی کند. می‌توان گفت متغیر تصادفی، نسبت دادن یک عدد حقیقی به هر یک از پیش آمدهای فضای نمونه است و توزیع احتمالی، قانونی است که به هر مقدار متغیر تصادفی یک احتمال نسبت می‌دهد. تعیین احتمال فضای نمونه گسسته و پیوسته متفاوت است.

● تابع احتمال فضای نمونه گسسته $P(x)$ عبارت است از:

$$P(x_i) = P(X=x_i)$$

یعنی متغیر تصادفی X با احتمالی دارای مقدار x_i است:

$$\sum_{i=1}^n P(x_i) = 1$$

مجموع احتمالات پیش آمدها یک است. احتمال هر پیش آمد مقداری کوچکتر یا مساوی ۱ است.

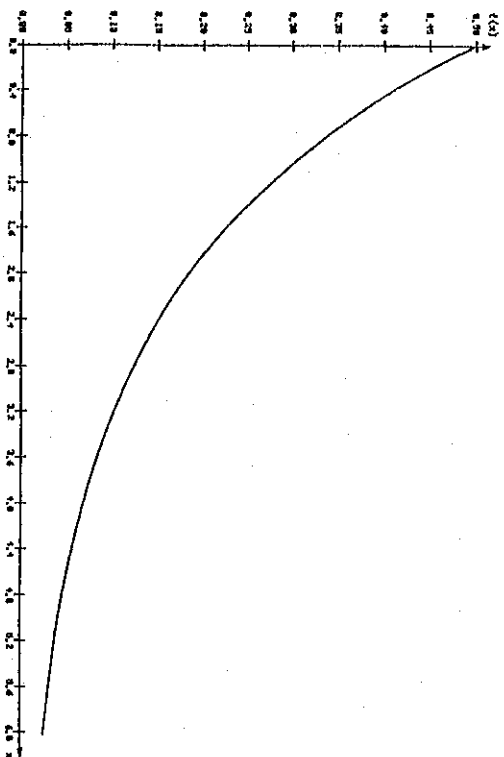
۳-۳-۱ مدل‌های آماری در شبیه‌سازی

۱- متغیرهای تصادفی گسسته، X را یک متغیر تصادفی در نظر می‌گیریم. اگر مقادیر ممکن X معین یا نامعین و قابل شمارش باشد، X یک متغیر تصادفی گسسته است. مقادیر ممکن X به این صورت فهرست می‌شود: x_1, x_2, \dots در حالت معین فهرست فوق انتها دارد و در حالت قابل شمارش نامعین فهرست بصورت نامحدود ادامه می‌یابد.

مثال ۳-۲- تعداد کارهایی که هر هفته به یک مغازه می‌رسد را مد نظر قرار می‌دهیم. متغیر تصادفی X = تعداد کارهایی که هر هفته می‌رسد $R_X = \{0,1,2,\dots\}$ مقادیر ممکن X در محدوده فضای R_X به وسیله R_X بیان می‌شود.

$$f(x) = \begin{cases} \frac{1}{2}e^{-x/2} & x \geq 0 \\ 0 & \text{در غیر آن} \end{cases}$$

این pdf گرانیکی در شکل ۳-۳ نشان داده شده است. گریند متغیر تصادفی X دارای توزیع نمایی با میانگین ۲ سال است.



شکل ۳-۳ PDF برای عمر لامپ CRT

احتمال آنکه عمر لامپ CRT بین ۲ و ۳ سال تعیین شود:

$$\begin{aligned} P(2 \leq X \leq 3) &= \frac{1}{2} \int_2^3 e^{-x/2} dx \\ &= -e^{-x/2} + e^{-1} = -0.223 + 0.368 = 0.145 \end{aligned}$$

۳- تابع توزیع تجمعی.

تابع توزیع تجمعی (CDF) به وسیله $F(x)$ ارائه می شود و احتمال را اندازه می گیرد. متغیر تصادفی X فرض می شود دارای مقداری کوچکتر یا مساوی x است، $P(X \leq x) = F(x)$ اگر X گسسته باشد:

$$F(x) = \sum_{x_i \leq x} p(x_i) \quad (\text{الف } ۳-۳)$$

و اگر X پیوسته باشد:

$$F(x) = \int_{-\infty}^x f(t) dt \quad (\text{ب } ۳-۳)$$

۳- متغیرهای تصادفی پیوسته. اگر محدوده فضای R_X از متغیر تصادفی X یک فاصله یا مجموعه ای از فواصل باشد، X را یک متغیر تصادفی پیوسته نامند. برای یک متغیر تصادفی پیوسته X، احتمال آن که

X در فاصله $[a, b]$ قرار گیرد:

$$P(a \leq X \leq b) = \int_a^b f(x) dx \quad (۳-۱)$$

تابع $f(x)$ تابع احتمال متغیر تصادفی X نام دارد با رصیتهای زیر:

- برای تمامی x در R_X ، $f(x) \geq 0$
- $\int_{R_X} f(x) dx = 1$
- اگر x در R_X نیست، $f(x) = 0$

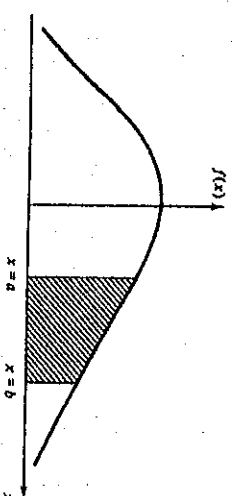
به عنوان یک نتیجه از معادله (۳-۱) برای هر مقدار معین x_0 ، $P(X=x_0) = 0$ است.

$$\int_a^a f(x) dx = 0$$

از این رو $P(X=x_0) = 0$ در معادلات زیر

$$P(a \leq X \leq b) = P(a < X \leq b) = P(a \leq X < b) = P(a < X < b) \quad (۳-۲)$$

ارائه گرانیکی معادله (۳-۱) در شکل ۳-۲ آمده است. ناحیه هاشور زده احتمال قرار گرفتن X در فاصله $[a, b]$ است.



شکل ۳-۲ ارائه گرانیکی $P(a \leq X \leq b)$

مثال ۳-۳- عمر یک لامپ انبوه کاندی (CRT) بر وسیله X نشان می دهد که یک متغیر تصادفی پیوسته برای تمامی مقادیر $x \geq 0$ می باشد. تابع چگالی احتمال (pdf) عمر لامپ را در سالها به صورت زیر است:

بعضی از خواص CDF به صورت زیر است:

$$\begin{aligned} \text{اگر } a < b \text{ سپس } F(a) &\leq F(b) \\ \lim_{x \rightarrow -\infty} F(x) &= 0 \\ \lim_{x \rightarrow \infty} F(x) &= 1 \end{aligned}$$

تمامی معادلات احتمالاتی X می‌تواند به صورت CDF بیان شود. برای مثال:

$$P(a < X \leq b) = F(b) - F(a)$$

(۳.۴) برای هر چه $a < b$

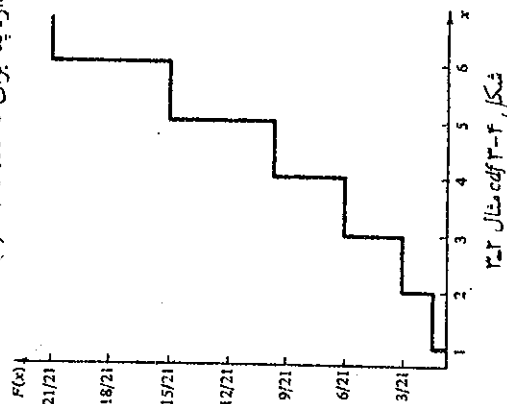
برای توزیع‌های پیوسته نه تنها معادله (۳.۴) برقرار است بلکه احتمال معادله (۳.۲) برابر: $F(b) - F(a)$

مثال ۳-۴- آزمایش شرح داده شده در مثال ۳-۲ یک CDF بصورت زیر دارد:

x	$(-\infty, 1)$	$[1, 2)$	$[2, 3)$	$[3, 4)$	$[4, 5)$	$[5, 6)$	$[6, \infty)$
$F(x)$	0	1/21	3/21	6/21	10/21	15/21	21/21

که $\{a \leq x < b\}$ است. این مثال در شکل گرافیکی ۳-۴ نشان داده شده است.

اگر X یک متغیر تصادفی گسسته با مقادیر ممکن $\dots, x_{i-1}, x_i, x_{i+1}, \dots$ که $x_1 < x_2 < \dots$ باشد و آن یک تابع پله‌ای است. مقدار cdf در فاصله $[x_{i-1}, x_i]$ ثابتند و پس یک پله می‌یزد یا با سایر $P(x_i)$ در x_i پرش می‌کند. بنابراین در مثال ۳-۴، اندازه پله برای $X=3$ ، $P(3)=3/21$ است.



شکل ۳-۴ مثال ۳-۴

مثال ۳-۵- cdf مثال ۳-۳ برای لامپ CRT:

$$F(x) = \frac{1}{2} \int_0^x e^{-t/2} dt = 1 - e^{-x/2}$$

احتمال آن که لامپ CRT برای کمتر از ۲ سال عمر کند:

$$P(0 \leq X \leq 2) = F(2) - F(0) = F(2) = 1 - e^{-1} = 0.632$$

احتمال آن که عمر CRT بین ۲ تا ۳ سال تعیین شود:

$$P(2 \leq X \leq 3) = F(3) - F(2) = (1 - e^{-3/2}) - (1 - e^{-1})$$

$$= -e^{-3/2} + e^{-1} = -0.223 + 0.368 = 0.145$$

که در مثال ۳-۳ یافت شد.

۴- امید ریاضی.

مهمترین مفهوم در تئوری احتمال، امید ریاضی متغیر تصادفی است. اگر X یک متغیر تصادفی باشد. مقدار امید ریاضی X به وسیله $E(X)$ ابرانه و برای متغیرهای گسسته و پیوسته به صورت زیر تعریف می‌شود:

(الف ۳.۵) اگر X گسسته باشد

$$E(X) = \sum_{i=1}^{\infty} x_i p(x_i)$$

و

(ب ۳.۵) اگر X پیوسته باشد

$$E(X) = \int_{-\infty}^{\infty} x f(x) dx$$

امید ریاضی متوسط وزن احتمالی کلیه مقادیر تصادفی X می‌باشد.

مقدار امید ریاضی $E(X)$ متغیر تصادفی X که به صورت میانگین ابرانه می‌شود به عنوان اولین گشتاور است. $E(X^n)$ برای $n \geq 1$ امین گشتاور X می‌باشد و به صورت زیر محاسبه می‌شود:

(الف ۳.۶) اگر X گسسته باشد

$$E(X^n) = \sum_{i=1}^{\infty} x_i^n p(x_i)$$

و

برای محاسبه $V(X)$ از معادله (۳۷) استفاده می‌کنیم، لذا اول $E(X^2)$ از معادله (ب) (۳۶) می‌شود:

$$E(X^2) = \frac{1}{2} \int_0^{\infty} x^2 e^{-x/2} dx$$

$$E(X^2) = -x^2 e^{-x/2} \Big|_0^{\infty} + 2 \int_0^{\infty} x e^{-x/2} dx = 8$$

بنابراین

$$V(X) = 8 - 2^2 = 4 \text{ years}^2$$

بنابراین

$$\sigma = \sqrt{V(X)} = 2 \text{ years}$$

با میانگین عمر ۲ سال و انحراف معیار ۲ سال است.

۵۵-د

از مُد در شرح چندین مدل آماری که در این فصل می‌آید استفاده می‌شود. در حالت گسسته، مُد مقدار متغیر تصادفی است که بیشترین رخ می‌دهد و در حالت پیوسته، ماکزیمم مقدار pdf است. مُد ممکن است واحد نباشد و اگر مقدار مُد در دو مقدار متغیر تصادفی رخ دهد، توزیع چند مُدی گفته می‌شود.

۳.۳.۲ - مدل‌های آماری مفید

رضیت‌های جدی در یک شبیه‌سازی جایی است که پیش آمدهای تصادفی فراهم شده و بتوان انتخاب کرد. در یک سیستم صف، زمان ورود و زمان سروس اغلب تصادفی است. در مدل انبار زمان بین تقاضاها و زمان تأخیرات (زمان بین یک سفارش و دریافت آن) و در یک مدل مطمئن نیز زمان خرابی ممکن است تصادفی باشد. در هر کدام از این سیستم‌ها، شبیه‌ساز نیاز به تولید پیش آمدهای تصادفی و استفاده از مدل آماری شناخته شده تحت توزیع دارد. در زیر مدل‌های آماری مقتضی این کاربردها شرح داده خواهد شد. به‌علاوه مدل‌های آماری مفید در حالت داده‌های محدود نیز منظور شده است.

۱- سیستم‌های صف

در یک خط انتظار (صف) زمان بین ورود و زمان سروس بطور تصادفی داده می‌شود گرچه داشتن زمان ثابت بین ورود (مانند سرهم کردن اتوبوس) یا زمان ثابت سروس (مانند زمان بستن چرخ

$$E(X^n) = \int_0^{\infty} x^n f(x) dx$$

(ب) اگر X پیوسته باشد

واریانس (گشتاور دوم) $E(X)$ متغیر تصادفی X که به صورت $V(X)$ یا σ^2 ارائه می‌شود:

$$V(X) = E[(X - E(X))^2]$$

تعیین مفید $V(X)$ می‌شود:

$$V(X) = E(X^2) - [E(X)]^2 \quad (۳۷)$$

میانگین $E(X)$ میزان اندازه‌گیری پراکندگی یک متغیر تصادفی و واریانس X اندازه‌گیری مویع امید ریاضی متغیر تصادفی از مقدار امید ریاضی آن است. بنابراین واریانس $V(X)$ یک میزان پراکندگی مقدار ممکن X حول میانگین $E(X)$ می‌باشد. انحراف معیار σ جذر واریانس σ^2 است. میانگین $E(X)$ و انحراف معیار عبارتند از:

$$\sigma = \sqrt{V(X)}$$

مثال ۳۶-۱: میانگین و واریانس مثال ۳۲ بصورت زیر تعریف می‌شود:

$$E(X) = 1\left(\frac{1}{21}\right) + 2\left(\frac{2}{21}\right) + \dots + 6\left(\frac{6}{21}\right) = \frac{91}{21} = 4.33$$

برای محاسبه $V(X)$ از معادله (۳۷) استفاده می‌کنیم، اول از معادله (الف) $E(X^2)$ می‌شود:

$$E(X^2) = 1^2\left(\frac{1}{21}\right) + 2^2\left(\frac{2}{21}\right) + \dots + 6^2\left(\frac{6}{21}\right) = 21$$

بنابراین:

$$V(X) = 21 - \left(\frac{91}{21}\right)^2 = 21 - 18.78 = 2.22$$

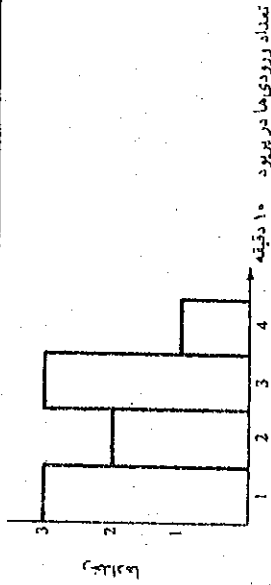
$$\sigma = \sqrt{V(X)} = 1.49$$

و

مثال ۳۷-۱: میانگین و واریانس عمر CRT مثال ۳۳ به صورت زیر است:

$$E(X) = \frac{1}{2} \int_0^{\infty} x e^{-x/2} dx = -x e^{-x/2} \Big|_0^{\infty} + \int_0^{\infty} e^{-x/2} dx$$

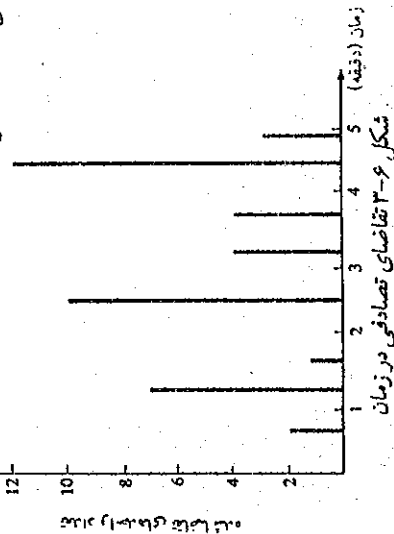
$$= 0 + \frac{1}{1/2} e^{-x/2} \Big|_0^{\infty} = 2 \text{ years}$$



شکل ۳-۵ هیستوگرام ورود در هر پریود زمانی

توزیع زمان بین ورودی‌ها و توزیع تعداد ورودی‌ها در هر پریود زمانی در شبیه‌سازی خط انتظار مهم است. زمان سرویس می‌تواند ثابت یا تصادفی باشد. اگر تصادفی است معمولاً توزیع نمایی برای شبیه‌سازی استفاده می‌شود ولی توزیع‌های دیگری نیز می‌تواند داشته باشد.

۲- سیستم‌های انبار
در سیستم‌های قابل انعطاف انبار سه متغیر تصادفی داریم: تعداد واحدهای مورد تقاضا در هر سفارش یا هر پریود زمان، زمان بین تقاضاها و زمان تأخیر. (زمان تأخیر به صورت زمان بین یک سفارش برای موجودی سیستم انبار تا دریافت آن سفارش است.) در بسیاری مدل‌های ریاضی ساده سیستم‌های انبار، تقاضا ثابت است و زمان تأخیر صفر یا ثابت فرض می‌شود؛ گویچه در بیشتر مواقع در مدل‌های شبیه‌سازی، تقاضا به صورت تصادفی در زمان رخ می‌دهد و تعداد واحدهای مورد تقاضا در هر زمان نیز تصادفی است (شکل ۳-۶)



شکل ۳-۶ تقاضای تصادفی در زمان

در عمل، توزیع زمان تأخیر اغلب می‌تواند به خوبی با یک توزیع گاما ارائه شود. توزیع‌های هندسی، پواسن و دو جمله‌ای محدوده توزیع‌ها برای الگوی تقاضا می‌باشد. توزیع هندسی حالت خاص دو جمله‌ای است.

اتوبوس توسط زبانت) نیز ممکن است. مثال زیر چگونگی رخ داد زمانهای بین ورود تصادفی را ارائه می‌کند.

مثال ۳-۸- مکانیک‌ها به مرکز ابزار مطابق جدول ۳-۱ وارد می‌شوند. بررسی تقاضا و دریافت ابزار مکانیک‌ها انجام می‌گیرد. جمع‌آوری داده‌ها در زمان ۱۰:۰۰ صبح شروع می‌شود و تداوم می‌یابد تا ۲۰ زمان ورود ثبت گردد. زمان واقعی ورود هر کدام باید محاسبه شود. پس اولین مکانیک در زمان صفر می‌رسد و دومی ۷:۱۳ (۷ دقیقه و ۱۳ ثانیه) بعد وارد می‌گردد و مانند آن.

ردیف	زمان بین ورود	تایم دقیق ساعت	دورانی
۱	—	10:05:03	1
2	7:13	10:12:16	2
3	3:32	10:15:48	3
4	8:39	10:24:27	4
5	7:52	10:32:19	5
6	3:24	10:35:43	6
7	4:08	10:39:51	7
8	0:39	10:40:30	8
9	0:47	10:41:17	9
10	2:55	10:44:12	10
11	1:33	10:45:47	11
12	5:00	10:50:47	12
13	9:18	11:00:05	13
14	4:53	11:04:58	14
15	1:14	11:06:12	15
16	5:11	11:11:23	16
17	5:08	11:16:31	17
18	0:47	11:17:18	18
19	4:08	11:21:26	19
20	3:17	11:24:43	20
21	6:36	11:31:19	21

جدول ۳-۱ داده‌های ورودها

مثال ۳-۹- ارائه دیگر داده‌های ورودی برای تعیین تعداد ورودیها در هر پریود زمانی می‌باشد. این ورودیها تقریباً در ۱/۵ ساعت رخ می‌دهد. در ۱۰ دقیقه پریود اول، یک ورود در ۱۰:۰۵:۰۳ رخ می‌دهد. در پریود دوم دو مکانیک وارد می‌شوند و مانند آن. نتایج در جدول ۳-۲ خلاصه شده است. این داده به صورت هیستوگرام در شکل ۳-۵ ترسیم شده است.

تعداد ورودی	پریود زمانی	تعداد ورودی	پریود زمانی
1	6	1	1
2	7	2	2
3	8	1	3
4	9	3	4
5	—	4	5

جدول ۳-۲ پریودهای زمانی ورودها

بنابراین

$$p(x_1, x_2, \dots, x_n) = p_1(x_1) \cdot p_2(x_2) \cdot \dots \cdot p_n(x_n)$$

و

$$p_j(x_j) = \begin{cases} p & x_j = 1, 2, \dots, n \\ 1-p=q & x_j = 0, 1, 2, \dots, n \\ 0 & \text{در غیر آن} \end{cases} \quad (3-8)$$

برای یک آزمون توزیع معادله ۳-۸ توزیع برنولی نام دارد.
می‌توانیم واریانس x بصورت زیر محاسبه می‌شود:

$$E(X) = 0 \cdot q + 1 \cdot p = p$$

$$V(X) = [(0^2 \cdot q) + (1^2 \cdot p)] - p^2 = p(1-p)$$

۲. توزیع دو جمله‌ای

متغیر تصادفی X تعداد موفقیت‌های در n آزمون برنولی را ارائه می‌کند که توزیع برنولی دارد. $P(x)$ جایی که:

$$P(x) = \begin{cases} \binom{n}{x} p^x q^{n-x} & x = 0, 1, 2, \dots, n \\ 0 & \text{در غیر آن} \end{cases} \quad (3-9)$$

معادله ۳-۹ تعیین احتمال تمامی موفقیت‌های خاص برپایه S در اولین x آزمون است و بدینال آن $(n-x)$ شکست می‌آید که با F نمایش می‌دهیم.

$$P(SSS \dots SSFF \dots FF) = p^x q^{n-x}$$

جایی که $q=1-p$ و

$$\binom{n}{x} = \frac{n!}{x!(n-x)!}$$

خروجی تعداد S و F را نیاز دارد. برای راحتی تعیین میانگین و واریانس در جمله‌ای، X به عنوان جمع n متغیر تصادفی مستقل برنولی هر کدام با میانگین p و واریانس $p(1-p)=pq$ می‌باشد، پس:

$$X = X_1 + X_2 + \dots + X_n$$

و میانگین $E(X)$ می‌شود:

۳- قابلیت نگهداری و اطمینان

زمان بین خرابی می‌تواند به وسیله توزیع خاصی بیان شود، مانند: نمایی، گاما و ویبل. اگر فقط خرابیهای تصادفی رخ دهد، توزیع زمان بین ورودی‌ها به صورت نمایی می‌تواند مدل شود. توزیع گاما برای مدل‌سازی افزونگی قطعات بدکی است که اجزاء، زمان خرابی نمایی دارند. توزیع ویبل برای زمان بین خرابی استفاده می‌شود برای وقتی است که مشاهدات مشهود است. وقتی یک تعدادی عنصر در یک سیستم با خرابی زیاد وجود دارند، توزیع ویبل به نظر می‌آید مناسب باشد. توزیع نرمال برای تعیین زمان خرابی بعضی عناصر در مدل‌های قابلیت اطمینان مناسب است.

۴- داده محدود

در بعضی موارد اهداف شبیه‌سازی قبل از جمع‌آوری داده‌های کامل شده، انجام گرفته است. سه توزیع یکپارخت، نمایی و بتا در داده محدود یا ناکامل کاربرد دارد. توزیع یکپارخت وقتی که زمان بین ورود و سروس به صورت تصادفی شناخته می‌شود و هیچ اطلاعی درباره توزیع بطور آبی در دست نباشد، فقط برای تعیین متغیر تصادفی پیرسته بکار می‌رود. از توزیع نمایی وقتی که تصور ماکزیم، می‌تیمم و حد وسط برای متغیر تصادفی متصور باشد استفاده می‌گردد و در نهایت توزیع بتا یک تغییر اشکال توزیعی در واحد ورودی است که با تعریف مقتضی به هر ورودی مورد نیاز منتقل می‌شود. توزیع یکپارخت حالت خاص توزیع بتا است.

۵- توزیع‌های دیگر

توزیع‌های دیگری که در شبیه‌سازی سیستم‌های گسسته ممکن است مفید باشند، توزیع‌های برنولی و دو جمله‌ای در توزیع‌های گسسته هستند. توزیع فوق نمایی مشابه توزیع نمایی است اما تغییر وسیع‌تری دارد و در کارهای مطمئن مفید می‌باشد.

۳-۳-۳- توزیع‌های گسسته

متغیرهای تصادفی گسسته برای تصادفی‌هایی استفاده می‌شود که مقادیر صحیح می‌گیرند. مثالهای بخش ۳-۲ مانند تقاضای انبار است. چهار توزیع در زیر شرح داده می‌شود.

۱. توزیع برنولی و آزمایشات برنولی

شرح یک آزمایش شامل چندین آزمون که هر کدام آنها می‌تواند با موفقیت یا شکست باشد $1 = \text{پُر}$ باشد یعنی از بین آزمایش با موفقیت همراه و $0 = \text{پریمی}$ نتیجه آزمایش با شکست توأم می‌باشد. n آزمون برنولی، یک پروسه برنولی است اگر آزمایشها مستقلند و هر آزمون در حالت ممکن (شکست یا موفقیت) دارد. احتمال یک موفقیت از یک آزمون به دیگری ثابت است.

$$p(x) = \begin{cases} q^{x-1}p & x = 1, 2, \dots \\ 0 & \text{در غیر آن} \end{cases}$$

پیش آمد. $\{X = x\}$ وقتی رخ می دهد که یک موفقیت یا - باختاری دنبال شود. هر خرابی احتمال $q = 1 - p$ و هر موفقیت احتمال p دارد. بنابراین:

$$P(FFF \dots FS) = q^{x-1}p \quad (3.12)$$

میانگین و واریانس می شود:

$$E(X) = \frac{1}{p} \quad (3.13)$$

$$V(X) = \frac{q}{p^2} \quad (3.14)$$

مثال ۳-۱۱ - موتاز میکروپروسسورها از خط بازرسی رد می شود. بیایید احتمال آن که اولین میکروپروسسور قابل قبول در سومین بازرسی باشد. هر بازرسی به عنوان آزمون برنولی با $q=0.4$ و $p=0.6$ فرض شود:

$$p(3) = 0.4^2(0.6) = 0.096$$

بنابراین فقط حدود ۱۰٪ وضعیتهای اولین میکروپروسسور قابل قبول در سومین بازرسی است.

۴. توزیع پواسون
توزیع پواسون برای شرح بسیاری از پروسه های تصادفی کاملاً مناسب و از نظر ریاضی ساده است. توزیع پواسون در سال ۱۸۳۷ بوسیله پواسون ارائه شد. تابع احتمال پواسون می شود:

$$p(x) = \begin{cases} \frac{e^{-a} a^x}{x!} & x = 0, 1, \dots \\ 0 & \text{در غیر آن} \end{cases} \quad (3.15)$$

جایی که $a > 0$ است. از خواص مهم توزیع پواسون آن که میانگین و واریانس آن a است:

$$E(X) = p + p + \dots + p = np \quad (3.10)$$

و واریانس $V(X)$ می شود:

$$V(X) = pq + pq + \dots + pq = npq \quad (3.11)$$

مثال ۳-۱۰ - پروسه ساخت محصول چیپ های نیمه هادی در میکروپروسسورها با میانگین ۲٪ عیب روبرو است. هر روز یک نمونه تصادفی با اندازه ۵۰ پروسه بررسی می شود. اگر نتیجه نظای بیشتر از دو عیب باشد، پروسه ساخت متوقف می شود. تعیین کنید احتمال آن که پروسه بوسیله نمونه برداری متوقف شود چقدر است؟

فرض پروسه نمونه برداری با آزمون برنولی $n=50$ که $p=2\%$ است و تعداد کل عیوب در نمونه X یک توزیع دو جمله ای به این صورت دارد:

$$p(x) = \begin{cases} \binom{50}{x} (0.02)^x (0.98)^{50-x} & x = 0, 1, 2, \dots, 50 \\ 0 & \text{در غیر آن} \end{cases}$$

محاسبه تعیین احتمال بیش از دو خرابی می شود:

$$P(X > 2) = 1 - P(X \leq 2)$$

احتمال $P(X \leq 2)$ عبارتست از:

$$\begin{aligned} P(X \leq 2) &= \sum_{x=0}^2 \binom{50}{x} (0.02)^x (0.98)^{50-x} \\ &= (0.98)^{50} + 50(0.02)(0.98)^{49} + 1225(0.02)^2(0.98)^{48} \\ &= 0.92 \end{aligned}$$

پس احتمال آن که پروسه محصول در هر روز متوقف شود تقریباً ۸٪ می باشد. میانگین تعداد عیوب در یک نمونه تصادفی با ۵۰ عضو هست:

$$E(X) = np = 50(0.02) = 1$$

و واریانس می شود:

$$V(X) = npq = 50(0.02)(0.98) = 0.98$$

۳. توزیع هندسی

توزیع هندسی به یک توالی آزمونهای برنولی مرتبط می شود، متغیر تصادفی X به عنوان تعداد آزمونهای اولین موفقیت تعریف می شود. توزیع X می شود:

مثال ۳-۱۴- زمان تأخیر تقاضای یک سیستم انبار مجتمع تقاضاهای یک واحد تا نقطه‌ای است که سفارش دریافت می‌شود. پس

$$L = \sum_{i=1}^{\infty} i D_i \quad (3.17)$$

جایی که L زمان تأخیر تقاضا، D_i تقاضا در انشای i امین پی‌رود است و T تعداد پی‌رودهای زمانی در انشای زمان تأخیر است. هر دو D_i و T ممکن است متغیرهای تصادفی باشند. یک مدیر انبار نیاز دارد تا احتمال آن که یک کاهش موجودی در انشای زمان تأخیر ناموفق باشد را بداند. برای مثال ممکن است احتمال کمبود در انشای زمان تأخیر از ۵٪ بیشتر نشود.

اگر زمان تأخیر تقاضا دارای توزیع پواسون باشد، تعیین نقطه سفارش مجدد آسان می‌شود. نقطه سفارش مجدد سطح انبار است وقتی که یک سفارش جدید انجام می‌شود. فرض کنیم که زمان تأخیر تقاضا دارای توزیع پواسون با میانگین $a=10$ واحد است و ۹۵٪ و حفاظت موجودی انبار نیاز است. بنابراین کمترین مقدار X که احتمال زمان تأخیر تقاضای X بیشتر یا مساوی ۰.۰۵ نشود.

$$P(X) = \sum_{i=0}^{\infty} \frac{e^{-10} 10^i}{i!} \geq 0.95$$

نتیجه در $X=15$ رخ می‌دهد که می‌تواند با استفاده از جدول مقادیر cdf یا محاسبه و $P(0), P(1), \dots$ یافت شود.

۳-۳-۴- توزیع‌های پیوسته

از متغیرهای تصادفی پیوسته برای شرح پدیده تصادفی که در یک محدوده‌ای می‌تواند هر مقداری بگیرد استفاده شود. مثال آن زمان بین خرابی یک وسیله برقی است. هفت توزیع را در این بخش شرح می‌دهیم.

۱. توزیع یکدخت

متغیر تصادفی X دارای توزیع یکدخت در فاصله $[a, b]$ است اگر pdf بشود:

$$f(x) = \begin{cases} \frac{1}{b-a} & a < x \leq b \\ 0 & \text{در غیر آن} \end{cases} \quad (3.18)$$

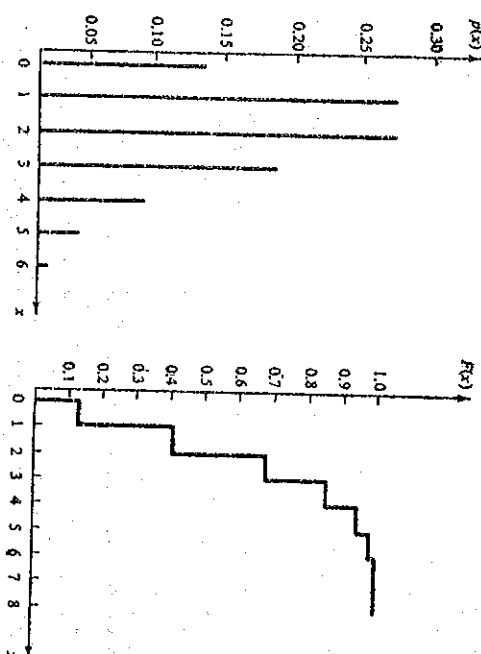
cdf می‌شود:

$$E(X) = a = V(X)$$

تابع توزیع تجمعی می‌شود:

$$F(x) = \sum_{i=0}^{\infty} \frac{e^{-a} a^i}{i!}$$

pmf و cdf توزیع پواسون با $a=2$ در شکل ۳-۷ نشان داده شده است.



شکل ۳-۷- pmf و cdf پواسون

مثال ۳-۱۲- یک ترمینال شخصی کامپیوتر هر بار که یک سرویس تقاضا می‌شود یک بوق می‌زند.

تعداد بوقها در هر ساعت مطابق توزیع پواسون با میانگین $a=2$ می‌باشد. احتمال سه بوق در ساعت بعدی بر وسیله معادله (3.15) با $x=3$ به صورت زیر است:

$$P(3) = \frac{e^{-2} 2^3}{3!} = \frac{(0.1353)(8)}{6} = 0.18$$

مثال ۳-۱۳- در مثال ۳-۱۲ تعیین احتمال ۲ یا ۳ بوق در پی‌رود یک ساعت:

$$P(2) = 1 - P(0) - P(1) = 1 - F(1) = 1 - 0.406 = 0.594$$

احتمال تجمعی $F(1)$ از سمت راست شکل ۳-۷ بدست آمد.

$$F(x) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ 1 & x > b \end{cases}$$

(۳.۱۹)

$$P(x_1 < X < x_2) = F(x_2) - F(x_1) = \frac{x_2 - x_1}{b - a}$$

توجه کنید که میانگین و واریانس توزیع می‌شود:

$$E(X) = \frac{a+b}{2}$$

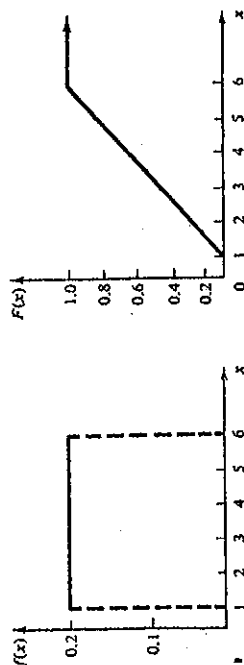
(۳.۲۰)

$$V(X) = \frac{(b-a)^2}{12}$$

و

(۳.۲۱)

pdf و cdf وقتی که $a=1$ و $b=6$ است در شکل ۳-۸ نشان داده شده است.



شکل ۳-۸ PDF و CDF برای توزیع یکتوخت

اعداد تصادفی دارای توزیع یکتوخت بین ۰ و ۱ می‌باشد.

مثال ۳-۱۵ - شبیه‌سازی عملیات یک ایستگاه ترافیک را داریم. در حدود هر سه دقیقه یک مراجعه برای اپراتور جهت یافتن مکان معین وجود دارد. فرض زمان بین مراجعات دارای توزیع یکتوخت با میانگین سه دقیقه است. بوسیله معادله (۳.۲۱) توزیع یکتوخت با میانگین ۳ یا پارامترهای $b=6$ و $a=0$ دقیقه می‌باشد. یک توزیع یکتوخت با بزرگترین واریانس معمولاً امن‌ترین توزیع است.

مثال ۳-۱۶ - یک اتوبوس هر ۲۰ دقیقه با شروع از ساعت ۶:۳۰ تا ساعت ۸:۳۰ در ایستگاه خاص وارد می‌شود. یک مسافر زمان‌بندی را نمی‌داند اما بطور تصادفی بین ۷:۳۰ و ۷:۴۰ هر روز صبح

می‌رسد. احتمال آن که مسافر بیش از ۵ دقیقه برای هر اتوبوس منتظر شود، چیست؟ مسافر بیش از ۵ دقیقه منتظر می‌شود اگر زمان ورود اولین ۷:۰۰ و ۷:۲۰ یا بین ۷:۳۰ و ۷:۴۰ باشد. اگر X متغیر تصادفی باشد که تعداد دقیق بعد از ۷:۰۰ را نشان می‌دهد که مسافر وارد می‌شود، احتمال آن:

$$P(0 < X < 15) + P(20 < X < 30)$$

حال X یک متغیر تصادفی یکتوخت بین (0,30) است. بنابراین احتمال مورد نیاز می‌شود:

$$F(15) + F(30) - F(20) = \frac{15}{30} + 1 - \frac{20}{30} = \frac{5}{6}$$

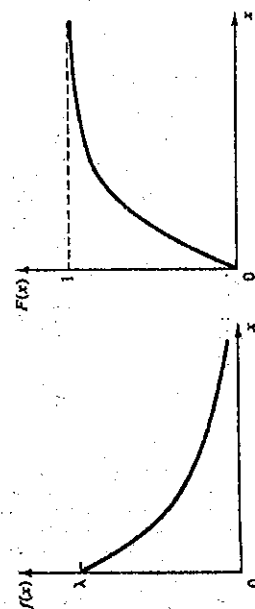
۲. توزیع نمایی

متغیر تصادفی X گفته می‌شود دارای توزیع نمایی با پارامتر $\lambda > 0$ می‌باشد اگر pdf آن:

(۳.۲۲)

$$f(x) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0 \\ 0, & \text{در غیر آن} \end{cases}$$

تابع چگالی احتمال در اشکال ۳-۹ و ۳-۱۰ آمده همچنین در شکل ۳-۹ cdf آن نشان داده شده است. توزیع نمایی بکار می‌رود که زمانهای بین ورود را وقتی که کاملاً تصادفی هستند مدل کند و مدل برای زمانهای سرویس می‌باشد وقتی که تغییرات زیاد است. در این منظور لریک فرخ است: ورود در هر ساعت یا سرویس در هر دقیقه. توزیع نمایی برای مدل عمر عناصری که خراب می‌شوند بکار می‌رود، پس لریک خرابی است.



شکل ۳-۹ تابع چگالی و تابع توزیع تجمعی نمایی

احتمال آن که متغیر تصادفی بیشتر از میانگین باشد 0.368 برای تمامی مقادیر λ است. احتمال آنکه لامپ تولیدی عمری بین ۲۰۰۰ تا ۳۰۰۰ ساعت داشته باشد:

$$P(2 \leq X \leq 3) = F(3) - F(2)$$

با معادله (۳.۲۲)

$$F(3) - F(2) = (1 - e^{-3/\lambda}) - (1 - e^{-2/\lambda}) \\ = -0.368 + 0.513 = 0.145$$

از مهمترین خواص توزیع نمائی این که "بدون حافظه" است منظور آنکه برای تمامی $s \geq 0$ و $t \geq 0$

$$P(X > s + t | X > s) = P(X > t)$$

(۳.۲۵)

X را عمر یک عنصر (یک لامپ نوری و...) در نظر بگیرید و فرض کنید که X دارای توزیع نمائی است. معادله (۳.۲۵) برای احتمال آنکه قطعه عمر حداقل $s+t$ ساعت و مانند آن می باشد که احتمال عمر حداقل t ساعت دارد. اگر قطعه در زمان s سالم باشد ($X > s$) پس توزیع باقیمانده زمان $X-s$ است که مانند توزیع اصلی قطعه جدید می باشد.

معادل (۳.۲۵) بوسیله احتمالات شرطی بیان می شود:

$$P(X > s + t | X > s) = \frac{P(X > s + t)}{P(X > s)}$$

(۳.۲۶)

معادله (۳.۲۶) می تواند برای تعیین غیر کسری معادله (۳.۲۶) باشد:

$$P(X > s + t | X > s) = \frac{e^{-\lambda(s+t)}}{e^{-\lambda s}} = e^{-\lambda t} \\ = P(X > t)$$

$$P(X > 3.5 | X > 2.5) = P(X > 1) = e^{-1/\lambda} = 0.717$$

۳. توزیع χ^2

یک تابع که توزیع گاما را تعریف می کند، تابع گاما است که برای تمامی $\beta > 0$ می شود

$$\Gamma(\beta) = \int_0^{\infty} x^{\beta-1} e^{-x} dx$$

(۳.۲۷)

با معادله (۳.۲۷) می شود

$$\Gamma(\beta) = (\beta - 1) \Gamma(\beta - 1)$$

(۳.۲۸)

در شکل ۳-۱۰ چندین PDF مختلف نمائی نشان داده شده است. مقدار محور افقی مساوی مقادیر λ است. توزیع نمائی دارای میانگین و واریانس زیر است:

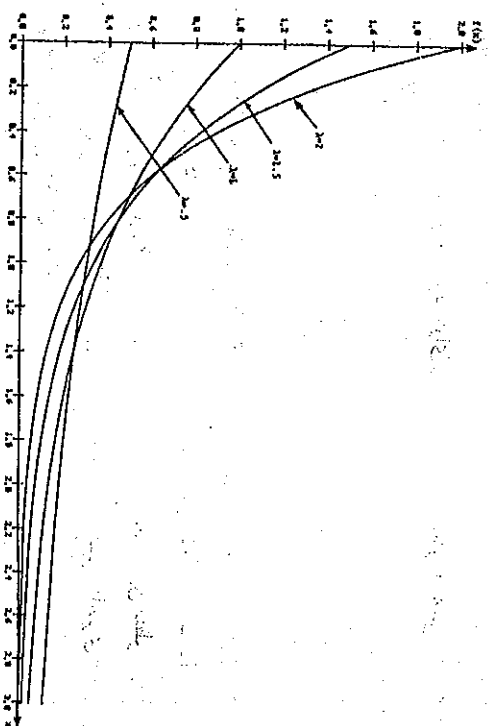
$$E(X) = \frac{1}{\lambda} \quad V(X) = \frac{1}{\lambda^2}$$

(۳.۲۳)

بنابراین میانگین و انحراف معیار برابر هستند. cdf می تواند با معادله (۳.۲۲) بدست آید.

$$F(x) = \begin{cases} 0 & x < 0 \\ \int_0^x \lambda e^{-\lambda t} dt = 1 - e^{-\lambda x} & x \geq 0 \end{cases}$$

(۳.۲۴)



شکل ۳-۱۰ PDFهای چندین توزیع نمائی

مثال ۳-۱۷- فرض کنید که عمر یک لامپ تولیدی در هزاران ساعت، دارای توزیع نمائی با نرخ خرابی $\lambda = \frac{1}{3}$ (یک خرابی هر ۳۰۰۰ ساعت) است. احتمال آن که لامپ عمر بیش از میانگین ۳۰۰۰ ساعت داشته باشد می شود:

$$P(X > 3) = 1 - P(X \leq 3) = 1 - F(3)$$

از معادله (۳.۲۴) برای محاسبه $F(3)$ استفاده می شود:

$$P(X > 3) = 1 - (1 - e^{-\lambda/3}) = e^{-1/3} = 0.368$$

X_i می شود:

$$F(x) = \begin{cases} 1 - \int_0^x \frac{\beta\theta}{\Gamma(\beta)} (\beta\theta t)^{\beta-1} e^{-\beta\theta t} dt & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (3.33)$$

وقتی که β صحیح است، توزیع گاما به توزیع نمائی متصل می شود. اگر متغیر تصادفی X جمع β متغیر تصادفی با توزیع نمائی با پارامتر $\beta\theta$ باشد پس X دارای توزیع گاما با پارامترهای β و θ است.

بنابراین اگر:

$$X = X_1 + X_2 + \dots + X_\beta \quad (3.34)$$

جایی که θ_i pdf می شود

$$g(x_i) = \begin{cases} (\beta\theta)e^{-\beta\theta x_i} & x_i \geq 0 \\ 0 & \text{در غیر آن} \end{cases}$$

X_i مستقلند و X دارای pdf معادله (۳.۳۰) است. وقتی $\beta=1$ نتیجه یک توزیع نمائی است.

۲. توزیع ارلنگ

pdf ارائه شده معادله (۳.۳۰) به عنوان توزیع ارلنگ با K وقتی که $K=\beta$ مقدار صحیح است می باشد. ارلنگ توسعه تئوری صف بود. توزیع ارلنگ بصورت زیر است: یک سری از K ایستگاه که به منظور تکمیل سروس یک مشتری بیان می شود. یک مشتری اضافی نمی تواند به ایستگاه اول وارد شود تا مسافر در این پروسه تمامی ایستگاهها را طی کند. هر ایستگاه زمان سروس توزیع نمائی و پارامتر $k\theta$ دارد. معادلات (۳.۳۱) و (۳.۳۲) برای تعیین میانگین و واریانس توزیع گاما با مقدار β است. مقدار امید ریاضی مجموع متغیرهای تصادفی، مقدار امیدهای ریاضی برای هر متغیر تصادفی است.

بنابراین

$$E(X) = E(X_1) + E(X_2) + \dots + E(X_k)$$

امید ریاضی توزیع نمائی X_i دارای مقدار $\frac{1}{k\theta}$ است. بنابراین

$$E(X) = \frac{1}{k\theta} + \frac{1}{k\theta} + \dots + \frac{1}{k\theta} = \frac{1}{\theta}$$

اگر β یک عدد صحیح باشد سپس $\Gamma(1)=1$ و معادله (۳.۲۸) می شود

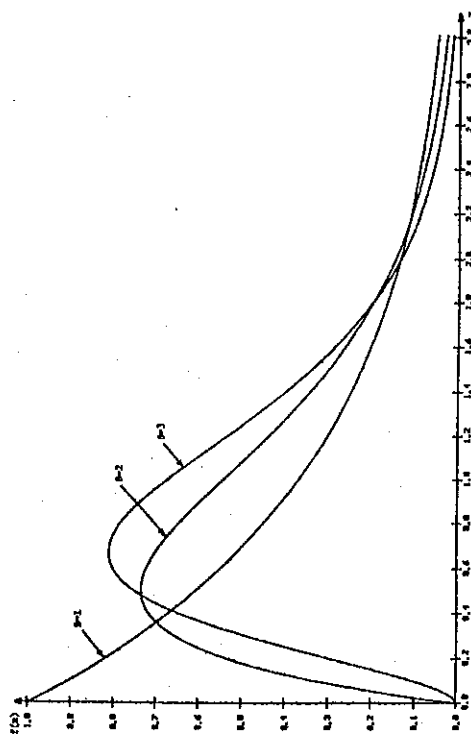
$$(3.29)$$

یک متغیر تصادفی X توزیع گاما با پارامتر β و θ دارد، اگر pdf باشد:

$$(3.30)$$

$$f(x) = \begin{cases} \frac{\beta\theta}{\Gamma(\beta)} (\beta\theta x)^{\beta-1} e^{-\beta\theta x} & x > 0 \\ 0 & \text{در غیر آن} \end{cases}$$

پارامتر β پارامتر شکل و θ پارامتر مقیاس است. در شکل ۳-۱۱ چندین توزیع گاما برای $\theta=1$ و مقادیر مختلف β نشان داده شده است.



شکل ۳-۱۱ pdf ها برای چندین توزیع گاما وقتی که $\theta=1$

میانگین و واریانس توزیع گاما می شود:

$$E(X) = \frac{1}{\theta} \quad (3.31)$$

و

$$V(X) = \frac{1}{\beta\theta^2} \quad (3.32)$$

$$F(50) = 1 - \sum_{i=0}^{\infty} \frac{e^{-(3)(1/60)} (3)(1/60)^{50} i!}{i!} \\ = 1 - \sum_{i=0}^{\infty} \frac{e^{-3/2} (5/2)^i}{i!}$$

از توزیع جمعیتی پواسون

$$F(50) = 1 - 0.543 = 0.457$$

احتمال آن که آزمایش زمان ۵۰ دقیقه یا کمتر ببرد ۰/۴۵۷ است. امید ریاضی از معادله (۳-۳۱) به

$$E(X) = \frac{1}{\theta} = \frac{1}{1/60} = 60 \text{ دقیقه}$$

صورت زیر تعیین می‌شود:

بعلاوه، واریانس $V(X) = 1/\theta^2 = 1200$ دقیقه است. معمولاً یک توزیع گاما از توزیع نمایی با میانگین یکسان، تغییرات کمتری دارد. عملاً متد توزیع ارلنگ می‌شود:

$$\text{Mode} = \frac{k-1}{k\theta} \quad (3-36)$$

بنابراین مقدار مد در این مثال است:

$$\text{Mode} = \frac{3-1}{3(1/60)} = 40 \text{ دقیقه}$$

۵. توزیع نرمال

متغیر تصادفی X با میانگین μ ($-\infty < \mu < \infty$) و واریانس $\sigma^2 > 0$ دارای توزیع نرمال است اگر دارای pdf

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{x-\mu}{\sigma} \right)^2 \right] \quad (3-37)$$

زیر باشد

$$-\infty < x < \infty$$

توزیع نرمال اغلب با نمایش $X \sim N(\mu, \sigma^2)$ بیان می‌شود که μ میانگین و σ^2 واریانس آن است. pdf نرمال در شکل ۳-۱۲ نشان داده شده است.

متغیرهای تصادفی X مستقلند، واریانس مجموع آنها جمع واریانسها است

$$V(X) = \frac{1}{(k\theta)^2} + \frac{1}{(k\theta)^2} + \dots + \frac{1}{(k\theta)^2} = \frac{1}{k\theta^2}$$

وقتی که $k = \lambda$ یک عدد صحیح مثبت است، $\theta = 1/\lambda$ بوسیله معادله (۳-۳۳) می‌شود

$$F(x) = \begin{cases} 1 - \sum_{i=0}^{k-1} \frac{e^{-k\theta x} (k\theta x)^i}{i!} & x > 0 \\ 0 & x \leq 0 \end{cases} \quad (3-35)$$

جمع بخشهای پواسون با میانگین $k\theta x = \lambda$ است.

مثال ۳-۱۹ - پرسنورهای یک کالج خانه را برای تابستان ترک می‌کنند اما درست دارند که یک لامپ روشن در تمامی زمان در خانه باشند. پرسنور یک دستگاه دارد که دو لامپ را روشن نگه می‌دارد. اگر اولین لامپ بسوزد دستگاه به دومی سوئیچ می‌کند. گفته می‌شود لامپ با میانگین عمر ۱۰۰۰ ساعت دارای توزیع نمایی است. پرسنورها ۹۰ روز (ساعت ۲۱۶۰) رفته‌اند. احتمال آن که نور موجود باشد وقتی که تابستان تمام می‌شود و پرسنورها برمی‌گردند، چیست؟

وقتی که تابستان تمام می‌شود $R(x)$ گویند که $R(x) = 1 - F(x)$ احتمال آن که سیستم حداقل x ساعت کار کند به آن تابع اطمینان $R(x)$ گویند که

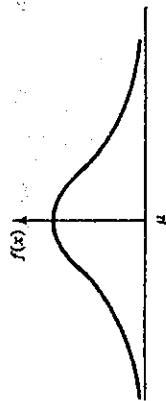
$$F(2160) = 1 - \sum_{i=0}^{\infty} \frac{e^{-(3)(1/2000)(2160)} (2)(1/2000)(2160)^i}{i!} \\ = 1 - e^{-2.16} \sum_{i=0}^{\infty} \frac{(2.16)^i}{i!} = 0.636$$

می‌گردد

در این حالت، کل عمر سیستم بوسیله معادله (۳-۳۴) با $k=2$ و $\theta=1/1000$ در هر ساعت و همچنین $\theta=1/2000$ هر ساعت بیان می‌شود. بنابراین $F(2160)$ از معادله (۳-۳۵) به صورت زیر تعیین

بنابراین ۳۶٪ شانس است که لامپ وقتی که پرسنور برمی‌گردند روشن باشد.

مثال ۳-۲۰ - یک آزمایش در سه مرحله بوسیله یک فیزیکدان انجام می‌شود. هر مرحله دارای توزیع نمایی با میانگین زمان سروس ۲۰ دقیقه است. احتمال آن را بیابید که آزمایش ۵۰ دقیقه یا کمتر زمان ببرد. همچنین امید ریاضی آزمایش را تعیین کنید. در این حالت $K=3$ مرحله و $k\theta=1/20$ که $\theta=1/60$ در هر دقیقه است. بنابراین $F(50)$ از معادله (۳-۳۵) به صورت زیر تعیین می‌گردد:



شکل ۳-۱۲ توزیع PDF نرمال

بعضی خواص توزیع نرمال بصورت زیر است:

● حد $f(x)=0$ است:

$$\lim_{x \rightarrow \pm\infty} f(x) = 0$$

● $f(\mu+x) = f(\mu-x)$ در حوالی μ متقارن است.

● ماکزیمم مقدار pdf در $x=\mu$ رخ می‌دهد (میانگین و مِد برابرند).

cdf توزیع نرمال می‌شود

$$F(x) = P(X \leq x) = \int_{-\infty}^x \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2\right] dt \quad (3.38)$$

حل معادله (۳.۳۸) ممکن نیست. روشهای عددی استفاده می‌شود، اما حل انتگرال برای هر جفت (μ, σ^2) ضروری است. گرچه یک تبدیل متغیر $Z=(x-\mu)/\sigma$ اجازه می‌دهد که حل مستقل از μ و σ باشد.

اگر $X \sim N(\mu, \sigma^2)$ و $Z=(x-\mu)/\sigma$ باشد:

$$\begin{aligned} F(x) &= P(X \leq x) = P\left(Z \leq \frac{x-\mu}{\sigma}\right) \\ &= \int_{-\infty}^{(x-\mu)/\sigma} \frac{1}{\sqrt{2\pi}} e^{-z^2/2} dz \\ &= \int_{-\infty}^{(x-\mu)/\sigma} \phi(z) dz = \Phi\left(\frac{x-\mu}{\sigma}\right) \end{aligned} \quad (3.39)$$

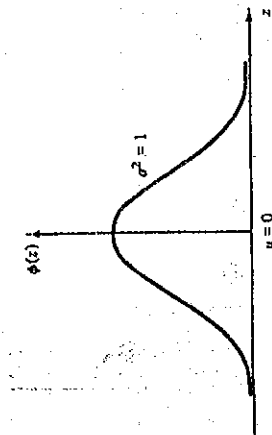
pdf توزیع نرمال با میانگین صفر و واریانس ۱ است:

$$\phi(z) = \frac{1}{\sqrt{2\pi}} e^{-z^2/2} \quad -\infty < z < \infty \quad (3.40)$$

بخش دوم / فصل سوم - شیوه‌سازی سیستم‌های گسسته

بنابراین $Z \sim N(0,1)$ گفته می‌شود که Z دارای یک توزیع نرمال استاندارد است. در شکل ۳-۱۳ توزیع نرمال استاندارد نشان داده شده است. cdf نرمال استاندارد می‌شود معادله (۴.۱.۳) مسطح شده است.

$$\Phi(z) = \int_{-\infty}^z \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt \quad (4.1.3)$$



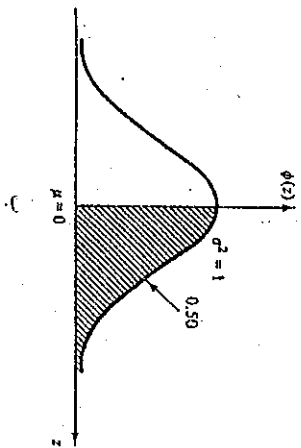
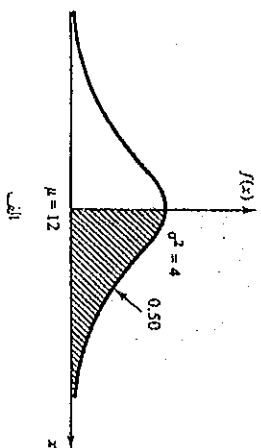
شکل ۳-۱۳ pdf توزیع نرمال استاندارد

مثال ۳-۲۱ - می‌دانیم که $X \sim N(50,9)$ است، تعیین $F(59)=P(X \leq 56)$ با معادله (۳.۳۹) $\Phi(2)=0.9772$ $\Phi(\frac{56-50}{3})=F(56)$ از جدول مربوطه نرمال بدست می‌آید. در شکل ۳-۱۴ تبدیل به نرمال استاندارد نشان داده شده است. در شکل (الف) $\Phi(2)$ و $\Phi(9)$ یا $X \sim N(50,9)$ مقدار $x_0=56$ مشخص است. بخش هاشورخورده احتمال مورد نیاز است. شکل (ب) $\Phi(2)$ نرمال استاندارد یا $Z \sim N(0,1)$ مقدار $x_0=56$ برای $z_0=3/2$ بزرگتر از میانگین است.

مثال ۳-۲۲ - زمان مورد نیاز باریگری کشتی اقیانوس پیمای X دارای توزیع $N(12.4)$ است. احتمال آن که کشتی در کمتر از ۱۰ ساعت بار شود با $F(10)$ میان می‌شود جایی که

$$F(10) = \Phi\left(\frac{10-12.4}{2}\right) = \Phi(-1.2) = 0.1587$$

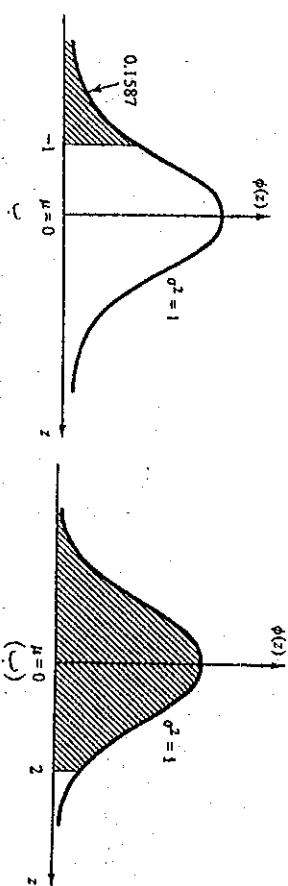
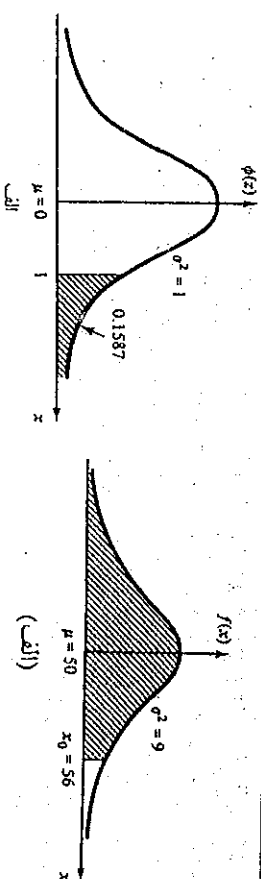
$\Phi(-1)$ از جدول توزیع نرمال بدست آمده است. و $\Phi(1)=0.8413$ مکمل $\Phi(1)$ در انتها است که در شکل (الف) $\Phi(1)$ نشان داده شده است. از شکل (ب) $\Phi(-1)$ می‌فهمیم که $\Phi(-x)=1-\Phi(x)$ $\Phi(-1)=1-\Phi(1)=0.1587$



شکل ۳-۱۶ تعیین احتمال به وسیله معاینه و باز دید

ناحیه نشان داده شده در (الف) ۳-۱۸ با احتمال $F(12) - F(9)$ می‌باشد. در شکل (ب) ۳-۱۸ احتمال معادل $\phi(-0.333) - \phi(0.667) = 0.7476$ برابر با جدول $\phi(0.667)$ است. پس با جدول $\phi(0.667) = 0.7476$ حال معادل $1 - 0.6504 = 0.3496$ است. بنابراین $\phi(0.333) = 0.3780$ است. احتمال آن که مشتری از صف در زمان بین ۹ تا ۱۲ دقیقه عبور کند، ۰.۳۷۸۰ است.

مثال ۳-۲۴ - (توزیع نرمال برشی) در بسیاری موقعیتهای عملی بنظر می‌رسد که دارای pdf نرمال است ولی متغیر تصادفی در ناحیه معینی قرار دارد. برای مثال متغیر تصادفی مقادیر منفی نمی‌گیرد (عمر لامپ، قیمت یک موجودی و مانند آن). برای مقادیر معین μ و σ^2 مقدار غیر منفی جدی نیست ولی وقتی که $\mu > 0$ و $\sigma > 0$ می‌شود از این رو $\phi(3) = 0.0044$ که فقط در حدود ۱ از ۱۰۰۰ وضیعت منفی تصادفی کمتر از صفر است.



شکل ۳-۱۵ خواص هندسی توزیع نرمال

شکل ۳-۱۴ تبدیل به نرمال استاندارد

احتمال آنکه زمان بازگویی ۱۲ ساعت یا بیشتر باشد از طریق معاینه و باز دید، از شکل (۳-۱۶) بدست می‌آوریم. حال $P(X > 12) = 1 - P(12) = 1 - 0.50 = 0.50$ است. احتمال آن که بین ۱۰ و ۱۲ ساعت باشد: $P(10 \leq X \leq 12) = P(12) - P(10) = 0.5000 - 0.1587 = 0.3413$

ناحیه مورد نظر در شکل (الف) ۳-۱۷ نشان داده شده است. مسئله معادل نرمال استاندارد در شکل (ب) ۳-۱۷ نشان داده شده است.

مثال ۳-۲۳ - زمان عبور از یک صف در یک رستوران به صورت $N(10, 9)$ یافت شد. احتمال آن که یک مشتری ورودی بین ۹ تا ۱۲ دقیقه منتظر شود:

$$P(9 \leq X \leq 12) = F(12) - F(9) = \Phi\left(\frac{12-10}{3}\right) - \Phi\left(\frac{9-10}{3}\right) = \Phi(0.667) - \Phi(-0.333)$$

بخش دوم / فصل سوم - شبیه‌سازی سیستم‌های گسسته

در بعضی موارد برش در سمت راست می‌باشد. فرض $X \sim N(5.0, 0.04)$ متغیر تصادفی X متغیر تصادفی $X \sim N(5.0, 0.04)$ است و نباید از مقدار $5/4$ تجاوز کند. لذا تمامی مقادیر بیش از $5/4$ در ریزه می‌شود. نتیجه متغیر تصادفی جدید Y است که یک توزیع نرمال برش خورده دارد. تابع توزیع تجمعی متغیر X با توزیع نرمال برشی (با برش راست در $x=r$) بصورت زیر است:

$$F(y) = \begin{cases} c \Phi\left(\frac{y-\mu}{\sigma}\right) & y \leq r \\ 1 & y > r \end{cases} \quad (3.42)$$

جایی که

$$c = \left[\Phi\left(\frac{r-\mu}{\sigma}\right) \right]^{-1} \quad (3.43)$$

در مثال $r=5.4$ و c از معادله (۳.۴۳) است:

$$c = \left[\Phi\left(\frac{5.4-5.0}{0.2}\right) \right]^{-1} = \frac{1}{0.9772} = 1.023$$

بنابراین $P(Y < 5.2)$ به صورت زیر تعیین می‌شود

$$\begin{aligned} P(Y < 5.2) &= F(5.2) = 1.023 \Phi\left(\frac{5.2-5.0}{0.2}\right) \\ &= 1.023 \Phi(1) = (1.023)(0.8413) = 0.8606 \end{aligned}$$

اگر متغیر تصادفی نرمال $X \sim N(\mu, \sigma^2)$ از چپ $X=L$ برش بخورد، pdf آن می‌شود

$$f(y) = \begin{cases} c \Phi\left(\frac{y-\mu}{\sigma}\right) & y \geq l \\ 0 & y < l \end{cases} \quad (3.44)$$

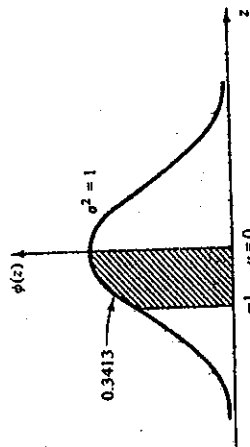
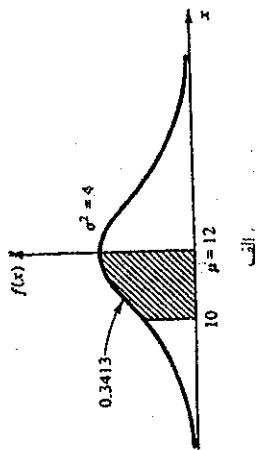
جایی که

$$c = \left[1 - \Phi\left(\frac{l-\mu}{\sigma}\right) \right]^{-1}$$

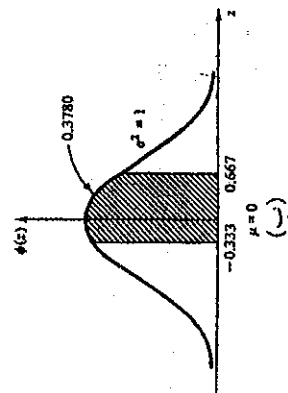
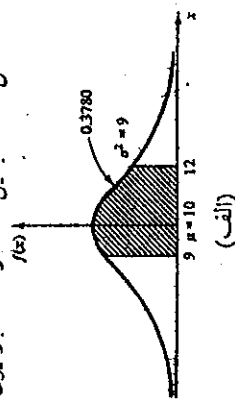
۶. توزیع ویبل

متغیر تصادفی X یک توزیع ویبل دارد اگر pdf آن به‌صورت زیر باشد:

$$f(x) = \begin{cases} \frac{\beta}{\alpha} \left(\frac{x-v}{\alpha}\right)^{\beta-1} \exp\left[-\left(\frac{x-v}{\alpha}\right)^\beta\right] & x \geq v \\ 0 & \text{در غیر آن} \end{cases} \quad (3.45)$$



شکل ۳-۱۷ تبدیل استاندارد مسئله بارگیری کشتی



شکل ۳-۱۸ تبدیل استاندارد مسئله کافی تریا

جایی که $F(0)$ به وسیله معادله (۳.۲۷) تعریف می‌شود. بنابراین پارامتر γ روی واریانس اثری ندارد گرچه میانگین با γ افزایش و کاهش می‌یابد. pdf توزیع ویبل می‌شود:

$$F(x) = \begin{cases} 0 & x < \gamma \\ 1 - \exp\left[-\left(\frac{x-\gamma}{\alpha}\right)^\beta\right] & x \geq \gamma \end{cases} \quad (3.28)$$

مثال ۳-۲۵- زمان بین خرابی یک عنصر الکترونیکی دارای توزیع ویبل با $\gamma=0$ ، $\beta=\frac{1}{3}$ و $\alpha=200$ می‌باشد. میانگین زمان خرابی به وسیله معادله (۳.۲۶) است

$$E(X) = 200(3 + 1) = 1200 \text{ hours}$$

احتمال آن که یک واحد کمتر از ۲۰۰ ساعت خراب شود از معادله (۳.۲۸) داریم

$$\begin{aligned} F(200) &= 1 - \exp\left[-\left(\frac{200}{200}\right)^{1/3}\right] \\ &= 1 - e^{-1} = 1 - e^{-0.333} = 0.2835 \end{aligned}$$

مثال ۳-۲۶- زمانی که یک هواپیمای از زمین بلند می‌شود و خط مسیر را ترک می‌کند، دارای یک توزیع ویبل با $\gamma=1034$ ، $\beta=0.5$ و $\alpha=0.04$ دقیقه است. احتمال آن که یک هواپیمای بیش از $1/5$ دقیقه از زمین بلند و خط مسیر را ترک کند چیست؟ در این حالت $P(X > 1.5)$ به صورت زیر تعیین می‌شود

$$\begin{aligned} P(X \leq 1.5) &= F(1.5) \\ &= 1 - \exp\left[-\left(\frac{1.5-1.34}{0.04}\right)^{0.5}\right] \\ &= 1 - e^{-2} = 1 - 0.135 = 0.865 \end{aligned}$$

$$P(X > 1.5) = 1 - P(X \leq 1.5) = 0.135$$

پس مقدار احتمال می‌شود:
۷. توزیع منطقی
متغیر تصادفی X دارای توزیع منطقی است اگر pdf آن باشد

$$f(x) = \begin{cases} \frac{2(x-a)}{(b-a)(c-a)} & a \leq x \leq b \\ \frac{2(c-x)}{(c-b)(c-a)} & b < x \leq c \\ 0 & \text{در غیر این صورت} \end{cases} \quad (3.29)$$

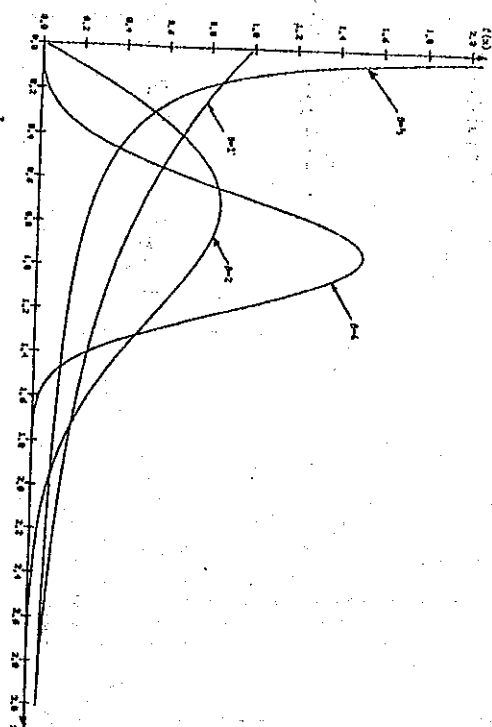
سه پارامتر توزیع ویبل γ ($0 < \gamma < \infty$)، مکان، α ($\alpha > 0$) پارامتر مقیاس و β ($\beta > 0$) پارامتر شکل می‌باشند. وقتی که $\gamma=0$ است، pdf ویبل می‌شود:

$$f(x) = \begin{cases} \frac{\beta}{\alpha} \left(\frac{x}{\alpha}\right)^{\beta-1} \exp\left[-\left(\frac{x}{\alpha}\right)^\beta\right] & x \geq 0 \\ 0 & \text{در غیر این صورت} \end{cases}$$

شکل ۳-۱۹ چند توزیع ویبل با $\gamma=0$ و $\alpha=1$ آمده است. با قرار دادن $\beta=1$ توزیع ویبل تقلیل می‌یابد:

$$f(x) = \begin{cases} \frac{1}{\alpha} e^{-x/\alpha} & x \geq 0 \\ 0 & \text{در غیر این صورت} \end{cases}$$

که توزیع نمایی با پارامتر $\lambda=1/\alpha$ می‌باشد.



شکل ۳-۱۹- PDFهای ویبل برای $\gamma=0$ و $\alpha=1$ و $\beta=\frac{1}{2}, 1, 2, 3, 5, 10$

میانگین واریانس توزیع ویبل با عبارت زیر می‌شود

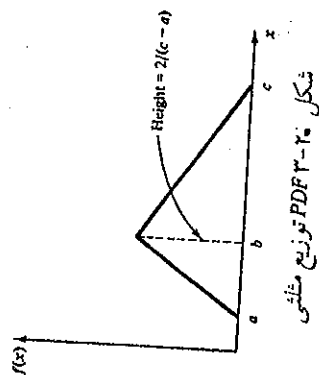
$$E(X) = \gamma + \alpha \Gamma\left(\frac{1}{\beta} + 1\right) \quad (3.30)$$

$$V(X) = \alpha^2 \left[\Gamma\left(\frac{2}{\beta} + 1\right) - \left[\Gamma\left(\frac{1}{\beta} + 1\right) \right]^2 \right] \quad (3.31)$$

جایی که $a \leq b \leq c$ است. مُد در $x=a$ می‌دهد. در شکل ۳-۲۰ pdf مثلثی نشان داده شده است. پارامترهای (a, b, c) می‌تواند به میانگین و مُد به صورت زیر تبدیل شود.

$$E(X) = \frac{a+b+c}{3}$$

(۳.۵۰)



از معادله (۳.۵۰) مُد به صورت زیر تعیین می‌شود

$$\text{Mode} = b = 3E(X) - (a + c)$$

(۳.۵۱)

از آنجا که $a \leq b \leq c$ است پس

$$\frac{2a+c}{3} \leq E(X) \leq \frac{a+2c}{3}$$

در توزیع مثلثی از مُد بیشتر از میانگین استفاده می‌شود. در شکل ۳-۲۰، ارتفاع بالای محور x می‌باشد. واریانس این توزیع $V(X)$ کم استفاده می‌شود. pdf برای توزیع مثلثی هست

$$f(x) = \begin{cases} \frac{(x-a)^2}{(b-a)(c-a)} & x \leq a \\ 1 - \frac{(c-x)^2}{(c-b)(c-a)} & a < x \leq b \\ \frac{(c-x)^2}{(c-b)(c-a)} & b < x \leq c \\ 0 & x > c \end{cases} \quad (3-52)$$

مثال ۳-۲۷ - پردازنده مرکزی برای اجرای برنامه‌ها دارای توزیع مثلثی با ثانیه $\mu = 0.05$ و $b = 1.1$ ثانیه $c = 6.5$ است. تعیین کنید احتمال اینکه زمان μ برای یک برنامه تصادفی 2.5 ثانیه یا کمتر باشد چیست؟ مقدار $F(2.5)$ بخشی از cdf در فاصله (1.1 و 0.05) بعلاوه فاصله (2.5 و 1.1) است.

با معادله (۳.۵۲)

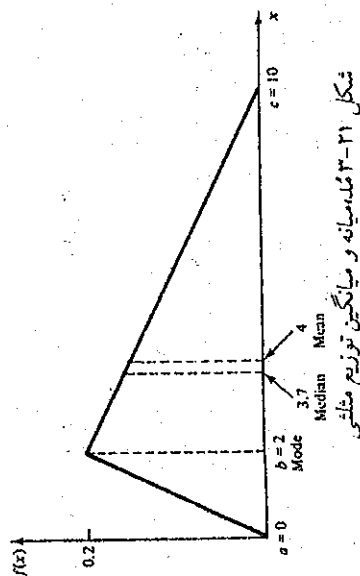
$$F(2.5) = 1 - \frac{(6.5 - 2.5)^2}{(6.5 - 0.05)(6.5 - 1.1)} = 0.541$$

بنابراین احتمال آن که μ زمان 2.5 ثانیه یا کمتر نیاز داشته باشد 0.541 است.

مثال ۳-۲۸ - از یک سنسور الکترونیکی برای تعیین کیفیت یک چپ نیمه هادی استفاده می‌شود و اگر خراب شود مورد قبول واقع نشده و رد می‌شود. براساس تقاضا، سنسور ماکزیمم و مینیمم تعداد رد شده‌ها در انشای هر ساعت در ۲۴ ساعت گذشته را می‌دهد. دپارتمان کنترل کیفیت فرض می‌کند که تعداد چپهای رد شده تقریبی، توزیع مثلثی دارد. داده‌های فعلی حداقل تعداد چپهای خراب رد شده را در انشای هر ساعت صفر و ماکزیمم ۱۰ برده است و دارای میانگین ۴ باشد. پس $a=0$ و $c=10$ و $E(X)=4$ مقدار b با استفاده از معادله (۳.۵۱) می‌شود

$$b = 3(4) - (0 + 10) = 2$$

ارتفاع مُد که در شکل ۳-۲۱ نیز ترسیم شده است $2(10-0)$ می‌باشد.



میان نقطه‌ای که 0.5 ناحیه سمت چپ و 0.5 ناحیه سمت راست آن باشد. در این مثال میانگین 3.7 است. ناحیه سمت چپ مُد با معادله (۳.۵۲) تعیین می‌شود:

$$F(2) = \frac{2^2}{20} = 0.2$$

بنابراین میانگین μ است. با قرار دادن $f(x)=0.5$ در معادله (۳.۵۲) و حل:

$$P\{N(t) - N(s) = n\} = \frac{e^{-\lambda(t-s)} [\lambda(t-s)]^n}{n!}$$

$$E[N(t) - N(s)] = \lambda(t-s) = \lambda[N(t) - N(s)]$$

زمان بین ورودی‌ها با پروسه پواسون رخ می‌دهد. اگر اولین ورودی در زمان A_1 رخ دهد، دومین ورودی در زمان $A_1 + A_2$ رخ می‌دهد و مانند آن که در شکل ۳-۲۲ نشان داده شده است. بنابراین، A_1, A_2, \dots زمانهای بین ورودیها هستند.



شکل ۳-۲۲ پروسه ورود

اولین ورودی بعد از زمان t رخ دهد اگر و فقط اگر در فاصله $[0, t]$ ورودی نباشد. بنظر می‌رسد که

$$\{A_1 > t\} = \{N(t) = 0\}$$

و بنابراین

$$P(A_1 > t) = P\{N(t) = 0\} = e^{-\lambda t}$$

احتمال این که اولین ورود در $[0, t]$ رخ دهد به وسیله معادله زیر داده می‌شود:

$$P(A_1 \leq t) = 1 - e^{-\lambda t}$$

که λt برای یک توزیع نمائی با پارامتر λ است. A_1 بصورت نمائی با میانگین $E(A_1) = 1/\lambda$ توزیع شده است. احتمال ورودی‌ها فقط به طول زمانهای بین ورودی‌ها بستگی دارد.

مثال ۳-۲۹- کارها به یک ماشین مطابق پروسه پواسون با میانگین، کار در هر ساعت $\lambda = 2$ وارد می‌شوند. بنابراین زمانهای بین ورودی‌ها بطور نمائی توزیع شده است و با ساعت $\lambda = \frac{1}{2}$ می‌باشد.

خواص پروسه پواسون

اولین خاصیت ارتباط تصادفی‌های جدا شده است. یک پروسه پواسون $\{N(t), t \geq 0\}$ با نرخ λ به وسیله بخش سمت چپ شکل ۳-۲۳ ارائه می‌شود.

شیه سازی

$$0.5 = 1 - \frac{(10 - 3.7)}{(10)(8)}$$

$$x = 3.7$$

۸۴
میانه x
که

این مثال نشان می‌دهد که میانه و میانگین ضرورتاً مساوی نیستند.

۳-۴- پروسه پواسون

پیش آمدهای تصادفی به صورت ورود کارها به یک مغازه، ورود هواپیما به پایانه، ورود کشتی به بندر و مانند آن می‌باشد. این وقایع می‌توانند به وسیله تابع شمارش $N(t)$ برای $t \geq 0$ تعریف بشود. این تابع شمارنده تعداد وقایعی است که در فاصله زمانی $[0, t]$ رخ می‌دهد. زمان صفر نقطه شروع است که مشاهده آغاز می‌شود. برای فاصله $[0, t]$ مقدار $N(t)$ یک مشاهده از متغیر تصادفی شرح است که می‌تواند بر وسیله $N(t)$ فرض شود مقدار صحیح $0, 1, 2, \dots$ است.

است که مقادیر ممکنه که می‌تواند بر وسیله $N(t)$ گفته می‌شود که پروسه پواسون با نرخ میانگین λ است اگر موارد زیر پروسه شمارش، $\{N(t), t \geq 0\}$ گفته می‌شود که پروسه پواسون با نرخ میانگین λ است اگر موارد زیر رعایت شود:

۱- ورود هر کدام از ورودی‌ها در یک زمان رخ می‌دهد.

۲- $\{N(t), t \geq 0\}$ افزایش ساکن دارد.

۳- $\{N(t), t \geq 0\}$ افزایش مستقل دارد.

اگر ورود مطابق پروسه پواسون باشد مورد بالا رعایت شده است احتمال آن که $N(t)$ برابر n باشد:

$$P\{N(t) = n\} = \frac{e^{-\lambda t} (\lambda t)^n}{n!}, \quad t \geq 0, \quad n = 0, 1, 2, \dots$$

(۳.۵۳)

با مقایسه معادلات (۳.۵۳) و (۳.۱۵) به نظر می‌رسد که $N(t)$ توزیع پواسون با پارامتر λt است.

بنابراین میانگین و واریانس عبارتند از:

$$E[N(t)] = \lambda t = \lambda t = \lambda t$$

$$n = 0, 1, 2, \dots$$

برای زمان t که $t \rightarrow \infty$ است فرض می‌شود که متغیر تصادفی $N(t)$ تعداد ورود در فاصله t دارای توزیع پواسون با میانگین λt است، بنابراین برای



شکل ۳-۲۴ پروسه ائتلافی

مثال ۳-۳۲ - (پروسه ائتلافی)

یک جریان ورودی پواسون با $\lambda_1 = 10$ و ورودی در هر ساعت با یک جریان ورودی پواسون با نرخ $\lambda_2 = 17$ در هر ساعت ترکیب می‌شود. پروسه ترکیبی یک پروسه پواسون با نرخ $\lambda = 27$ و ورودی در هر ساعت است ($\lambda = \lambda_1 + \lambda_2 = 27$).

۳.۵ - توزیع‌های تجربی

یک توزیع تجربی ممکن است به شکل پیوسته یا گسسته باشد. وقتی که غیرممکن بودن یا غیرضروری بودن هر توزیع شناخته شده خاص تشخیص داده شود از این امر استفاده می‌شود. یک مزیت استفاده از یک توزیع شناخته شده در شیمی‌سازی، ساده‌سازی پارامترهایی است که آنالیز محسوس را فراهم می‌کند.

۳.۶ - استقلال و یکنواختی

۳.۶.۱ - استقلال

اگر گشتاور m حول میانگین (μx) تعریف شود:

$$E((x - E(x))^m)$$

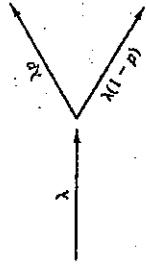
گشتاور دوم که مهم است و واریانس نام دارد، برای حالت گسسته و پیوسته بدین شرح است:

$$\sigma^2 x = E[(X - \mu x)^2] = \sum (X - \mu x)^2 P(x)$$

$$\sigma^2 x = E[(X - \mu x)^2] = \int (X - \mu x)^2 f(x) dx$$

فضای نمونه گسسته

فضای نمونه پیوسته



شکل ۳-۲۳ تقسیم تصادفی

فرض کنید هر زمان که یک پیش آمد رخ می‌دهد در پیش آمدهای نوع ۱ و یا ۲ دسته‌بندی می‌شود و پیش آمد نوع ۱ با احتمال p و پیش آمد نوع ۲ با احتمال $(1-p)$ رخ می‌دهد. $N_1(t)$ و $N_2(t)$ متغیرهای تصادفی هستند که تعداد پیش آمدهای نوع ۱ و ۲ در $[0, t]$ رخ می‌دهد را ارائه می‌کند. توجه شود که $N(t) = N_1(t) + N_2(t)$ است. هر دو پروسه‌های $N_1(t)$ و $N_2(t)$ پواسون با نرخ λp و $\lambda(1-p)$ مانند شکل ۳-۲۳ می‌باشند و هر دو مستقلند.

مثال ۳-۳۳ - (تقسیم تصادفی)

فرض کنید که کارها مطابق توزیع پواسون به یک مغازه می‌رسند (رانرخ λ). فرض کنید که هر ورودی با احتمال $\frac{1}{3}$ به عنوان الویت بالا و با احتمال $\frac{2}{3}$ به عنوان کم اولویت. جلالت گذاری می‌شوند. سپس پیش آمد نوع ۱ به ورودیهای با الویت بالا و پیش آمد نوع ۲ به ورودیها با الویت پایین مرتبط می‌شوند. اگر $N_1(t)$ و $N_2(t)$ به صورت بالا تعریف شوند هر دو متغیرهای دارای توزیع پواسون با نرخهای $\frac{\lambda}{3}$ و $2\frac{\lambda}{3}$ می‌باشند.

مثال ۳-۳۱ - نرخ مثال $(\lambda = 3)$ هر ساعت ۳ است. احتمال آن که هیچ ورودی با الویت بالایی در یک پرویه ۲ ساعته وارد نشود، بوسیله توزیع پواسون با پارامترهای $\alpha = 4pt = 2$ داده شود. پس

$$P(0) = \frac{e^{-2} 2^0}{0!} = 0.135$$

اکثرن حالت متضاد تقسیم تصادفی را فرض کنید که پرویه ائتلافی از دو جریان ورودی است. این پرویه در شکل ۳-۲۴ نشان داده شده است. اگر $N(t)$ متغیر تصادفی مستقل پرویه‌های پواسون با نرخ λ_1 برای $t=12$ باشد، $N(t) = N_1(t) + N_2(t)$ یک پرویه پواسون با نرخ $\lambda_1 + \lambda_2$ است.

$$P(x_1) = P(x_2) = \dots = P(x_n)$$

$$P(x_1) + P(x_2) + \dots + P(x_n) = \sum_{i=1}^n P(x_i) = 1$$

چون داریم که

$$P(x) = \begin{cases} \frac{1}{n} & x_1, x_2, \dots, x_n \\ \phi & \text{سایر} \end{cases}$$

لذا کلیه توزیع‌های با این شرایط و تابع احتمال را توزیع یکواخت گسسته نامند.

ب- پیوسته: اگر تصادفی تولیدی در محدوده $[A, B]$ احتمال آن که تصادفی‌های تولیدی در فاصله $[C, D]$

برابر باشد یا:

$$\frac{D-C}{B-A}$$

این تصادفی‌ها را دارای توزیع یکواخت پیوسته نامند و به این صورت نیز نشان می‌دهند:

$$U_i \in [0, 1]$$

که در فاصله ۰ و ۱ عدد تصادفی حاصل را در نظر می‌گیرند. تابع توزیع تجمعی آن می‌شود:

$$F_X(t) = \begin{cases} 0 & t \leq A \\ \frac{t-A}{B-A} & A \leq t \leq B \\ 1 & t \geq B \end{cases}$$

تابع چگالی احتمال آن است:

$$F_X(x) = \begin{cases} \frac{1}{B-A} & A \leq t \leq B \\ 1 & \text{سایر} \end{cases}$$

که برای $U_i \in [0, 1]$ می‌شود:

$$F_X(x) = \begin{cases} 1 & A \leq t \leq B \\ 0 & \text{سایر} \end{cases}$$

۳.۷- استفاده از اعداد تصادفی در فضای نمونه

جهت تولید عناصر فضای نمونه از اعداد تصادفی استفاده می‌شود که با یک مثال مشابه ارتباط این متغیرهای تصادفی و فضای نمونه را بیان می‌کنیم. اگر فرض کنیم برای تولید وضعیت قیمت اجناس در

کم بودن واریانس نشان می‌دهد که نمونه‌های تصادفی بیشتر حول میانگین رخ داده‌اند.

اگر x و y دو متغیر تصادفی باشند، کورلیانس بین آنها برابر است با:

$$\text{Cov}(x, y) = E[(x - \mu_x)(y - \mu_y)] = \int \int [(x - \mu_x)(y - \mu_y) f(x, y)] dx dy$$

$$= \int [(x - \mu_x) f(x)] \left[\int (y - \mu_y) g(y|x) dy \right] dx$$

فضای نمونه پیوسته.

$$\text{Cov}(x, y) = \sum \sum (x - \mu_x)(y - \mu_y) P(xy)$$

فضای نمونه گسسته:

استقلال نمونه‌های y و x یعنی آن که کورلیانس آنها صفر شود و لذا هیچ همبستگی ندارند. زیرا از

$$g(y|x) = f(y)$$

استقلال y و x نتیجه می‌شود:

$$\int (y - \mu_y) g(y|x) dy = \int (y - \mu_y) f(y) dy = 0$$

پس

پس می‌توان گفت که استقلال تصادفی‌های y و x یعنی $\text{Cov}(xy) = 0$ است. در واقع کورلیانسهمبستگی را تعیین می‌کند و ضریب همبستگی بین y و x به صورت زیر تعیین می‌شود:

$$r_{xy} = \frac{\text{Cov}(x, y)}{\sigma_x \sigma_y}$$

هرچه مقدار این ضریب بیشتر باشد، همبستگی تصادفی‌های y و x بیشتر می‌شود. (σ_x) را انحراف معیار x گویند و جذر واریانس است. چنانچه مقدار r_{xy} صفر باشند گویند x و y مستقل از یکدیگرند.

۳.۸- یکواختی

الف- گسسته: اگر فرض کنیم که متغیرهای تصادفی x_1, x_2, \dots, x_n دارای احتمال یکسان باشند:

بازار دارای حالت‌های زیر باشیم:

افزایش: یعنی قیمت جنس افزایش یافته است.

کاهش: یعنی قیمت جنس کاهش یافته است.

ثبات: یعنی قیمت جنس تغییر نکرده است.

پس فضای نمونه می‌شود: {ثبات، کاهش، افزایش} $S =$ حال اگر فرض کنیم که با احتمال $1/2$ افزایش قیمت و $1/3$ کاهش قیمت و $1/5$ ثبات قیمت داریم، چگونه مولد تصادفی ایجاد کنیم که خروجی آن این چنین عمل کند؟

نگاشت:

اگر فرض کنیم مولد تصادفی داشته باشیم که اعداد ۰ تا ۹ را تولید کند پس می‌توان احتمالات فوق را این چنین اعمال کرد:

$$S_1 = \{0,1\} \quad \text{افزایش}$$

$$S_2 = \{2,3,4\} \quad \text{کاهش}$$

$$S_3 = \{5,6,7,8,9\} \quad \text{ثبات}$$

لذا خروجی مولد تصادفی ۰ تا ۹ در جداول فضای نمونه: افزایش (S_1) ، کاهش (S_2) و ثبات (S_3) بررسی می‌گردد. هر کدام که در فضای نمونه مربوطه باشد، عنصر فضای نمونه S را با احتمالات ۰.۲ و ۰.۳ می‌سازند. در فضاهای نمونه زیر احتمالات به این صورت اعمال شده است:

$$P(S_1) = \frac{S_1}{S} = \frac{2}{10} = 0.2$$

$$P(S_2) = \frac{S_2}{S} = \frac{3}{10} = 0.3$$

$$P(S_3) = \frac{S_3}{S} = \frac{5}{10} = 0.5$$

$$P(S) = P(S_1) + P(S_2) + P(S_3) = 1$$

بطور کلی جهت تطبیق احتمالات در تولید تصادفی‌های نهایی، نوعی نگاشت از فضای کلی نمونه‌ها به فضای نمونه کوچکتر داریم و این طراحی در هر مسئله متفاوت است. پس دو گام داریم:

الف- تولید عدد تصادفی ابتدایی.

ب- تطبیق با جداول نگاشت و تولید تصادفی فضای نمونه.

۳-۸- شبه تصادفی ^(۱)

اگر روش تولید اعداد تصادفی شناخته شده باشد و بتوان سری اعداد تصادفی تولیدی را تکرار کرد، سلسله تولیدی تصادفی نیست و می‌توان گفت شبه تصادفی است. پس اعداد تصادفی تولیدی ما اعداد شبه تصادفی هستند و از حیث آن که رخداد آنها را بتوان تکرار کرد ماهیت تصادفی بودن آن مرتفع می‌گردد.

اعداد شبه تصادفی برای مواردی از قبیل خطایابی ^(۲) بکار می‌رود و باید بتوان تصادفی‌های قبلی را دوباره تولید نمود. اگر بتوان تکرار کرد، واقعاً تصادفی بوده‌اند و در پارامی مولد و پست‌دیده نیست. زیرا برای بررسی روند عملکرد مولد و آزمونه‌ای یکتواختی، استقلال و مانند آن به سلسله اعداد تصادفی تولیدی و تکرار آن نیاز می‌باشد. اعداد تصادفی تولیدی ممکن است یکتواخت و مستقل باشند یا از توزیع خاصی پیروی کنند. اعداد تصادفی بوسیله کامپیوتر تولید می‌شوند و به عنوان بخشی از شبه‌سازی به حساب می‌آیند. در نهایت می‌توان گفت مولد تصادفی رفتاری دارد که اگر شرائط آن فراهم شود، رفتارش تکرار می‌گردد.

۳-۹- روتین‌های مولد تصادفی

روتین‌های مولد تصادفی در کامپیوتر باید دارای مشخصات خاصی باشند. از قبیل:

- ۱- سریع و ارزان باشند تا به سرعت مولدهای تصادفی را تولید کنند و هزینه زیادی نداشته باشند.
- ۲- عدم تخصیص هسته حافظه، یعنی عدم حضور وابستگی به حافظه خاص یا محدوده معینی از آن چون احتمال خرابی حافظه، کاهش حافظه، حذف ترونسپ روتین‌های دیگر می‌رود.
- ۳- پیروی طولانی تولید اعداد تصادفی، تا تکرار اعداد تصادفی در پیروی طولانی تری صورت گیرد.
- ۴- عدم وابستگی به محیط خاص باشد و به دور از محیطی که سیموله می‌شود، اجراء گردد.
- ۵- دقت محاسبات در تولید توالی‌های مورد نظر جهت صحت عملکرد را دارا باشد.

لذا روتین‌های برنامه‌نویسی شده یعنی مولدهای تصادفی خوب و قابل نقل و انتقال هستند، احتمال خرابی روتین از جهت نرم‌افزار و سخت‌افزار را کاهش داده‌ایم و مسئله دقت را به عنوان یکی از عوامل صحت عملکرد به حساب می‌آوریم.

فصل چهارم

مولدهای اعداد تصادفی یکنواخت

مولدهای اعداد تصادفی

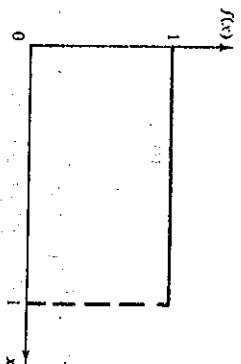
مولدهای تصادفی جهت تولید اعداد تصادفی یکنواخت می‌روند و از اعداد تولیدی با نتایج حاصل جهت سیموله پارامترهای تصادفی استفاده می‌گردد. تصادفی‌های تولیدی ممکن است دارای توزیع مشخص باشند و از روند خاصی پیروی نمایند یا آن‌که از استقلال و یکنواختی برخوردار باشند.

می‌توان مولدهای اعداد تصادفی را به دو دسته تقسیم کرد:

- ۱- مولدهای اعداد تصادفی یکنواخت.
- ۲- مولدهای اعداد تصادفی غیر یکنواخت.

در مولدهای تصادفی یکنواخت، سلسله تصادفی‌های تولیدی دارای خاصیت یکنواختی و استقلال می‌باشند. هر عدد تصادفی به عنوان نمونه مستقلی از یک توزیع آماری یکنواخت (پیوسته) با پارامترهای صفر و یک محسوب می‌گردد. تابع چگالی احتمال چنین متغیر تصادفی عبارتست از:

$$f(x) = \begin{cases} 1 & 0 \leq x \leq 1 \\ 0 & \text{سایر} \end{cases}$$



که به عنوان استقلال یکنواختی و خواص توزیع آماری پیوسته برای آن می‌توان موارد زیر را ذکر کرد:

۱- اگر فاصله (0,1) به m بخش مساوی تقسیم گردد، انتظار می‌رود که از N مشاهده، $\frac{N}{m}$ آن در هر بخش قرار گیرد (یکنواختی).

۲- احتمال حصول یک مشاهده در هر بخش مستقل از سایر مشاهدات است (استقلال).

۴-۱- مولدهای اعداد تصادفی یکنواخت

مولدی را مولد تصادفی یکنواخت گویند که اعداد تصادفی تولیدی آن از توزیع یکنواخت برخوردار و مستقل از یکدیگر باشند. برای این منظور باید خروجی های مولد تصادفی یکنواخت را از جهت استقلال و یکنواختی آزمایش کرد. در این فصل ابتدا به چند نمونه از مولدهای یکنواخت اشاره می شود و سپس روشهای آزمون را ارائه می کنیم. البته بعضی از روشهای تولید اعداد تصادفی به خاطر ایرادهای آن امروزه منسوخ شده اند و ارائه آنها جنبه معرفی و مقایسه دارد.

۴-۱-۱- مولدهای همبستگی

و c را همبستگی مدول m گوئیم؛ چنانچه باقیمانده تقسیم آنها بر m برابر باشند و گویند a همبستگی c به پیمانه m است.

$$a \equiv c \pmod{m}$$

$$18 \equiv 8 \pmod{5}$$

مثال ۴-۱- ۸ و ۱۸ را همبستگی مدول ۵ گوئیم:

این فرمول اساس مولدهای همبستگی را تشکیل می دهد که دارای روشهای ضربی، جمع و مرکب می باشد. در فرمول کلی آن اگر $c > 0$ باشد روش همبستگی مرکب و اگر $c = 0$ همبستگی ضربی گویند:

$$X_{i+1} \equiv (ax_i + c) \pmod{m}$$

۴-۱-۱-۱- مولد همبستگی مرکب

$$X_{i+1} \equiv (ax_i + c) \pmod{m} \quad i=1, 2, \dots, c > 0$$

در این فرمول برای تولید رشته اعداد تصادفی به X_0 نیاز داریم و تقسیم $(ax_i + c)$ به m مقدار ادامه می یابد تا باقیمانده آن از m کمتر شود. پس عدد بعدی X_{i+1} برابر مقدار این باقیمانده می گردد لذا پروسه با X_0 آغاز می شود که به آن مقدار اولیه گویند. از نظر تجربی مولد رابطه عبارت است از $X_{i+1} = ax_i + c \pmod{m}$ که با معرف بزرگترین عدد صحیح موجود در $\frac{ax_i + c}{m}$ می باشد.

مثال ۴-۲- با روش همبستگی مرکب و پارامترهای $m=7$ ، $X_0=3$ ، $c=3$ و $a=5$ مولدی نوشته و اعداد تصادفی تولید می کنیم.

بخش دوم / فصل چهارم - مولدهای اعداد تصادفی یکنواخت

$$X_{i+1} = (5X_i + 3) \pmod{7}$$

$$X_{i+1} = (5X_i + 3) \pmod{7}$$

$$0 \leq x \leq m-1, 0 \leq \frac{x}{m-1} \leq 1$$

i	۰	۱	۲	۳	۴	۵	۶	۷
X_i	۳	۴	۲	۶	۵	۰	۳	۴
R_i		۱	۲	۶	۵	۰	۳	۴

فاصله تکرار اعداد تولیدی را دوره مولد (r) نامند که در این مثال $r=6$ است. اگر پارامترهای a و c تغییر کنند، دوره مولد نیز تغییر می کند.

مثال ۴-۳- با روش همبستگی مرکب و پارامترهای $m=7$ و $X_0=3$ و $c=10$ و $a=18$ مولد اعداد تصادفی می شود:

$$X_{i+1} = (18X_i + 10) \pmod{7}$$

i	۰	۱	۲	۳	۴	۵	۶	۷	۸
X_i	۳	۱	۰	۳	۱	۰	۳	۱	۰
R_i		۱	۰	۳	۱	۰	۳	۱	۰

مثال ۴-۴- با روش همبستگی مرکب و پارامترهای $m=10$ و $X_0=7$ و $c=3$ و $a=4$ مولد اعداد تصادفی می شود:

$$X_{i+1} = (4X_i + 3) \pmod{10}$$

i	۰	۱	۲	۳	۴	۵	۶	۷
X_i	۷	۱	۷	۱	۷	۱	۷	۱
R_i		۱	۷	۱	۷	۱	۷	۱

دوره تناوب کوتاه دارد (r) و مولد خوبی احساب نمی شود.

مثال ۴-۵- با روش همبستگی مرکب و پارامترهای $m=10$ و $X_0=3$ و $c=7$ و $a=11$ مولد اعداد تصادفی عبارتست از:

$$X_{i+1} = (11X_i + 7) \pmod{10}$$

تمامی اعداد تولیدی بین ۰ و ۱۴ می‌باشند. از روشهای خوب و مناسب است که به آزمونهایی یکپارچه و استقلال پاسخ مثبت داده است.

مقادیر بهینه‌تر برای پارامترهای m ، a و c در این روش چگونگی باید باشد؟ می‌توان مولدهای دیگری نیز بر اساس معادله قبلی ایجاد کرد که عملکرد مولدهای همبستگی به نوع کامپیوتر انتخابی بستگی دارد.

فرد باشد X_0

$$m=2^r \quad (r \geq 2)$$

$$a=k8 \pm 3$$

عدد صحیح مثبت است. k

بربرد توالی تصادفی‌های تولیدی جداگیر و معادل 2^{r-2} است. برای تبدیل به یک مولد خوب تصادفی، بهتر است پارامترها را به شکل زیر انتخاب کنیم:

(۱) m یکی بیشتر از بزرگترین عدد صحیح داخل یک کلمه کامپیوتر (word) انتخاب گردد. اگر فرض طول کلمه ۳۲ بیت باشد و بارش‌ترین بیت، بیت علامت است پس بزرگترین عدد:

$$2^{31}-1=2,147,483,647$$

$$m=2,147,483,648$$

لذا مقدار m باید باشد:

$$m=(2^{\text{wordlength}}-1)$$

پس:

اگر m بزرگتر از کلمه ماشین باشد، سرریزی^(۱) رخ می‌دهد و فقط باقیمانده نامشخص داریم و در بعضی زبانها در این حالت (مانند فورتن) هیچ عملی صورت نمی‌گیرد.

(۲) X_0 نسبت به m باید اول باشد چون m توانی از ۲ است و عدد صحیح مثبت فرد برای X_0 متصور است. چون اگر X_0 زوج باشد هر X_i زوج است.

(۳) انتخاب صحیح a که باید نسبت به m اول باشد و a عدد فرد است. بهترین انتخاب a :

$$a = k8 \pm 3$$

می‌باشد که در محدوده 2^{32} محدود می‌شود و با تعداد پیمای کلمه ماشین است. برای مثال در ماشین با ۳۲ بیت طول کلمه $2^{16}+3=65539$ انتخاب می‌شود.

این روش از بهترین روشهای تولید تصادفی است و زبانهای مانند SIMSCRIPT و GPSS از آن استفاده می‌کنند.

$r=10$										
i	۰	۱	۲	۳	۴	۵	۶	۷	۸	۹
X_i	۳	۰	۷	۴	۱	۸	۵	۲	۹	۶
R_i	۰	۷	۴	۱	۸	۵	۲	۹	۶	۳
	۰	۷	۴	۱	۸	۵	۲	۹	۶	۳

دوره تناوب زیاد است لذا مولدی خوب به حساب می‌آید.

m محدوده اعداد تولیدی را تعیین می‌کند که این اعداد از ۰ تا $m-1$ می‌باشند و برای آن که دامنه اعداد تولیدی زیاد باشد m را بزرگتر انتخاب می‌کنیم. اگر بخواهیم دوره مولد همبستگی مرکب ماکزیمم شود، یعنی $(m=2^r)$ ، اولاً باید a نسبت به m اول باشد. دوماً اگر k یک عامل m است، باقیمانده تقسیم a بر k یک باشد. مثلاً اگر 8 بر 2 تقسیم شود با عمل شیفیت این امر امکان پذیر است و لذا از سرعت بالایی برخوردار می‌باشد. سوماً (طول کلمه حافظه) رابطه $c < (m+k)$ برقرار باشد.

ادعا شده است که ضریب همبستگی زوجهای صحیح متوالی X_0, X_1, \dots, X_{m-1} در محدوده

$$\left| \frac{1}{a} \cdot \left(\frac{6c}{am} \right) \left(1 - \frac{c}{m} \right) \right| \pm \frac{a}{m}$$

منصور می‌باشد و انتخاب $a=\sqrt{m}$ که بزرگترین مقدار ضریب همبستگی پیوسته مرتبه یک (بین X_0+1 و X_1) خواهد بود.

$$X_{i+1} \equiv (X_i \cdot a) \bmod m \quad i = 0, 1, 2, \dots$$

۴.۱.۱.۲- مولد همبستگی ضریبی

در فرمول کلی (۴.۱.۱) لذا

فرمول روش همبستگی ضریبی می‌باشد. همانطور که از فرمول مشاهده می‌گردد در فرمول همبستگی مرکب $b=0$ قرار گرفته است.

مثال ۴.۱- با استفاده از روش همبستگی ضریبی با پارامترهای $m=15$ و $X_0=3$ و $a=7$ مولد اعداد $X_{i+1}=(7 \cdot X_i) \bmod 15$

تصادفی عبارت است از:					
i	۰	۱	۲	۳	۴
X_i	۳	۹	۱۲	۹	۳
R_i	۹	۱۲	۹	۳	۰

۴-۱-۱-۳- مولد همبستگی تجمعی (۱)

در این روش با استفاده از دو عدد قبلی، عدد مورد نظر تولید می گردد (با شروع (X_1, X_0) که معادله آن چنین است:

$$X_{i+1} \equiv (X_i + X_{i-1}) \bmod m \quad i=1, 2, \dots$$

روش مناسبی نمی باشد زیرا دارای سرعت کمی است. به آزمونهای آماری پاسخ مثبتی نداشته است و رفتار شناخته شده ای ندارد.

مثال ۳-۷ با روش همبستگی تجمعی (۱) و پارامترهای $X_1=7$ و $X_0=3$ اعداد تصادفی تولید می شود:

$$\begin{array}{ll} X_0=3 & X_1=7 \\ X_1=7 & X_2=1 \\ X_2=1 & X_3=7 \\ X_3=7 & X_4=7 \\ X_4=7 & X_5=7 \end{array} \quad \begin{array}{ll} R_1=1 \\ R_2=7 \\ R_3=7 \\ R_4=7 \\ R_5=7 \end{array}$$

مثال ۳-۸ در روش همبستگی تجمعی (۱) با انتخاب خاصی به دنباله های شناخته شده ای دست می یابیم مانند حالت خاص آن یعنی $X_0=0$ و $X_1=1$ که موسوم به دنباله فیبوناچی می باشد:

$$\begin{array}{ll} X_0=0 & X_1=1 \\ X_1=1 & X_2=1 \\ X_2=1 & X_3=2 \\ X_3=2 & X_4=3 \\ X_4=3 & X_5=5 \end{array} \quad \begin{array}{ll} R_1=1 \\ R_2=1 \\ R_3=2 \\ R_4=3 \\ R_5=5 \end{array}$$

۴-۱-۱-۴- روش همبستگی تجمعی (۲)

با استفاده از یک سلسله n عددی و حرکت از طرفین اعداد، این رشته عمل تولید تصادفی را انجام می دهد. رفتار شناخته شده ای ندارد و برای عمل باید رشته X_1, X_2, \dots, X_n را داشته باشیم (با استفاده از جویباری از اعداد به تولید تصادفی می رسد).

با رشته X_1, X_2, \dots, X_n ، $i=1, 2, \dots, n$ ، $X_i \equiv (X_{i-1} + X_{n-i}) \bmod m$

چون عمل ضرب نداریم بسیار سریع تر عمل می کند ولی در کل رفتارش وابسته به رشته اعداد ابتدایی است.

بخش دوم / فصل چهارم - مولدهای اعداد تصادفی یکپارچه

مثال ۴-۹- با روش همبستگی تجمعی (۲) و $m=100$ و رشته $57, 34, 89, 82, 16$ اعداد تصادفی تولید می شود:

$$\begin{array}{l} X_6 = (X_5 + X_1) \bmod 100 = 73 \bmod 100 = 73 \\ X_7 = (X_6 + X_2) \bmod 100 = 107 \bmod 100 = 7 \\ X_8 = (X_7 + X_3) \bmod 100 = 96 \bmod 100 = 96 \end{array}$$

۴-۱-۲- مولد میان مربعی (۱)

این روش توسط وان نیومن و متر ایلیس ارائه گردید. این روش با یک عدد اولیه بنام هسته (X_0) آغاز می شود. یعنی عدد محوری X در هر مرحله (با شروع X_0) به توان ۲ می رسد، ارقام میانی آن انتخاب می شود و به صورت ممیز دار کوچکتر از صفر به عنوان عدد تصادفی آن مرحله ارائه می شود. در هر مرحله عدد مورد محاسبه به توان دو می رسد و ارقام میانی حاصل به عنوان مولد تصادفی نهایی انتخاب می شود.

مثال ۴-۱۰- گستره ۸ رقمی داریم و ۴ رقم وسط حاصل کار است با شروع:

$$\begin{array}{l} X_0 = 5497 \\ X_0^2 = (5497)^2 = 30,217,009 \Rightarrow X_1 = 2170 \\ R_1 = 0.2170 \\ X_1^2 = (2170)^2 = 04,708,900 \Rightarrow X_2 = 7089 \\ R_2 = 0.7089 \\ X_2^2 = (7089)^2 = 50,253,921 \Rightarrow X_3 = 2539 \\ R_3 = 0.2539 \end{array}$$

اولین تصادفی

تصادفی بعدی

تصادفی بعدی

پربود طولانی دارد.

اگر عدد محور m رقمی باشد، مربع آن (مجذور) $2n$ رقمی و اگر n زوج باشد با حذف $\frac{n}{2}$ رقم از ارقام طرفین (چپ و راست) مجذور حاصل ارقام میانی را می توان تعیین نمود.

مثال ۴-۱۱- با انتخاب نامناسب X_0 ممکن است صفرهای تولیدی انتشار یابند و عدد حاصل را به صفر میل دهد. تصادفی تولیدی با روش میان مربعی با عدد محوری ۵۱۹۷ شروع می شود:

مثال ۴-۱۴ - با روش میان ضربی اعداد تصادفی $X_0=7229$ و $X_0=2938$ حاصل می شود:

$$\begin{aligned} U_1 &= X_0' X_0 = (2938)(7229) = 21,238,802 \implies X_1 = 2388 \\ R_1 &= 0.2388 \\ U_2 &= X_0' X_1 = (7229)(2388) = 17,262,832 \implies X_2 = 2628 \\ R_2 &= 0.2628 \\ U_3 &= X_1' X_2 = (2388)(2628) = 6,275,664 \implies X_3 = 2736 \\ R_3 &= 0.2736 \end{aligned}$$

روش کدی است و پاسخ مثبت نسبت به آزمونهایی آماری نداده است. مشابه روش میان ضربی می باشد با این تفاوت که ناهمبازیها ممکن است به ازای دو عدد محوری X_1, X_{1+1} بروز کند.

۴-۱۴- روش ثابت میان ضربی (مضرب ثابت)

این روش تفاوت کمی با روش میان ضربی دارد و عدد تصادفی از حاصلضرب یک عدد در یک مقدار ثابت و سپس انتخاب ارقام میانی بدست می آید. یعنی یکی از اعداد محوری، ثابت فرض می شود. با شروع از X_0 و ثابت k با ارقام یکسان:

مثال ۴-۱۵ - از روش مضرب ثابت با $X_0=7223$ ، $X_0=3987$ ، $k=$ اعداد تصادفی تولید می شود:

$$\begin{aligned} X_1 &= kX_0 = 28798101 & X_1 &= 7981 & R_1 &= 0.7981 \\ X_2 &= kX_1 = 31820247 & X_2 &= 8202 & R_2 &= 0.8202 \end{aligned}$$

روش کدی است و به آزمونهایی آماری پاسخ مثبت نداده است. همان ایرادهایی روش میان ضربی را دارد و عملکرد آن تا حدی به ثابت وابستگی دارد.

۴-۲- آزمون اعداد تصادفی

خواص سلسله اعداد تصادفی تولیدی یعنی یکنواختی و استقلال، باید مورد آزمایش قرار گیرد و صحت آن بررسی شود. بررسی گردد که اعداد به صورت یکنواخت توزیع شده اند و تکرار اعداد از استقلال برخوردار هستند و پیوستگی خاصی مشاهده نمی شود. وضعیت یکنواختی و استقلال را باید دارا باشند چون امکان دارد که اعداد یکنواخت توزیع شوند ولی از استقلال خارج باشند. در آزمونها گاهی بحث یکنواختی مورد توجه قرار می گیرد و گاه بحث استقلال و یا هر دو کنار هم بررسی می گردند. از این جهت هر کدام از روشهای آزمون ممکن است در یکی از مقوله های استقلال و یکنواختی کارا باشد. در این جا انواع آزمونها و محور بررسی استقلال یا یکنواختی آنها ارائه می شود.

$$\begin{aligned} X_0 &= 5197 \\ X_0^2 &= (5197)^2 = 27,008,809 \implies X_1 = 0088 \\ R_1 &= 0.0088 \\ X_1^2 &= 00,007,744 \implies X_2 = 0077 \\ R_2 &= 0.0077 \\ X_2^2 &= 00,005,929 \implies X_3 = 0059 \\ R_3 &= 0.0059 \end{aligned}$$

مثال ۴-۱۴ - ممکن است با انتخاب نامناسب X_0 یا تولید میانی نامناسب X_1 هیچ تولیدی نباشد. در روش میانی مربعی در صورت تولید عدد محوری مانند عملیات تولید اعداد تصادفی متوقف می شود:

$$\begin{aligned} X_i &= 0 & X_{i+1} &= 0 & R_i &= 0 \\ (X_{i+1}) &= 0 & X_{i+2} &= 0 & R_{i+1} &= 0 \end{aligned}$$

مثال ۴-۱۳ - تولید اعداد میانی که ممکن است در یک حلقه تکرار قرار گیرند و مولد یک عدد تکراری تولید کند. چنانچه در مرحله ای از مراحل عدد محوری مانند 0.05 تولید بشود، تصادفی تولیدی متوقف می شود:

$$\begin{aligned} X_i &= 6500 \\ X_i^2 &= (6500)^2 = 42,250,000 \implies X_{i+1} = 2500 \\ R_i &= 0.2500 \\ (X_{i+1})^2 &= 06,250,000 \implies X_{i+2} = 2500 \\ R_{i+1} &= 0.2500 \end{aligned}$$

روش میانی مربعی در کل روش کدی است و از آزمونهایی آماری موفق نبوده است. بیشتر جنبه مطالعه دارد زیرا نمی توان قواعد ساده برای عملکرد مطلوب الگوریتم جهت 0.05 یافت و مضافاً با مشکلات ارائه شده، احتمال بروز مشکل در خروجی تولیدی R_i بسیار زیاد است.

۴-۱۳- روش میان ضربی

مانند روش میان مربعی است که از حاصلضرب دو عدد متوالی با ارقام مساوی، ارقام میانی انتخاب می گردد و تصادفی نهایی تولید می شود.

$$U_i = X_{i-1}' X_i$$

که شروع با دو عدد آغازین X_0 و X_1 .

آزمون فراوانی (Frequency)

آزمون یکنواختی توزیع سلسله اعداد تصادفی را انجام می‌دهد. یعنی تکرار آنها بیش از آنچه انتظار می‌رود رخ ندهد. مثلاً چنانچه توالی ۵۰۰۰ عدد سه رقمی از ۹۹۹ تا ۹۹۹ داریم، انتظار می‌رود که حدود ۵۰۰ عدد در فاصله ۰۰ تا ۹۹، ۵۰۰ عدد در فاصله ۱۰۰ تا ۱۹۹ و مانند آن رخ دهد. در واقع انتظار رخ داد ۵۰۰ عدد در هر دسته ۱۰۰ تایی داریم و در غیر اینصورت یکنواختی ندارد. از روشهای مناسب آن آزمون خی دو (کای دو) و آزمون کولموگروف اسمیرنوف می‌باشد. روش کای دو مناسب نمونه‌های بیش از ۵۰ و روش کولموگروف اسمیرنوف مناسب نمونه‌های کمتر از ۵۰ می‌باشد و جداول آماری خاص خود را دارند.

آزمون امتداد (RUN)

آزمون استقلال سلسله اعداد تصادفی را انجام می‌دهد. امتداد دنباله به طرف بالا (افزایش) یا پائین (کاهش) بوسیله مقایسه مقادیر واقعی با مقادیر انتظاری است (حرکت زیر یا بالای میانه). از جدول آماری کای دو استفاده می‌کند.

آزمون همبستگی (Autocorrelation)

آزمون استقلال سلسله اعداد تصادفی را انجام می‌دهد. همبستگی بین اعداد تولیدی با نمونه‌ها مقایسه می‌شود و انتظار همبستگی صفر دارد. از جدول آماری کای دو استفاده می‌کند.

آزمون فاصله - شتاف (Gap)

آزمون استقلال سلسله اعداد تصادفی را انجام می‌دهد. این امر را با شمارش تعداد ارقامی که بین تکرارهای یک رقم خاص ظاهر می‌شود، صورت می‌دهد. از جدول آماری کولموگروف اسمیرنوف برای مقایسه با تعداد فواصل متظره استفاده می‌کند.

آزمون پوکر (Poker)

آزمون استقلال سلسله اعداد تصادفی را انجام می‌دهد. اعداد گروه‌بندی شده را با هم مانند دسته پوکر بحث می‌کند و سپس دسته‌های یافت شده با دسته‌های انتظاری مقایسه می‌گردند. از جدول آماری کای دو استفاده می‌کند.

۴-۳- یکنواختی و استقلال

یکنواختی و استقلال را می‌توان بصورت زیر تعریف کرد:

الف - یکنواختی و عدم یکنواختی:

$$H_0: R_i \sim u(0,1) \quad \text{یکنواختی}$$

$$H_1: R_i \neq u(0,1) \quad \text{عدم یکنواختی}$$

ب - استقلال و عدم استقلال:

$$H_0: R_i \sim \text{استقلال} \quad \text{استقلال}$$

$$H_1: R_i \neq \text{استقلال} \quad \text{وابستگی}$$

لذا با استفاده از استقلال یکنواختی یک ضریب α (احتمال) تعریف می‌کنیم:

$$\alpha = P\left(\frac{H_0}{H_0} \mid \text{صحت}\right)$$

که معمولاً مقادیر ۰.۰۱ یا ۰.۰۵ را به α نسبت می‌دهیم. چنانچه آزمونهای مختلفی روی یک سری اعداد انجام شود، احتمال رد شیدیفرونی، حداقل یک آزمون بصورت تصادفی افزایش می‌یابد. $\alpha=0.05$ یعنی چنانچه ۱۰۰ سری آزمون انجام شود احتمال می‌رود که از ۵ آزمون رد شود لذا می‌توان با ضریب اطمینان $1-\alpha=0.95$ به نتیجه عمل مطمئن بود. ولی نکته مهمی وجود دارد که خروجی باید بررسی و تأیید گردد یعنی اگر یک مولد تصادفی به تمامی آزمونها پاسخ مثبت داده باشد باز دلیل تصادفی بودن نیست چون ممکن است الگوی خاصی آشکار شود که قبلاً بروز نکرده است.

۴-۳-۱ آزمون فراوانی

جهت آزمون یکنواختی دو روش متفاوت وجود دارد که درجه توافق بین شکیک نمونه‌ها و انتظار ما را برآورده می‌کند. دارای دو روش آزمون یعنی آزمون کولموگروف اسمیرنوف و آزمون کای دو (خی دو) است.

۴-۳-۱-۱ روش کولموگروف-اسمیرنوف

این روش برای آزمون یکنواختی تصادفی‌های تولیدی استفاده می‌شود و مناسب نمونه‌های کم است ($N \leq 50$).

مثال ۴-۱۶ - فرض ۵ عدد تصادفی تولید شده‌اند:

0.44, 0.81, 0.14, 0.05, 0.93

با استفاده از مراحل تست کولموگروف اسمیرنوف با سطح $\alpha=0.05$ در رابطه با نمونه‌ها بررسی یکپارختی آن را آزمون نمایید.

گام ۱ و ۲ - مرتب کردن کوچک به بزرگ تصادفی‌ها و محاسبات یافتن D :

$R_{(0)}$	0.05	0.14	0.44	0.81	0.93
i/N	0.20	0.40	0.60	0.80	1.00
$i/N - R_{(0)}$	0.15	0.26	0.16	—	0.07
$R_{(0)} - (i-1)/N$	—	0.04	0.21	0.13	—

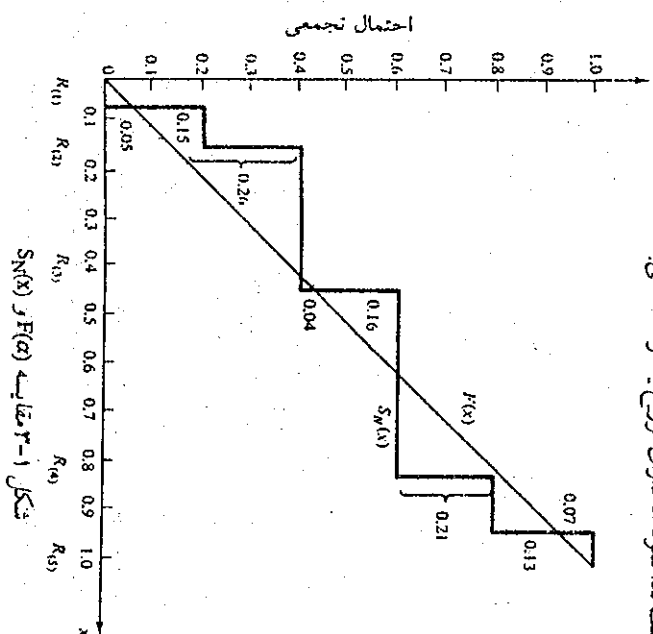
$$D = \max\{0.21, 0.26\} = 0.26$$

گام ۳ - محاسبه

$$D\alpha = 0.565 \text{ آزمون آماری از جدول آزمون}$$

گام ۴ - یافتن $D\alpha$ از جدول آماری توزیع یکپارخت می‌باشد.

گام ۵ - چون $D \leq D\alpha$ است لذا نمونه‌ها دارای توزیع یکپارخت می‌باشند.



$$F(x) = x$$

$$0 \leq x \leq 1$$

چنانچه تعداد نمونه‌های تصادفی تولیدی N باشد (نمونه‌ها):

$$R_1, R_2, \dots, R_N$$

$$cdf \quad SN(\alpha) \text{ عبارت است از:}$$

$$SN(x) = \frac{\text{تعداد اعداد } R_1, R_2, \dots, R_N \text{ که کوچکتر یا مساوی } x \text{ هستند}}{N} \quad (x \text{ نمونه})$$

چنانچه N بزرگتر شود $SN(x)$ که تقریب بهتری از $F(x)$ را می‌دهد. این آزمون براساس بیشترین مقدار D استوار است (فاصله بین $SN(x)$ و $F(x)$):

$$D = \max |F(x) - SN(x)|$$

(D تابعی از N است)

در واقع مقایسه cdf و $F(\alpha)$ توزیع یکپارخت با cdf از $SN(x)$ نمونه‌های مشاهده شده از N نمونه

است (شکل ۴-۱).

روش:

گام ۱ - مرتب کردن اعداد تصادفی تولیدی (نمونه‌ها) از کوچکتر به بزرگتر (فرض $R(1)$ کوچکترین

مقدار را دارد).

گام ۲ - محاسبه D^+ و D^-

$$D^+ = \max_{1 \leq i \leq N} \left\{ \frac{i}{N} - R_{(i)} \right\} \quad 1 \leq i \leq N$$

$$D^- = \max_{1 \leq i \leq N} \left\{ R_{(i)} - \frac{i-1}{N} \right\}$$

$$D = \max(D^+, D^-)$$

گام ۳ - محاسبه ماکزیمم D^+ و D^- :

گام ۴ - تعیین مقادیر بحرانی $D\alpha$ از جدول آماری آزمون در سطح α و نمونه‌های N

گام ۵ - اگر مقدار آماری D (نمونه‌ها) از مقدار بحرانی $D\alpha$ بزرگتر باشد، یکپارختی نمونه‌ها تأیید

نمی‌شود (یکپارختی نمونه‌ها: $D \leq D\alpha$)

صحت بررسی: اگر $D \leq D\alpha$ باشد یعنی اختلافی بین توزیع صحیح نمونه‌ها (R_1, R_2, \dots, R_N) و توزیع یکپارخت وجود ندارد و لذا یکپارخت هستند.

۴.۳.۱.۲ - آزمون کای دو (خی دو)

آزمون آماری خوبی برای تعیین یکنواختی اعداد و ارتباط با مشاهدات و انتظار مشاهده می‌باشد. برای نمونه‌های بیشتر از ۵۰ عدد استفاده می‌گردد ($N \geq 50$). اساس این روش بر تقسیم‌بندی دسته‌های مشاهدات استوار است. فراوانی اعداد تصادفی تولیدی در هر دسته را با فراوانی انتظار مشاهده مقایسه و نزدیکی آنها را می‌سنجد. دسته‌ها هیچ‌گونه رویهم افتادگی نباید داشته باشند (تعداد دسته‌ها باید ۳ یا بیشتر باشد).

سپس کای دو را بصورت زیر می‌یابیم:

$$\chi^2 = \sum_{i=1}^k \frac{(O_i - E_i)^2}{E_i}$$

که مجموع اختلاف مشاهدات و رخ داده‌ها در دسته‌هاست. هر چه مشاهدات و انتظارات از یکدیگر فاصله بگیرند، مقدار $(O_i - E_i)^2$ بیشتر می‌شود و لذا χ^2 افزایش می‌یابد و چنانچه این دو یکسان باشند مقدار χ^2 صفر می‌شود. روال کار چنین است:

- نمونه‌ها به n دسته تقسیم می‌گردند که باید $n \geq 3$ باشد.
- O_i تعداد مشاهدات در i امین دسته است.
- E_i تعداد انتظار مشاهده در i امین دسته است.
- $E_i = \frac{N}{n}$ که N تعداد کل نمونه‌های مشاهده شده است (انتظار مشاهده یکسان).
- نیاز به جدول کای دو می‌باشد که مقدار بحرانی را از آن بیابیم تا با χ^2 حاصل مقایسه گردد.
- چنانچه χ^2 مشاهده شده، از مقدار بحرانی از جدول کوچکتر باشد یکنواختی نمونه‌ها صحیح است.

یافتن مقدار بحرانی از جدول براساس درجه آزادی ($V=n-1$) و پارامتر α می‌باشد. می‌توان گفت که توزیع نمونه‌های χ^2 تقریباً توزیع کای دو با $(n-1)$ درجه آزادی است. چنانچه $\chi^2 \leq \chi^2_{v,p}$ باشد، از آزمون یکنواختی تأیید می‌شود.

مثال ۱۷ - ۴ - استفاده از آزمون χ^2 با $\alpha=0.05$ برای آزمون توزیع یکنواخت جهت ۵۰۰۰ نمونه بصورت زیر انجام می‌گیرد:

- تقسیم‌بندی اعداد به دسته‌های ۱۰۰ تایی و با توجه به محدوده اعداد که از ۹۹ تا ۹۹۹ رخ داده‌اند.

بخش دوم / فصل چهارم - مولدهای اعداد تصادفی یکنواخت

- کل دسته‌های موجود $n=10$ می‌باشد.
- کل مشاهدات $N=5000$ است.
- انتظار مشاهده در هر دسته $E_i = \frac{5000}{10} = 500$ است.
- مقدار بحرانی $\chi^2_{v,p}$ می‌باشد.

جدول زیر خلاصه عملیات است:

i	محدوده	O_i	E_i	$(O_i - E_i)^2$	$\frac{(O_i - E_i)^2}{E_i}$
1	000-099	468	500	1024	2.048
2	100-199	519	500	361	.722
3	200-299	480	500	400	.800
4	300-399	495	500	25	.050
5	400-499	508	500	64	.128
6	500-599	426	500	5476	10.952
7	600-699	497	500	9	.018
8	700-799	515	500	225	.450
9	800-899	463	500	1369	2.738
10	900-999	529	500	841	1.682

با یافتن $\chi^2 = 19.588$ نیاز به اخذ مقدار بحرانی از جدول داریم.

- درجه آزادی $v=n-1=10-1=9$
- یافتن $p=1-\alpha=1-0.05=0.95$
- $\chi^2_{v,p} = \chi^2_{9,0.95} = 16.9$
- چون $\chi^2 = 19.588 > \chi^2_{v,p} = 16.9$ مقدار بحرانی بزرگتر است لذا نمونه‌ها دارای یکنواختی نمی‌باشند و تأیید نمی‌شوند:

$$\chi^2 > \chi^2_{v,p}$$

۴.۳.۲ - آزمون امتداد (RUN Test)

آزمون استقلال نمونه‌ها می‌باشد که اساس آن بر امتداد اعداد استوار می‌گردد. افزایش یا کاهش مشاهده‌ها در سری تولیدی مورد بررسی قرار می‌گیرد و به عنوان توانی وقایع اتفاقی آزمون را سپری می‌کند. مقداری را از آزمون می‌یابیم و با مقدار بحرانی یافته شده از جدول کای دو مقایسه می‌کنیم. قبل از شرح یک امتداد (RUN)، نگاهی به یک سری در یافتن منطق این روش به ماکمک می‌کند.

تعداد امتدادها ۱ است و صعودی می باشد. $RUNS=1$

مثال ۱-۴۲۱- اگر توالی رشته زیر را در نظر بگیرد امتداد بیشتر در آن مشاهده می شود:

0.08 0.93 0.15 0.96 0.26 0.84 0.28 0.79 0.36 0.57

+ - + - + - + - +

توالی علامت حاصل عبارتست از:

$RUNS=9$

تعداد امتدادها ۹ است و یک در میان حرکت می کند

چنانچه N تعداد اعداد تولیدی باشد، حداکثر و حداقل امتدادها بصورت زیر است:

$1 \leq RUNS \leq N-1$

با توجه به مثالهای ۴۲۰ و ۴۲۱ می بینیم که $N=10$ پس $1 \leq RUNS \leq 9$ است.

تعریف روابط روش آزمون امتدادها با یک توالی n تایی اعداد تصادفی بصورت زیر خواهد بود.

تعداد کل امتدادها $a = RUNS$

میانگین $\mu_a = \frac{2N-1}{3}$

واریانس $\sigma_a^2 = \frac{16N-29}{90}$

برای $N > 20$ توزیع a با توزیع نرمال $N(\mu_a, \sigma_a^2)$ تقریب زده می شود. این تقریب برای آزمون استقلال اعداد تولیدی است. و لذا پارامتر Z_0 به اینصورت تعریف می گردد.

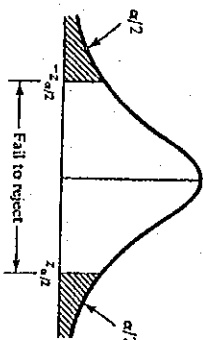
پارامتر اصلی:

$$Z_0 = \frac{a - \mu_a}{\sigma_a}$$

با جایگذاری

$$Z_0 = \frac{a - \left[\frac{(2N-1)}{3} \right]}{\sqrt{\frac{(16N-29)}{90}}}$$

با یافتن $Z_{\alpha/2}$ از جدول کای دو چنانچه $Z_{\alpha/2} \leq Z_0 \leq Z_{\alpha/2}$ باشد، آزمون استقلال با موفقیت رد شود و در ورنه رد می شود.



H T T H H T T H T

مثال ۴۱۸

سری از H (افزایش) و T (کاهش) داریم. در واقع می توانیم بگوئیم که این سلسله دارای وضعیت

زیر است:

۱- رانده H به T.

۲- رانده T به H.

۳- بدون رانده T به T یا H به H.

حال یک امتداد (RUN) را تعریف می کنیم: امتداد (RUN) یعنی یک توالی از وقایع مشابه و دنبال شده با یک رانده مختلف است. طول امتداد (RUN) یعنی تعداد وقایع که در یک سری رخ داده است.

در مثال ۴۱۸، امتداد داریم: $RUNS=6$.

طول امتدادها: اولی با طول ۱ (H)، دومی با طول ۲ (TT)، سومی با طول ۲ (HT)، چهارمی با طول ۳ (TTT)، پنجمی با طول ۱ (H) و ششمی با طول ۱ (T) می باشد.

می توان گفت افزایش داریم وقتی که از عدد کم به عدد زیاد می رویم ($H \rightarrow +$) و می توان گفت کاهش داریم وقتی که از عدد زیاد به عدد کم می رویم ($T \rightarrow -$).

مثال ۴۱۹- توالی زیر که از ۱۵ عدد تصادفی تشکیل شده است در نظر بگیرد:

0.87 -0.15 +0.23 +0.45 +0.69 -0.32 -0.30 -0.19 +0.24
-0.18 +0.65 +0.82 +0.93 -0.22 +0.81

توالی ۱۴ علامت حاصل عبارتست از:

+ - + + + - - - + + + -

تعداد امتدادها ۷ می باشد. $RUNS=7$

مثال ۴۲۰- در توالی رشته زیر تنها یک امتداد صعودی داریم:

0.08 +0.18 +0.23 +0.36 +0.42 +0.55 +0.63 +0.72 +0.89 +0.91

توالی علامت حاصل عبارتست از:

+ + + + + + + +

برای مقادیر زیر M توزیع P_{im} یا \hat{P}_{im} تقریباً نرمال است اگر سری

$$R_j, R_{i+m}, R_{i+2m}, \dots, R_{i+(M+1)m}$$

همبستگی نداشته باشند.

$$Z_0 = \frac{\hat{P}_{im}}{\sigma_{P_m}}$$

$$\hat{P}_{im} = \frac{1}{M+1} \left[\sum_{k=0}^M R_{i+km} R_{i+(k+1)m} \right] - 0.25$$

$$\sigma_{P_m} = \frac{\sqrt{13M+7}}{12(M+1)}$$

در صورتی که $-Z_{\alpha/2} \leq Z_0 \leq Z_{\alpha/2}$ (با سطح α) باشد، استقلال دارد. جدول آماری کای دو بدست می‌آید.

مثال ۴-۳-۴ - آزمون فاصله

آزمون استقلال سلسله اعداد تصادفی می‌باشد. اعداد قبل از تبدیل به مقادیر بین ۰ و ۱ مورد بررسی قرار می‌گیرند و در ارتباط با رخ داد اولین رقم و فاصله بین آنها تا همان رقم بررسی می‌شوند. یک فاصله به طول X بین تکرار رقم وجود دارد. با یک مثال این روش را شرح می‌دهیم.

مثال ۴-۳-۳ - بررسی طول فاصله با رقم ۳ در رشته تصادفی زیر است:

4, 1, 3, 5, 1, 7, 2, 8, 2, 0, 7, 9, 1, 3, 5, 2, 7, 9, 4, 1, 6, 3,
3, 9, 6, 3, 4, 8, 2, 3, 1, 9, 4, 4, 6, 8, 4, 1, 3, 8, 9, 5, 5, 7,
3, 9, 5, 9, 8, 5, 3, 2, 2, 3, 7, 4, 7, 0, 3, 6, 3, 5, 9, 9, 5, 5,
5, 0, 4, 6, 8, 0, 4, 7, 0, 3, 3, 0, 9, 5, 7, 9, 5, 1, 6, 6, 3, 8,
8, 8, 9, 2, 9, 1, 8, 5, 4, 4, 5, 0, 2, 3, 9, 7, 1, 2, 0, 3, 6, 3

زیر رقمهای ۳ خط‌کشی شده است. ۱۸ رقم ۳ داریم که فقط در ۱۷ فاصله رخ داده است. اولین فاصله به طول ۱۰، دومی به طول ۷ و مانند آن است. احتمال اولین فاصله به صورت زیر است:

شیه‌سازی

مثال ۴-۳-۲ - به ازای $\alpha=0.05$ تعیین کنید که آیا توالی ۴۰ عدد تصادفی زیر دارای خاصیت استقلال

است؟

| | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|
| 0.41 | 0.68 | 0.89 | 0.94 | 0.74 | 0.91 | 0.55 | 0.62 | 0.36 | 0.27 |
| 0.19 | 0.72 | 0.75 | 0.08 | 0.54 | 0.02 | 0.01 | 0.36 | 0.16 | 0.28 |
| 0.18 | 0.01 | 0.95 | 0.69 | 0.18 | 0.47 | 0.23 | 0.32 | 0.82 | 0.53 |
| 0.31 | 0.42 | 0.73 | 0.04 | 0.83 | 0.45 | 0.13 | 0.57 | 0.63 | 0.29 |

+ + + + + - - - - - + + + + + - - - - - + + + + + - - - - -

$$RUN: + = 13$$

$$RUN: - = 13 \quad a = 13 + 13 = 26 \quad N = 40$$

$$RUNS = a = 26$$

$$N = 40, a = 26 \quad \mu a = \frac{2(40)-1}{3} = 26.33$$

$$\sigma_a^2 = \frac{16(40) - 29}{90} = 6.79$$

$$Z_0 = \frac{26 - 26.33}{\sqrt{6.79}} = -0.13$$

$$-Z_{\alpha/2} \leq Z_0 \leq Z_{\alpha/2}$$

آزمون استقلال تأیید می‌گردد:

۴-۳-۳ - آزمون همبستگی

آزمون استقلال را برای نمونه‌ها انجام می‌دهد. محاسبه همبستگی بین هر m عدد با شروع از اولین عدد می‌باشد لذا همبستگی P_{im} بین اعداد زیر است:

$$R_j, R_{i+m}, R_{i+2m}, \dots, R_{i+(M+1)m}$$

که M بزرگترین عددی است که در معادله زیر صدق می‌کند:

$$i + (M+1)m \leq N$$

N تعداد کل اعداد در یک سری است (پس توالی با $M+2$ عدد آزمون می‌شود).

از آنجایی که همبستگی غیر صفر داریم:

$$H_0: \rho_{im} = 0$$

$$H_1: \rho_{im} \neq 0$$

در هر موقعیت، یک جفت ارقام مثل هم، در اعداد تولیدی ظاهر می شود. در اعداد ۳ رقمی فقط ۳ امکان وجود دارد:

۱- ارقام می توانند متفاوت باشند.

۲- ارقام می توانند یکسان باشند.

۳- یک جفت مانند هم باشند.

احتمال هر کدام می شود:

$$\begin{aligned} P(\text{دومین رقم متفاوت است از اولین رقم}) &= P(\text{سه رقم مختلف}) \\ &= P(\text{رقم سوم با دو رقم اول و دوم متفاوت باشد}) \\ &= (0.9)(0.8) = 0.72 \end{aligned}$$

$$\begin{aligned} P(\text{دومین رقم مثل اولین رقم است}) &= P(\text{سه رقم یکسان}) \\ &= P(\text{سومین رقم مثل اولین رقم است}) \\ &= (0.1)(0.1) = 0.01 \end{aligned}$$

$$P(\text{یک جفت یکسان}) = 1 - 0.72 - 0.01 = 0.27$$

آخرین نتیجه بصورت زیر است:

$$P(\text{یک جفت یکسان}) = \left(\frac{3}{2}\right)(0.1)(0.9) = 0.27$$

مثال ۴-۳: از بین ۱۰۰۰ عدد سه رقمی مورد مطالعه، ۶۸۰ عدد با سه رقم مختلف، ۲۸۹ عدد یک جفت رقم یکسان و ۳۱ عدد سه رقم یکسان دارند. بررسی کنید که آیا با آزمون یوکر استقلال دارند یا نه؟
سطح $\alpha = 0.05$

خلاصه:

| ترکیب | مشاهده O_i | انتظار E_i | $\frac{(O_i - E_i)^2}{E_i}$ |
|--------------|--------------|--------------|-----------------------------|
| سه رقم مختلف | 680 | 720 | 2.22 |
| سه رقم مشابه | 31 | 10 | 44.10 |
| دو رقم یکسان | 289 | 270 | 1.33 |
| | 1000 | 1000 | 47.65 |

$$\chi^2 = 47.65$$

از کای دو $\chi^2_{0.05,2} = 5.99$ است. چون χ^2 از این مقدار بزرگتر است لذا استقلال ندارد.

$$\begin{aligned} P(10) &= \overbrace{P(n03) \cdots P(n03)}^{10} \\ &= (0.9)^{10}(0.1) \end{aligned}$$

احتمال آن که یک رقم ۳ نباشد، 0.9 و احتمال آن که رقم ۳ باشد، 0.1 است.

$$P(x \text{ رقم بدون ۳ به دنبال رقم } x) = (0.9)^x(0.1), \quad x = 0, 1, 2, \dots$$

از نظر تئوری توزیع تکرار تصادفی رقم مورد نظر عبارتست از:

$$F(x) = 0.1 \sum_{i=0}^x (0.9)^i = 1 - 0.9^{x+1}$$

برای مقدار بحرانی از جدول آماری کو لمرگروف اسمیرنوف استفاده می شود.

مراحل آزمون:

۱- تعریف $hat{F}$ برای توزیع آماری

$$F(x) = 1 - 0.9^{x+1}$$

۲- دسته بندی نمونه های مشاهده شده از فاصله در توزیع با دسته های یکسان.

۳- یافتن D ماکزیمم فاصله بین $F(x)$ و $S_N(x)$ مانند:

$$D = \max |F(x) - S_N(x)|$$

۴- تعیین مقدار بحرانی $D_{\alpha, N}$ از جدول کو لمرگروف - اسمیرنوف.

۵- اگر مقدار D محاسبه شده از $D_{\alpha, N}$ بیشتر باشد، استقلال منتهی است.

۴-۳-۵ آزمون یوکر:

آزمون یوکر نیز جهت بررسی استقلال به کار می رود و اساس آن بر تکرار ارقام است که در یک سری اعداد تکرار می شوند. مثال زیر تکرار بی قاعده است:

مثال ۴-۳-۵

$$0.255, 0.577, 0.331, 0.414, 0.828, 0.909, 0.303, 0.001, \dots$$

فصل پنجم

مولدهای اعداد تصادفی غیر یکنواخت

۵- مولدهای اعداد تصادفی غیر یکنواخت

در فصل قبل چگونگی تولید اعداد تصادفی با توزیع یکنواخت $U[0,1]$ ارائه شد و اعداد تصادفی در فاصله $[A,B]$ تولید گردید که فرمول آن را بصورت زیر ارائه کردیم:

$$Y_i = A + (B - A) U_i$$

U_i عدد تصادفی یکنواخت در فاصله $[0,1]$ است که در این فرمول قرار می گیرد و اعداد تصادفی یکنواخت در فاصله $[A,B]$ را تولید می نماید.

در این فصل با تولید اعداد تصادفی یکنواخت و استفاده از آن و در پی یافتن روش تولید اعداد تصادفی با توزیع های مورد نظر و غیر یکسان هستیم، مانند توزیع نرمال، نماتی، گاما و غیره. برای تولید بسیاری از اعداد تصادفی در شبیه سازی به مولدهایی با توزیع های مشخص و معینی نیاز داریم، روشهای مختلف تولید وجود دارند که به بررسی و ارائه جزئیات روشها می پردازیم. برای این منظور چهار روش اساسی موجود است:

۱- روش تبدیل معکوس ^(۱)

۲- روش حذفی ^(۲)

۳- روش بکارگیری خواص ویژه

۴- روش ترکیبی

البته روش ترکیبی آمیخته ای از مجموع روشهای سه گانه فوق است.

۵.۱- روش تبدیل معکوس

متغیر تصادفی X دارای تابع توزیع تجمعی $F(x)$ است و اگر بنخواهیم n نمونه X_1, X_2, \dots, X_n از توزیع تجمعی فوق پیروی کند. احتمال این که نمونه x در فاصله $(X_1 \text{ و } X_2)$ باشد عبارتست از:

مثال ۵-۱- تابع توزیع یکدخت (Uniform)

فرض کنید متغیر تصادفی x در فاصله دلخواه $[a, b]$ موجود پیوسته و دارای توزیع آماری یکدخت باشد. پس داریم:

$$f(x) = \begin{cases} \frac{1}{b-a} & a \leq x \leq b \\ 0 & \text{سایر} \end{cases}$$

تابع چگالی احتمال آن را داریم:

راه حل:

$$F(x) = \int_{-\infty}^x f(t) dt$$

گام ۱- تابع $F(x)$ را می‌یابیم:

$$F(x) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ 1 & x > b \end{cases}$$

گام ۲- تابع $F(x)$ را برابر u قرار می‌دهیم:

$$\begin{aligned} u &= \frac{(x-a)}{(b-a)} \rightarrow U(b-a) = (x-a) \\ x &= a + u(b-a) \end{aligned}$$

گام ۳- x را از گام ۲ می‌یابیم:

گام ۴- فرمول کلی می‌شود:

$$X_i = a + U_i(b-a)$$

مثال ۵-۲- اگر بخواهیم مولدی برای تولید سرعت یک هواپیما در محدوده یکدخت $[200, 300]$ ماخ داشته باشیم. با فرمول بدست آمده از روش تبدیل معکوس:

$$X_i = 200 + U_i(300-200) = 200 + 100U_i$$

اگر $U_1 = 0.5$ $U_2 = 0.4$, ...

$$X_1 = 200 + 100 \times 0.5 = 250$$

$$X_2 = 200 + 100 \times 0.4 = 240$$

عدد تصادفی نهایی می‌شود:

یعنی اعداد تصادفی نهایی عبارتند از

$$R_1 = 250, R_2 = 240, \dots$$

مثال ۵-۳- تابع توزیع نمایی (exponential) متغیر تصادفی x دارای توزیع آماری نمایی است. تابع چگالی احتمال عبارتست از:

برای تمامی آنهایی که $X_1 \leq X_2$ ، $F(X_2) - F(X_1)$

و از آنجا که $F(x)$ پیوسته است، تمامی مقادیر بین ۰ و ۱ را در برمی‌گیرد و بنابراین برای هر عدد تصادفی u ، $0 \leq u \leq 1$ است و یک X واحد وجود دارد که $u = F(X_u)$ رابطه تابع $F(x)$ و چگالی احتمال عبارتست

$$F(x) = \int_{-\infty}^x f(t) dt$$

از:

و $0 \leq F(x)$ می‌باشد و همچنین می‌دانیم که $0 \leq F(x) \leq 1$ همیشه برقرار است.

اگر فرض کنیم که:

$$F(X_u) = u$$

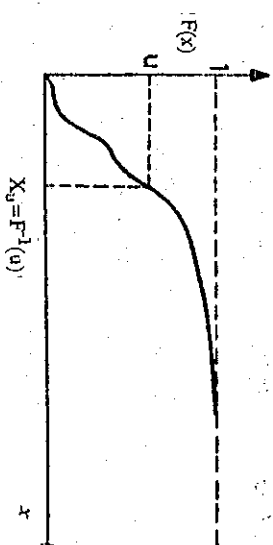
$$0 \leq u \leq 1$$

برای هر عدد تصادفی

$$X_u = F^{-1}(u)$$

پس معکوس تابع می‌شود:

$F^{-1}(u)$ را تابع معکوس گویند که شکل ۵-۱ این امر را نشان می‌دهد.



$$X_u = F^{-1}(u)$$

شکل ۵-۱

اگر n عدد تصادفی یکدخت تولید کنیم: U_1, U_2, \dots, U_n که در محدوده $[0, 1]$ باشند، تبدیل معکوس آنها

$$F^{-1}(U_1), F^{-1}(U_2), \dots, F^{-1}(U_n)$$

می‌شود:

بطور کلی می‌توان روش تبدیل معکوس را بصورت زیر بیان داشت:

$$X_i = F^{-1}(U_i)$$

$$i = 1, 2, \dots, n$$

که U_i یک عدد تصادفی یکدخت است و اگر نرم زیر برای آن برقرار می‌باشد:

الف- عدد تصادفی یکدخت U_i در محدوده ۰ و ۱ را تولید کنید.

ب- تابع معکوس $X_i = F^{-1}(U_i)$ را بیابید.

ج- یک عدد تصادفی (R_i) از توزیع $F(x)$ است.

بدین است که به منظور قابل تعریف بودن تبدیل معکوس یک تابع توزیع $F(x)$ باید وجود داشته

باشد و ضمناً این روش در مورد پیوسته و گسسته قابل اعمال خواهد بود.

$$f(x) = \begin{cases} \frac{\beta}{\alpha^\beta} x^{\beta-1} e^{-(x/\alpha)^\beta} & x \geq 0 \\ 0 & \text{سایر} \end{cases} \quad \begin{matrix} \alpha > 0 \\ \beta > 0 \end{matrix}$$

راه حل:

$$F(x) = 1 - e^{-(x/\alpha)^\beta}, \quad x \geq 0.$$

$$U = R$$

$$F(X) = 1 - e^{-(X/\alpha)^\beta} = R.$$

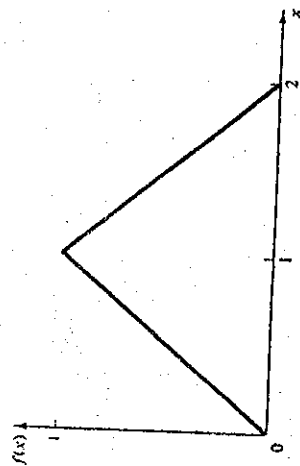
$$X = \alpha[-\ln(1-R)]^{1/\beta}$$

$$X_i = \alpha[-L_n(1-R)]^{1/\beta}$$

در این معادله نیز U_i عدد تصادفی یکپارچه در محدوده $[0,1]$ می باشد و عدد تصادفی نهایی (R_i) حاصل از معادله (۵.۳) دارای توزیع ویبل است.

مثال ۵.۵- توزیع مثلثی (Triangular) - متغیر تصادفی x دارای توزیع آماری مثلثی است. تابع چگالی احتمالی آن عبارتست از:

$$f(x) = \begin{cases} x & 0 \leq x \leq 1 \\ 2-x & 1 < x \leq 2 \\ 0 & \text{سایر} \end{cases}$$



$$F(x) = \begin{cases} 0 & x \leq 0 \\ \frac{x^2}{2} & 0 < x \leq 1 \\ 1 - \frac{(2-x)^2}{2} & 1 < x \leq 2 \\ 1 & x > 2 \end{cases}$$

راه حل:

$$F(x) \text{ - یافتن}$$

$$f(x) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0 \\ 0 & x < 0 \end{cases}$$

راه حل:

$$F(x) \text{ - یافتن}$$

$$F(X) = \int_{-\infty}^x f(t) dt = \int_{-\infty}^x \lambda e^{-\lambda t} dt = 1 - e^{-\lambda x}$$

$$U = 1 - e^{-\lambda x}$$

$$\text{گام ۲- قرار دادن}$$

$$1 - e^{-\lambda x} = R$$

$$e^{-\lambda x} = 1 - R$$

$$-\lambda x = \ln(1 - R)$$

$$x = -\frac{1}{\lambda} \ln(1 - R)$$

$$U_i = R_i$$

$$\text{گام ۳- یافتن } x$$

$$\text{گام ۴- فرمول کلی}$$

$$X_i = -\frac{1}{\lambda} \ln(1 - R_i)$$

$$(۵.۲)$$

از آنجا که U_i دارای توزیع یکپارچه بین $[0,1]$ است $(1-U_i)$ نیز دارای توزیع یکپارچه می باشد. لذا می توان در فرمول فوق به جای U_i از $(1-U_i)$ استفاده نمود:

$$X_i = -\frac{1}{\lambda} \ln U_i \quad i=1,2,\dots$$

در این توزیع اعداد تصادفی بصورت زیر انجام می گیرد:

۱- عدد تصادفی یکپارچه U_i در فاصله $[0,1]$ توسط مولدهای تصادفی یکپارچه تولید می شود.

۲- عدد تصادفی یکپارچه U_i را در فرمول X_i (۵.۲) قرار می دهیم و R_i عدد تصادفی تولیدی نهایی با

توزیع نمایی می باشد.

مثال ۵.۴- توزیع ویبل (Weibull) - فرض کنید که متغیر تصادفی x دارای توزیع آماری ویبل است.

تابع چگالی احتمال این توزیع عبارتست از:

$$f(t) \leq c, \quad A \leq t \leq B$$

پس داریم:

راه حل روش حذفی دارای مراحل زیر می باشد:

گام ۱- یک جفت عدد تصادفی یکنواخت در محدوده $[0,1]$ بنام های U_1 و U_2 تولید می کنیم.

گام ۲- U_1 یک نقطه P را روی محور افقی بصورت زیر تعیین می کند:

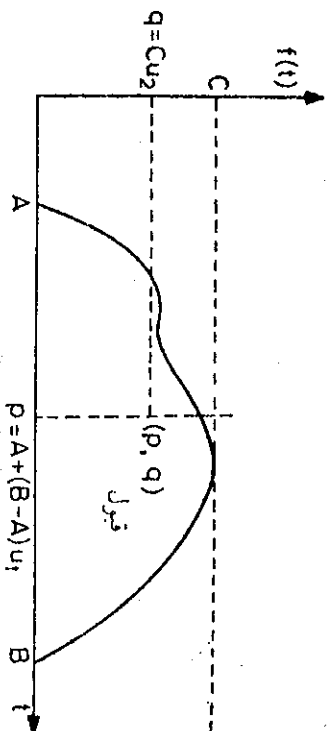
$$P = A + (B-A)U_1$$

گام ۳- U_2 یک نقطه q را روی محور عمودی بصورت زیر تعیین می کند:

$$q = C - U_2$$

گام ۴- اگر $f(P) > q$ باشد اعداد تولیدی U_1 و U_2 حذف می شوند و به گام ۱ برمی گردیم. در غیر اینصورت P به عنوان یک مقدار با توزیع داده شده است.

این روان تکراری شود تا تمامی نمونه ها تولید شده باشند. در این توزیع ها، X بین مقادیر $[A,B]$ تغییر می کند یعنی تغییر متغیر تصادفی مربوطه محدود است. لذا در این روش توزیع با شکلهای دلخواه می توان داشت و نقاط قبول شده مطابق تابع $f(t)$ خواهد بود (شکل ۵-۳).



شکل ۵-۳: یافتن نقطه (P,q) در تابع چگالی احتمال

مثال ۵-۲- فرض می خواهیم که براساس روش حذفی، مولد تصادفی با توزیع یکنواخت بین $\frac{1}{4}$ و $\frac{1}{2}$ را برای تولید X استفاده کنیم:

$$R = \frac{X^2}{2}, \quad 0 \leq X \leq 1$$

$$R = 1 - \frac{(2-X)^2}{2}, \quad 1 \leq X \leq 2.$$

گام ۲- قرار دادن

$$X = \sqrt{2R}, \quad 0 \leq R \leq \frac{1}{2}$$

گام ۳- یافتن X

$$X = 2 - \sqrt{2(1-R)}, \quad \frac{1}{2} \leq R \leq 1$$

گام ۴- فرمول کلی

$$X = \begin{cases} \sqrt{2R}, & 0 \leq R \leq \frac{1}{2} \\ 2 - \sqrt{2(1-R)}, & \frac{1}{2} < R \leq 1 \end{cases} \quad (5.4)$$

با همان عدد تصادفی با توزیع یکنواخت در فاصله $[0,1]$ می باشد که عدد تصادفی نهایی حاصل

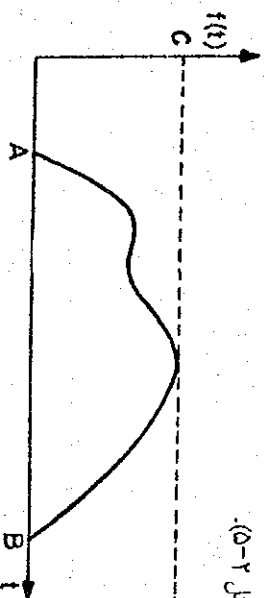
X_1 از توزیع آماری مثالی پیروی می کند.

۵.۲- روش حذفی

روش حذفی برای یافتن نمونه هایی از یک توزیع غیر یکنواخت است، اساس آن بر تولید اعداد تصادفی یکنواخت بطور مکرر می باشد و فقط آنهایی که دارای شرط خاصی هستند قبول می شوند.

در قبول اعداد تصادفی تولیدی یکنواخت آنهایی را که وضعیتی خاص دارند و مورد نظر ما هستند، انتخاب می کنیم و این انتخاب جایی است که در محدوده مورد نظر ما قرار گیرند. پس قبول عدد تولیدی وقتی صورت می گیرد که در محدوده ترسیمی از توزیع داده شده قرار داشته باشند و در غیر این صورت عدد تولیدی یکنواخت حذف می گردد.

در این روش تابع چگالی احتمال $f(t)$ باید در محدوده (A,B) غیر صفر باشد و تابع دارای محدوده بالای C است (شکل ۵-۲).



شکل ۵-۲: تابع چگالی احتمال $f(x)$

| نتیجه | قبول/حذف | P | $R_n + 1$ | n |
|-------|----------|--------|-----------|---|
| $N=0$ | قبول | ۰/۴۳۵۷ | ۰/۴۳۵۷ | ۰ |
| $N=0$ | قبول | ۰/۴۱۴۶ | ۰/۴۱۴۶ | ۰ |
| | حذف | ۰/۸۳۵۳ | ۰/۸۳۵۳ | ۰ |
| | حذف | ۰/۸۳۱۳ | ۰/۹۹۵۲ | ۱ |
| $N=2$ | قبول | ۰/۶۶۵۴ | ۰/۸۰۰۴ | ۲ |

مثال ۵-۹- گامهای تولید عدد تصادفی با توزیع گاما و روش حذفی با فرض پارامتر مقیاس θ و پارامتر

شکل β عبارتست از:

$$\text{گام ۱- محاسبه } \beta = (2\theta - 1)^{1/2}, \quad a = (2\theta - 1)^{1/2}, \quad b = 2\theta - \ln 4 + 1/a$$

گام ۲- تولید $R_1 R_2$

$$\text{گام ۳- محاسبه } X = \beta(R_1/(1-R_1))^a$$

گام ۴- اگر $x > b - \ln(R_1^2 R_2)$ باشد لذا حذف می شود و به مرحله ۲ می رویم.

گام ۵- $X = X/\beta\theta$ (با میانگین ۱ و واریانس $1/\theta^2$).

۵-۳ روش بکارگیری خواص ویژه

روشی برای تولید اعداد تصادفی با توزیع های گوناگون است و در مواقعی از این روش استفاده

می گردد که با استفاده از خواص ویژه ای بتوان پیچیدگی راه حل را کاهش داد. در بعضی موارد برای

یک توزیع خاص، خواص ویژه ای وجود دارد و به یک تابع دیگری آن را مرتبط می سازد که در آن تابع

جدید (فضای جدید) نمونه های تصادفی مورد نظر سهل تر بدست می آید. مثالهایی از آن عبارتند از:

- یک نمونه نرمال لگاریتمی محاسبه شده از \ln جایی که \ln نمونه ای با توزیع نرمال است.
- یک نمونه ارلنگ به عنوان جمع نمونه های گاما است.
- یک نمونه بتا به عنوان نرخ نمونه های گاما است.
- یک نمونه دو جمله ای به عنوان مجموع (جمع) رشته برتولی است.
- یک نمونه نرمال به عنوان جمع نمونه های یکپارچه است.
- یک نمونه کای دو (χ^2) اگر درجه آزادی زوج باشد از یک نمونه ارلنگ بدست می آید.
- یک نمونه توزیع χ^2 نرخ دو نمونه کای دو (χ^2) است.

گام ۱- تولید عدد تصادفی R

گام ۲- اگر R بزرگتر یا مساوی $\frac{1}{4}$ است، $x=R$ و به گام ۴ برویم.

گام ۳- حذف R و به گام ۱ برویم.

گام ۴- اگر عدد تصادفی یکپارچه دیگری نیاز است، به گام ۱ برویم.

مثال ۵-۷- با استفاده از روش حذفی برای اعداد با توزیع پواسون یک روال تولید عدد تصادفی

بنویسید:

$$p(n) = P(N=n) = \frac{e^{-\alpha} \alpha^n}{n!}, \quad n = 0, 1, 2, \dots$$

با استفاده از ارتباط توزیع پواسون (گسسته) و توزیع نمایی (پیوسته)، روال تولید یک عدد تصادفی

پواسون N عبارتست از:

$$\text{گام ۱- } p=1 \text{ و } n=0$$

گام ۲- تولید عدد تصادفی R_{n+1} و قرار دادن $P.R_{n+1}$ بجای P

گام ۳- اگر $P < e^{-\alpha}$ است لذا $N=n$ و در غیر اینصورت n حذف می شود $(n=n+1)$ و سپس به گام ۲

می رویم.

مثال ۵-۸- برای تولید ۳ متغیر پواسون با میانگین $\alpha=0.2$ و روش حذفی، اول $e^{-\alpha}=e^{-0.2}=e^{-0.2}$

محاسبه می کنیم. سپس مراحل زیر دنبال می شود:

$$\text{گام ۱- } p=1 \text{ و } n=0$$

$$\text{گام ۲- } R_1=0.4357, \quad P=1 \times R_1=0.4357$$

گام ۳- از آنجا که $0.18187 < 0.4357$ لذا $N=0$ قبول می شود.

$$\text{گام ۱- } p=1 \text{ و } n=0, \quad R_1=0.41146$$

$$\text{گام ۲- } R_1=0.8353$$

$$\text{گام ۳- از آنجا که } p \geq e^{-\alpha} \text{ لذا } n=0 \text{ حذف می شود و با } n=1 \text{ می رویم.}$$

$$\text{گام ۱- } p=1 \text{ و } n=0, \quad R_1=0.9952$$

$$\text{گام ۲- از آنجا که } p \geq e^{-\alpha} \text{ لذا } n=1 \text{ حذف می شود و با } n=2 \text{ می رویم.}$$

$$\text{گام ۲- } R_1=0.8004, \quad R_2=0.6654$$

$$\text{گام ۳- از آنجا که } p < e^{-\alpha} \text{ است لذا } N=2 \text{ قبول می شود.}$$

لذا می توان نشان داد:

$$S = (-\sqrt{2n}U_1) \frac{1}{\sqrt{2}} \cos(\sqrt{2n}U_1)$$

با فرض

هر نمونه تصادفی x با توزیع نرمال با مقدار مشخص μ و σ می توان یافت.اگر U_1 و U_2 دو عدد تصادفی یکپوش در $[0,1]$ باشند:معادله توزیع نرمال با پارامترهای μ و σ

$$X_1 = \sigma \cdot S + \mu$$

(۵.۵)

راه حل:

گام ۱- تولید تصادفی های یکپوش U_1 و U_2 گام ۲- یافتن تبدیل S گام ۳- تولید تصادفی X_1 با توزیع نرمالدر این روش برای افزایش دقت عملیات می توان روش دیگری نیز برای یافتن S کار برد. خاصیتحد مرکزی از ویژگیهای توزیع نرمال است که از آن استفاده می کنیم. در صورتی که x_1, x_2, \dots, x_n اعدادتصادفی یکپوش باشند، آنگاه $x_1 + x_2 + \dots + x_n = X$ دارای توزیع نرمال با میانگین μ و واریانس σ^2 خواهد بود. اگر x_1, x_2, \dots, x_n متغیرهای تصادفی در فاصله U_1, U_2, \dots, U_n باشند، میانگین آنها در فاصله

$$[0,1] \text{ است و در فواصل بالا نیز } \sigma^2 = \frac{1}{12} \text{ می شود.}$$

$$S = \frac{\sum_{i=1}^n U_i - n\mu}{\sigma \sqrt{n}}$$

می توان به σ مقدار مشخصی داد. ساده ترین راه آن است که $n=12$ فرض کنیم، $\mu=\frac{1}{2}$ و $\sigma^2=\frac{1}{12}$:

$$S = \sum_{i=1}^{11} U_i - 6$$

لذا با انتخاب ۱۲ عدد تصادفی، S مورد نظر را می یابیم و در فرمول $X_1 = \sigma S + \mu$ قرار می دهیم.

مثال ۵.۱۱- توزیع برنولی - مولد تصادفی تابع توزیع برنولی را بدست آورید.

متغیرهای تصادفی گسسته x تنها مقادیر ۰ و ۱ می گیرند. که احتمال هر کدام به صورت زیر است:

$$P(x=0) = P(U \leq P) = P \quad \text{احتمال پیروزی} \quad p+q=1$$

$$P(x=1) = P(U > P) = q = 1-P \quad \text{احتمال شکست} \quad (U \text{ تصادفی یکپوش})$$

لذا اگر x دارای توزیع برنولی با پارامتر P می باشد. چنانچه متغیر تصادفی یکپوش U از P کمتر یا

در این روش به جای استفاده از تبدیل معکوس از خواص ویژه ای استفاده می شود که مثالهایی از آن را برای چند توزیع ارائه می نمایم.

مثال ۵.۱ - توزیع نرمال (Normal gaussian) - تابع مولد تصادفی توزیع نرمال را بیابید. تابع چگالی

احتمال آن را داریم:

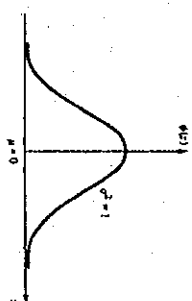
$$f(x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{x-\mu}{\sigma} \right)^2}$$

$$-\infty < x < \infty.$$

با فرض $N(0,1)$ توزیع نرمال استاندارد بدست می آید:

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$$

$$\sigma = 1, \quad \mu = 0$$



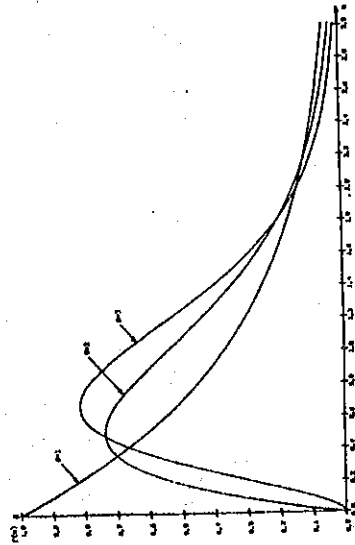
شکل ۵-۴- نرمال استاندارد

$$F(x) = \int_{-\infty}^x \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt$$

داریم:

برای حل آن روشهای بسیاری وجود دارد که استفاده از تبدیلات خاص یکی از آنها است. به

عنوان نمونه با تبدیل باکس-مولر می توان پاسخ را چنین داشت:



شکل ۵-۵ توزیع گاما

اگر x_1, x_2, \dots, x_n دارای توزیع نمایی با میانگین β باشند آنگاه:

$$X = X_1 + X_2 + \dots + X_n$$

دارای توزیع گاما با پارامترهای α و β می باشد

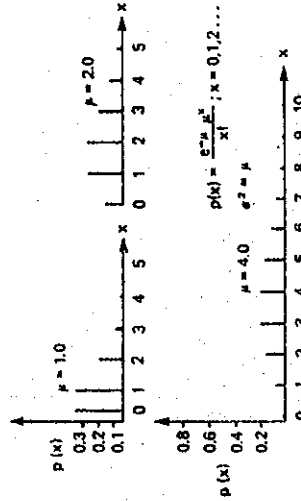
$$X = X_1 + X_2 + \dots + X_n = \sum_{i=1}^n X_i = -\beta \sum_{i=1}^n \ln(U_i) \quad (5.6)$$

$$= -\beta \ln(u_1 \times u_2 \times \dots \times u_n) = -\beta \ln\left(\prod_{i=1}^n U_i\right)$$

پس $X_i = -\beta \ln\left(\prod_{i=1}^n U_i\right)$ برای $i=1, 2, \dots, n$

مثال ۵-۱۶ - توزیع پواسون - مولد تصادفی بیاید که از توزیع پواسون پیروی کند.

چنانچه زمان بین وقوع پیشامدها x_1, x_2, \dots, x_n دارای توزیع نمایی با پارامتر λ باشد، آنگاه متغیر تصادفی تعداد پیش آمدها در t واحد زمان دارای توزیع پواسون با پارامتر λt است.



شکل ۵-۶ توزیع پواسون

مسای آن باشد، متغیر تصادفی نهایی تابع توزیع U_i تصادفی یکنواخت

برنولی می شود و در غیر اینصورت ۱ است.

مثال ۵-۱۲ - اگر فرض کنیم که تصادفی های مورد نیاز دارای سیستمی با توزیع برنولی با پارامتر $P=0.6$ باشد.

تصادفی یکنواخت $U_i: 0.3, 0.4, 0.8, 0.2, 0.7$

تصادفی با توزیع برنولی $U_i: 0, 0, 1, 0, 1$

مثال ۵-۱۳ - توزیع دو جمله ای - مولد تصادفی بیاید که از توزیع دو جمله ای پیروی کند.

با توجه به این که تابع مولد گشتاور توزیع دو جمله ای $M(t) = (1-p+pe^t)^M$ است و تابع مولد گشتاور

برنولی $M(t) = (1-p+pe^t)$ می باشد. بنابر قضایای احتمال، اگر فرض کنیم که x یعنی تعداد موفقیتها در n آزمایش است:

$$\binom{n}{x} = \frac{n!}{x!(n-x)!}$$

چنانچه اعداد تصادفی x_1, x_2, \dots, x_n دارای توزیع برنولی باشند، لذا

$$X_i = \sum_{i=1}^n x_i$$

آنگاه x_i دارای توزیع دو جمله ای است.

مثال ۵-۱۴ - در توزیع برنولی رشته مثال ۵.۱۲ داشتیم: 0,0,1,0,1

لذا یک دو جمله ای است.

$$X_1 = \sum_{i=1}^5 x_i = 2$$

مثال ۵-۱۵ - توزیع گاما - چنانچه تابع چگالی احتمال توزیع گاما بصورت زیر باشد، مولد تصادفی آن

رایباید.

تابع چگالی احتمال آن را داریم:

$$f(x) = \begin{cases} \frac{1}{\beta^\alpha \Gamma(\alpha)} e^{-x/\beta} x^{\alpha-1} & x > 0 \\ 0 & x \leq 0 \end{cases}$$

α پارامتر شکل،

β پارامتر مقیاس،

$$\mu = \frac{\alpha}{\beta}, \sigma^2 = \frac{\alpha}{\beta^2}$$

فصل ششم

شبیه‌سازی مونت کارلو

تعاریف بسیاری برای روش مونت کارلو ارائه شده است. در اینجا روش مونت کارلو یا شبیه‌سازی

مونت کارلو به بیان زیر تعریف می‌گردد:

مونت کارلو روشی است که به منظور حل مسائل قطعی (غیر تصادفی) و یا برخی مسائل تصادفی که گذشت زمان نقش اساسی در آنها ندارد، از اعداد تصادفی استفاده می‌کند.

مظور از اعداد تصادفی متغیرهای تصادفی بکوارخت و مستقل دربارهٔ [0,1] است. نام مونت کارلو از شهر مدیرانامای مونت کارلو گرفته شده است و در ابتدای جنگ دوم جهانی در آزمایشگاه لوس آلamos توسط نیرومن و اولام این روش پایه‌گذاری گردید. آزمایش شبیه‌سازی مونت کارلو به منظور طراحی حفاظ مواد رادیو اکتیو و دانستن چگونگی حرکت نوترونها در مواد مختلف انجام شد و از آنجا که حل تحلیلی این مسئله مشکل بود با استفاده از کامپیوترهای سریع و اعداد تصادفی صورت گرفت. این روش را مونت کارلو نامیدند چون براساس فلسفه شرط بندی قرار داشت.

مونت کارلو روشی ایستا و نه پویا شمرده می‌شود. در مقابل روش مونت کارلو می‌توان روش شبیه‌سازی را قرار داد. اگرچه شبیه‌سازی نیز مانند مونت کارلو از اعداد تصادفی استفاده می‌کند ولی تشابه دو روش در همین جا به پایان می‌رسد. در واقع عامل زمان در شبیه‌سازی دخالت دارد و به بیان دیگر شبیه‌سازی روشی پویا محسوب می‌شود. مضافاً آن که اکثر مسائل شبیه‌سازی دارای طبیعتی تصادفی هستند.

تصور عمومی مردم از روشی مونت کارلو شامل محاسباتی است که از اعداد تصادفی استفاده می‌شود. گرچه قبول مونت کارلو تعیین اعداد تصادفی است که راه حل مسئله را بدهند و این ذاتاً محاسبه قطعی است. محاسبه گر اعداد تصادفی را برای حل مسئله تصادفی نیز برمی‌گیرند که سیستم تصادفی می‌باشد. برای چنین محاسباتی که اعداد تصادفی برای حل مسائل تصادفی بکار می‌رود واژهٔ "شبیه‌سازی تصادفی" استفاده می‌گردد. به عنوان مثال سیستم انبار یک شبیه‌سازی تصادفی است و نه روش مونت کارلو.

تابع احتمال خواهد بود: $P(x) = \frac{(n!)^x e^{-nx}}{x!}$ $x = 1, 2, 3, \dots$ (اعداد پیش آمده)

اگر U_1 عدد تصادفی بکوارخت باشد، دارای توزیع نمایی بصورت زیر خواهد بود:

$$f_i = -\frac{1}{\lambda} \ln(U_i)$$

$$\sum_{i=1}^x t_i \leq t < \sum_{i=1}^{x+1} t_i$$

می‌توان t را چنین در نظر گرفت:

$$\sum_{i=1}^x -\frac{1}{\lambda} \ln(U_i) \leq t < \sum_{i=1}^{x+1} -\frac{1}{\lambda} \ln(U_i)$$

با جایگذاری t می‌شود:

$$-\frac{1}{\lambda} \sum_{i=1}^x \ln(U_i) \leq t < -\frac{1}{\lambda} \sum_{i=1}^{x+1} \ln(U_i)$$

$$\sum_{i=1}^x \ln(U_i) \geq -\lambda t > \sum_{i=1}^{x+1} \ln(U_i)$$

$$\sum_{i=1}^k \ln(U_i) = \ln(\prod_{i=1}^k U_i)$$

چون

$$\ln \prod_{i=1}^x U_i \geq -\lambda t > \ln \prod_{i=1}^{x+1} U_i$$

با جایگذاری می‌شود:

$$\prod_{i=1}^x U_i \geq e^{-\lambda t} > \ln \prod_{i=1}^{x+1} U_i$$

پس خواهیم داشت:

راه حل:

گام ۱- با مقدار اولیه $x=1$ و فرض $P=1$ شروع می‌شود.

گام ۲- عدد تصادفی بکوارخت U_1 تولید می‌شود.

گام ۳- $P = P \cdot U_1$ محاسبه می‌شود.

گام ۴- اگر $P < e^{-\lambda t}$

باشد آنگاه $x=x+1$ و به گام ۲ می‌رویم.

اگر $P > e^{-\lambda t}$

روش مونت کارلو عمومی‌ترین روش تقریب احتمالاتی حالت‌های یک مدل از طریق نمونه‌های $\{x_1, \dots, x_n\}$ است. این نوع شبه‌سازی تفکیک ثابت نمونه‌ها می‌باشد که تصور یک مدل را فراهم نمی‌آورد بلکه فقط همبستگی آماری بین ورودی‌ها و خروجی‌ها کاوش می‌شود. اشیاء سیستم پرسیله متغیرها ارائه می‌شوند و ارتباط آنها از طریق تکرار تولید اعداد تصادفی یا توزیع احتمالاتی اعداد برقرار می‌گردد. زمانی که راه‌های مدلهای غیر احتمالاتی پیچیده و پرهزینه است از این روش استفاده می‌شود که در آن به دنبال آنالیز نتایج نیستیم بلکه همبستگی‌های آماری را جستجو می‌کنیم. در بخش زیر به اختصار توضیحاتی در زمینه موارد استفاده از مونت کارلو عرضه می‌شود.

الف - یک مورد کاربرد مونت کارلو مربوط به حل مسائل قطعی (غیر تصادفی) با استفاده از اعداد تصادفی است. به عنوان مثال، فرض کنید که قصد برآورد $I = \int_a^b g(x) dx$ را داریم به طوری که انتگرال تابع حقیقی $g(x)$ را بتوان از طریق روش تحلیلی پیدا کرد. برای اینکه با استفاده از روش مونت کارلو این مسئله قطعی (غیر تصادفی) را حل کنیم، متغیر تصادفی y را بصورت $y = (b-a)g(x)$ تعریف می‌کنیم، جایی که x یک متغیر تصادفی یکپارچه در بازه $[a, b]$ است $X \sim U(a, b)$ حال می‌توان نشان داد که امید ریاضی y به شرح زیر بدست می‌آید:

$$\begin{aligned} E(y) &= E[(b-a)g(x)] = (b-a)E[g(x)] \\ &= (b-a) \int_a^b g(x)f(x) dx \\ &= (b-a) \int_a^b g(x) dx / (b-a) = I \end{aligned}$$

در رابطه فوق تابع $f(x)$ چگالی متغیر تصادفی x علامتگذاری شده و به صورت $f(x) = \frac{1}{b-a}$ تعریف می‌شود. بدین ترتیب، مسئله یافتن انتگرال به تقریب امید ریاضی y تبدیل شده است، لذا برای یافتن $E(y) = I$ از میانگین نمونه، یعنی:

$$\bar{y}(n) = \sum_{i=1}^n \frac{y_i}{n} = (b-a) \sum_{i=1}^n \frac{g(x_i)}{n}$$

استفاده می‌کنیم به طوری که x_1, x_2, \dots, x_n مستقل و یکپارچه در بازه $[a, b]$ باشند. برای درک بهتر فرض می‌کنیم که $\bar{y}(n)$ متوسط مساحت n مستطیل است که طول همه آنها $(b-a)$ و ارتفاع هر یک از آنها $g(x_i)$ است. مضافاً می‌توان نشان داد که روابط $E[\bar{y}(n)] = I$ و

$Var(\bar{y}(n)) = Var(y)/n$ را به I نزدیک کرد. با توجه به وجود روشهای کارآتر تقریب انتگرال‌های ساده، احتمال کاربرد روش مونت کارلو در رابطه با مثالهایی از نوع بالا چندان زیاد نیست. در واقع اگر انتگرال مضاعف و تابع $g(x)$ نیز تابعی پیچیده باشد، انتخاب مونت کارلو به عنوان روش تحلیل معقول‌تر است.

ب - مورد دیگر کاربرد مونت کارلو نمونه‌گیری از توزیعهای آماری مجهول است. هدف از این نمونه‌گیری یافتن توزیع آماری یک یا چند متغیر تصادفی یا پارامترهای آن است. متغیر تصادفی مورد نظر را متغیر پاسخ می‌نامیم. متغیر پاسخ تابعی از یک یا چند متغیر تصادفی شناخته شده است. به منظور ارائه تخمینی برای توزیع آماری متغیر پاسخ، مقادیری برای کلیه متغیرهای تصادفی ورودی تولید و مقدار متناظر متغیر پاسخ را بر اساس آنها محاسبه می‌کنیم. این نمونه‌گیری آفندر تکرار می‌گردد تا تخمینی از توزیع آماری متغیر تصادفی ایجاد شود. مثالی از آن، برآورد تابع توزیع تقاضا در خلال مهلت تحویل در یک مسئله کنترل موجودی است. در زیر به مثالهایی از کاربرد این روش اشاره می‌کنیم.

۶-۱- کاربردهای شبه‌سازی مونت کارلو

مثال ۶-۱ - حل یک مسئله تصادفی - در این مثال به حل یک مسئله تصادفی جهت ایجاد همبستگی آماری در یک سیستم صف با روش مونت کارلو می‌پردازیم. شبه‌سازی سیستم صف پاسخگوئی (مشاوره) استاید با دانشجویان در آشنای هفته ثبت نام است. فرض n شاگرد در زمان $t=0$ و عدد ملاقات دارند و ورود بقیه در فواصل نیم ساعته زمانبندی می‌شوند. زمان مراجعه نمائی و با میانگین ۳۰ دقیقه است. سؤال اینست که زمانبندی ملاقات دانشجویان چگونه باشد تا تعامل بین زمان انتظار دانشجویان و زمان انتظار استاید برقرار گردد. برای دو وضعیت $n=1$ و $n=2$ پروسه شبه‌سازی انجام می‌گیرد:

x = متغیر تصادفی با توزیع نمائی و میانگین ۱۰۰ است.

$Sarr$ = زمان ورود یک دانشجو مطابق وعده ملاقات است که از یک عدد تصادفی برای تولید زمان ورود استفاده می‌شود.

t_{beg} = زمان شروع مشورت دانشجو می‌باشد.

t_{end} = زمان پایان مشورت دانشجو می‌باشد.

برای تغییر یا تبدیل مسئله اصلی جهت دریافت برآوردهای مطلوب بکار می‌رود. البته این روش‌ها کمی پیچیده هستند و برای افزایش دقت بکار می‌روند.

یک روش تقلیل واریانس برای افزایش دقت بکار می‌روند. یک روش تقلیل واریانس با حداقل فرمول ریاضی جهت کاربرد این مسئله را شرح می‌دهیم. $N1$ به عنوان متغیر تصادفی زمان انتظار برای $N1$ تعریف می‌شود. $\bar{N1}$ میانگین تقریب $N1$ است. بطور مشابه $N2$ را به عنوان زمان انتظار برای $N2$ با تقریب میانگین $\bar{N2}$ است. اختلاف L و تقریب میانگین دو وضعیت می‌شود:

$$\bar{D} = \bar{N}_1 - \bar{N}_2$$

تقلیل واریانس وقتی مهم است که دو حالت را مقایسه می‌کنیم زیرا می‌خواهیم تقریب اختلاف نسبی \bar{D} را از قدر مطلق $N1, N2$ تعیین کنیم. اگر از دو توالی تصادفی برای یافتن $N1$ و $N2$ استفاده کنیم؛ پس واریانس را بصورت زیر می‌یابیم:

$$\sigma^2[D] = \sigma^2[N_1] + \sigma^2[N_2]$$

گرچه، اگر از توالی اعداد تصادفی یکسان برای هر دو استفاده می‌کنیم؛ سپس هر دو وضعیت تحت شرایط یکسان مقایسه خواهند شد. استفاده از اعداد تصادفی یکسان همبستگی بین متغیرهای تصادفی $N1$ و $N2$ را تشخیص می‌دهد. حال متغیر تصادفی D^* را تقریب می‌زنیم که میانگین یکسان مانند D دارد که:

$$\bar{D}^* = \bar{D}$$

واریانسش است:

$$\sigma^2[D^*] = \sigma^2[N_1] + \sigma^2[N_2] - 2\sigma^2[N_1, N_2]$$

وقتی که $N2, N1$ همبستگی دارند $[D^*]$ از $[D]$ σ^2 کوچکتر است. جدول ۲-۶ همان روند را برای تولید Y از سری متقابل (antithetic) با توزیع نمایی جدول ۱-۶ دارد.

CT زمان گفتگو برای مشاوره است و عدد تصادفی با توزیع نمایی و میانگین ۳۰ است که عبارت است از:

$$CT = (30/100)X$$

فرض دانشجویانی که در صف انتظار قرار دارند با روش FIFO پاسخگویی می‌شوند:

فرمول اساسی عبارتست از:

$$T_{end} = T_{beg} + CT$$

جدول ۱-۶ نتایج هر دو شبیه‌سازی برای $m=10$ دانشجو را نشان می‌دهد.

| شماره دانشجو | $n=1$ | | | | | $n=2$ | | | | |
|--------------|-----------|-----------|-----|------|-----------|-----------|-----------|-----|------|-----------|
| | S_{sum} | T_{sum} | X | CT | T_{end} | S_{sum} | T_{sum} | X | CT | T_{end} |
| 1 | 0* | 0 | 199 | 60 | 60 | 0* | 60 | | | |
| 2 | 30 | 60 | 286 | 86 | 146 | 0 | 146 | | | |
| 3 | 60 | 146 | 55 | 17 | 163 | 30 | 163 | | | |
| 4 | 90 | 163 | 280 | 84 | 247 | 60 | 247 | | | |
| 5 | 120 | 247 | 8 | 2 | 249 | 90 | 249 | | | |
| 6 | 150 | 249 | 186 | 56 | 305 | 120 | 305 | | | |
| 7 | 180 | 305 | 77 | 23 | 328 | 150 | 328 | | | |
| 8 | 210 | 328 | 81 | 24 | 352 | 180 | 352 | | | |
| 9 | 240 | 352 | 127 | 38 | 390 | 210 | 390 | | | |
| 10 | 270 | 390 | 38 | 11 | 401 | 240 | 401 | | | |
| | 1350 | | | 401 | 2641 | 1080 | 2647 | | | |

جدول ۱-۶ نتایج شبیه‌سازی برای دو آزمایش

* دانشجو انتظار رکشیده است.

از جدول فوق انتظار دانشجویان را داریم:

| زمان انتظار اساتید | زمان انتظار دانشجویان |
|------------------------|---|
| $(T_{sum} - \sum CT)$ | $(\sum T_{sum} - \sum S_{sum} - \sum CT)$ |
| $n=1$
401 - 401 = 0 | 2641 - 1350 - 401 = 890 |
| $n=2$
401 - 401 = 0 | 2647 - 1080 - 401 = 1166 |

به علت مشکلات و سردرگمی روش مونت کارلو، فنون کاهش واریانس مقبولیت یافته است که اساس آن، بکارگیری دانش و اطلاعات مربوط به ساختار مدل و خواص ورودیهاست. این دانش

تغییر در شبیه‌سازی بهتر است. دو روش تقلیل واریانس داریم:

- ۱- استفاده از سری‌های متقابل
- ۲- استفاده از اعداد یکسان در وضعیت‌های مرتبط

مثال ۶-۲- در ۱۰ شبیه‌سازی سیستم صف مثال ۱، کل زمان در سیستم یعنی زمان از ورود تا خروج عبارت بود از:

۱۹۹ ۲۸۶ ۲۹۴ ۵۵ ۲۸۰ ۸ ۱۸۶ ۷۷ ۸۱ ۱۲۷

میانگین این مقادیر $179/3$ و واریانس آن $104/76$ است. اگر از سری متقابل مولد اعداد تصادفی استفاده کنیم، کل زمان برای ۱۰ شبیه‌سازی خواهد بود:

۱۵ ۶ ۵ ۸۵ ۲۵۳ ۱۷ ۶۲ ۵۹ ۳۳ ۱۱۵

میانگین ۶۵ و واریانس $75/42$ می‌باشد.

روش تقلیل واریانس از تکرار آزمایشات شبیه‌سازی می‌کاهد و معطوف به سری‌های تصادفی مورد نظر می‌شود.

مثال ۶-۳ - نمونه‌برداری از توزیع مجهول آماری - در هر سیستم کنترل موجودی که مهلت تحویل کالا برابر صفر نباشد، تقاضای طی مهلت تحویل کالا می‌تواند مطرح گردد. مهلت تحویل از لحظه صدور سفارش تا لحظه دریافت کالا طول می‌کشد. می‌توان با یک نگرش صحیح مهلت تحویل کالا را به منزله یک متغیر تصادفی مورد بررسی قرار داد و تغییرات تقاضای طی مهلت تحویل کالا نیز دارای طبیعتی تصادفی خواهد بود. اگر مهلت تحویل کالا T واحد زمان طول بکشد و تقاضا در آن واحد زمان برابر D_i باشد ($i=0,1,\dots,T$)، مهلت تحویل کالا با T و تقاضای مهلت تحویل با D_i مشخص می‌شود. به منظور تعیین توزیع آماری تقاضای طی مهلت تحویل کالا، می‌توان به دفعات بسیار به شبیه‌سازی مهلت تحویل دوره‌های مختلف انبار پرداخت و براساس نتایج حاصله نمودار متواتر آن را رسم کرد.

مؤسسه‌ای به فروش کاغذ روزانه بصورت عدلی اشتغال دارد. توزیع آماری تقاضای روزانه

بشرح زیر است:

$n = 1$ $n = 2$

| شماره دانشجو | S_{err} | T_{err} | Y | CT | T_{end} | S_{err} | T_{end} |
|--------------|-----------|-----------|-----|------|-----------|-----------|-----------|
| 1 | 0* | 0 | 15 | 5 | 5 | 0* | 5 |
| 2 | 30* | 30 | 6 | 2 | 32 | 0* | 7 |
| 3 | 60* | 60 | 5 | 2 | 62 | 30* | 32 |
| 4 | 90* | 90 | 86 | 26 | 116 | 60* | 86 |
| 5 | 120* | 120 | 6 | 2 | 122 | 90* | 92 |
| 6 | 150* | 150 | 253 | 76 | 226 | 120* | 196 |
| 7 | 180 | 226 | 17 | 5 | 231 | 150 | 201 |
| 8 | 210* | 231 | 62 | 19 | 250 | 180 | 220 |
| 9 | 240* | 250 | 59 | 18 | 268 | 210 | 238 |
| 10 | 270 | 270 | 33 | 10 | 280 | 240* | 250 |
| | 1350 | | | 165 | 1392 | 1080 | 1372 |

جدول ۶-۲ نتایج شبیه‌سازی با استفاده از سری‌های متقابل

از جدول ۶-۲

| | انتظار اساتید | انتظار دانشجویان |
|---------|---------------------|--------------------------|
| $n = 1$ | $280 - 165 = 116$ | $1592 - 135 = 165 = 77$ |
| $n = 2$ | $250 - 165 = 85$ | $1327 - 1080 = 247 = 82$ |
| $n = 1$ | $(0 + 116)/2 = 58$ | $(890 + 77)/2 = 484$ |
| $n = 2$ | $(0 + 85)/2 = 42.5$ | $(1159 + 82)/2 = 620.5$ |

خلاصه هر دو شبیه‌سازی بوسیله سری میانگین زیر دارد:

انتظار دانشجویان

انتظار اساتید

اگر می‌خواهیم به مقدار n مطمئن برسیم باید آزمایشات زیادی انجام دهیم. با زبان‌های شبیه‌سازی عمومی مانند (GASP یا SIMAN، SLAM، SIMSCRIPT، GPSS) می‌توان با اجرای چندین شبیه‌سازی مقدار n مورد نظر یافت.

تقلیل واریانس

برای جلوگیری از تکرار شبیه‌سازی در رسیدن به پاسخ مورد نظر، از روش تقلیل انحراف معیار جهت تولید تصادفی‌ها استفاده می‌کنیم. در مثال ۱، مثالی از استفاده سری‌های متقابل و محاسبه میانگین هر دو جفت آزمایش ارائه شد. از سری متقابل استفاده شد زیرا اثر تغییر روی میانگین از اثر

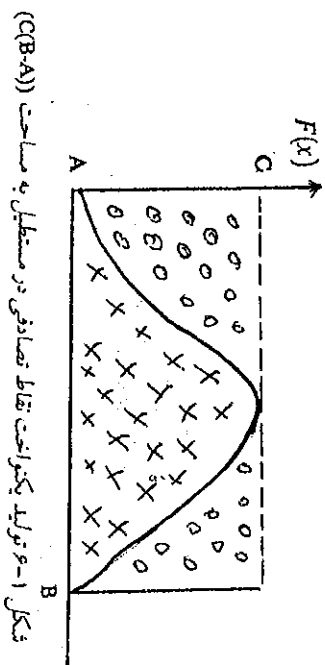
| دوره | زمان تأخیر
مهاجرت | عدد تصادفی
برای تأخیر | زمان تأخیر
تأخیر | عدد تصادفی
برای تأخیر | زمان تأخیر
تأخیر |
|------|----------------------|--------------------------|---------------------|--------------------------|---------------------|
| 1 | 57 | 2 | 87 | 6 | 10 |
| 2 | 33 | 1 | 34 | 4 | 5 |
| 3 | 93 | 3 | 82 | 5 | 12 |
| | | | 28 | 4 | 10 |
| | | | 19 | 3 | 5 |
| | | | 63 | 5 | 6 |
| 4 | 55 | 2 | 91 | 6 | 4 |
| | | | 26 | 4 | 10 |
| | | | | | |
| | | | | | |
| | | | | | |

جدول ۵-۶ جدول شبیه‌سازی تقاضای طی مهاجرت کالا

نمودار متواتر می‌تواند مانند شکلی که نمودار جدول ۵-۶ نشان می‌دهد، باشد. این مثال نشان می‌دهد که چگونه می‌توان با استفاده از شبیه‌سازی مونت کارلو یک نمونه تصادفی از یک توزیع آماری نامشخص، توزیع آماری مزبور را مورد مطالعه قرار داد.

مثال ۱-۳ حل یک مسئله قطعی (فرض تصادفی)

تقریب مساحت یک ناحیه با استفاده از روش مونت کارلو را توضیح می‌دهیم که از روش حذفی برای این امر استفاده می‌کند. مساحت مجهول باید در ناحیه‌ای با مساحت معلوم محاط گردد. اگر برای سهولت ارائه مطلب، تابع چگالی بنا را محور بحث قرار دهیم؛ به راحتی قادر به محاط کردن آن در یک چهارضایی مشابه شکل ۱-۶ خواهیم بود.



| تقاضای روزانه (عدلی) | 3 | 4 | 5 | 6 |
|----------------------|------|------|------|------|
| احتمال: | 0.20 | 0.35 | 0.30 | 0.15 |

طول مهاجرت تحول کالا بر حسب روز بیان می‌شود و توزیع آماری آن عبارتست از:

| تأخیر تحول (روز) | 1 | 2 | 3 |
|------------------|------|------|------|
| احتمال: | 0.36 | 0.42 | 0.22 |

چگونگی تخصیص اعداد تصادفی برای تقاضا و طول مهاجرت تحول کالا در جدول ۳-۶ و ۴-۶ نشان داده شده است.

| اعداد تصادفی | احتمال تجمعی | احتمال | تقاضای روزانه |
|--------------|--------------|--------|---------------|
| 01-20 | 0.20 | 0.20 | 3 |
| 21-55 | 0.35 | 0.35 | 4 |
| 56-85 | 0.85 | 0.30 | 5 |
| 86-100 | 1.00 | 0.15 | 6 |

جدول ۳-۶ تخصیص اعداد تصادفی جهت تقاضا

| اعداد تصادفی | احتمال تجمعی | احتمال | زمان تأخیر
(روز) |
|--------------|--------------|--------|---------------------|
| 01-36 | 0.36 | 0.36 | 1 |
| 37-78 | 0.78 | 0.42 | 2 |
| 79-100 | 1.00 | 0.22 | 3 |

جدول ۴-۶ تخصیص اعداد تصادفی جهت مهاجرت کالا

نتایج ناتمام شبیه‌سازی این مسئله در جدول ۵-۶ ارائه شده است. در اولین دوره بازرسی انبار، عدد تصادفی ۵۷ تولید شده که معرف یک مهاجرت تحول ۲ روزه کالا است. به این ترتیب باید دو عدد تصادفی دو رقمی تولید نمود که اولین عدد در رقمی ۸۷ منحصر به دریافت تقاضای معادل ۶ عدلی باشد. می‌گردد و دومین عدد تصادفی دو رقمی ۳۴ با تقاضای متناظر ۴ عدلی است. در نتیجه در اولین دوره تقاضای طی مهاجرت تحول کالا معادل ۱۰ عدلی می‌شود. پس از شبیه‌سازی برای تعداد فراوانی دوره، می‌توان براساس نتایج حاصله یک نمودار متواتر رسم کرد.

در شکل ۶-۱ دامنه تغییرات x با (AB) مشخص شده است. برای تولید یکنواخت اعداد تصادفی در داخل مستطیل فوق، کافی است که به طور یکنواخت به تولید مقادیر تصادفی برای دو مختصات نقطه (U_1, U_2) اقدام کنیم. بدیهی است که U_1 و U_2 معروف دو عدد تصادفی هستند و باید به طور مستقل تولید شوند. از نقاط تولید شده، آنهایی که در بالای تابع چگالی قرار می‌گیرند را حذف کرده و مختصات باقی‌مانده نقاط را به عنوان مقادیر تصادفی از توزیع آماری X می‌پذیریم.

از آنجا که مساحت مستطیل نمودار فوق معادل $C(B-A)$ و مساحت زیر تابع چگالی $f(x)$ مساوی یک واحد است، احتمال پذیرش یک نقطه (احتمال موفقیت) مساوی $f(x)/C(B-A)$ است. بنابراین هرچه C کوچکتر باشد احتمال موفقیت و کارایی روش بیشتر می‌شود.

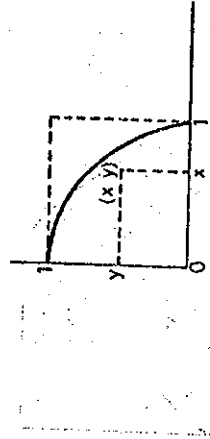
- چنانچه نقاط تولید شده زیر منحنی $f(x)$ را فرض کنیم.
- کل نقاط داخل مستطیل شکل ۶-۱ را N فرض کنیم.
- نسبت $\frac{n}{N}$ تقریباً برابر است با تقسیم سطح زیر منحنی $f(x)$ بر سطح مستطیل شکل ۶-۱:

$$\frac{\int_A^B f(x)dx}{C(B-A)} \approx \frac{n}{N}$$

لذا می‌توان انتگرال $f(x)$ را با نقاط تأیید شده زیر سطح منحنی تقریب زد.

$$\int_A^B f(x)dx \approx \frac{n}{N} C(B-A)$$

با فرض فوق و براساس تعریف عدد π ناحیه $\frac{\pi}{4}$ را در نظر می‌گیریم و می‌خواهیم با استفاده از روش مونت کارلو آن را تقریب بزنیم. تمامی نقاط دارای معادله زیر است: $x^2 + y^2 \leq 1$



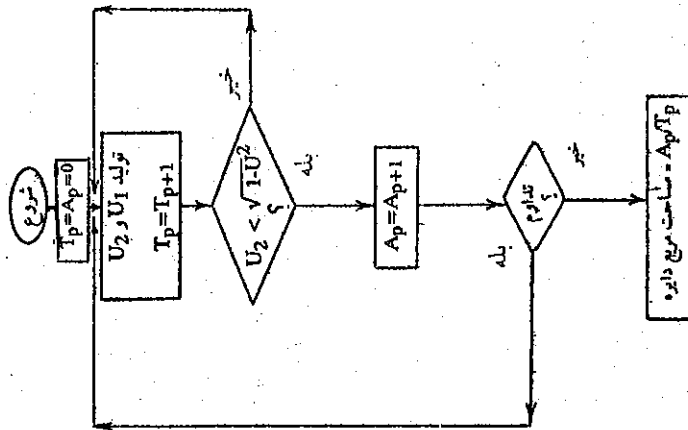
شکل ۶-۲ حل مونت کارلو از π

با استفاده از $x^2 + y^2 \leq 1$ می‌توان نوشت:

می‌توان روال زیر را در نظر گرفت که برای مختصات یک نقطه انجام می‌گیرد:

- ۱- یک جفت عدد تصادفی یکنواخت U_1 و U_2 در محدوده $[0,1]$ تولید می‌شود.
- ۲- شرط $U_2 < \sqrt{1-U_1^2}$ بررسی می‌گردد.
- ۳- چنانچه شرط فوق برقرار نباشد، به مرحله ۱ باز می‌گردد. (جذف)
- ۴- U_1 و U_2 پذیرفته می‌شوند.

مطابق روال فوق یک جفت تصادفی دیگر آزمایش می‌گردد و مانند آن با تولید تصادفی‌های بسیار ناحیه منحنی نیم دایره شکل ۶-۲ تقریب زده می‌شود. الگوریتم آن به صورت زیر است که در آن تعداد جفت اعداد تصادفی تولیدی در T_p و تعداد جفت اعداد تصادفی قبول شده در A_p می‌باشد.



شکل ۶-۳ الگوریتم محاسبه مساحت ربع دایره شکل ۶-۲

بخش چهارم

شبیه‌سازی کامپیوتری

فصل نهم - الگوهای شبیه‌سازی سیستمهای گسترده

فصل دهم - سیستمهای صف

فصل یازدهم - شبیه‌سازی سیستمهای انبار

فصل دوازدهم - شبیه‌سازی شبکه‌های ارزیابی و

تجدید نظر در برنامه‌ها (PERT)

فصل سیزدهم - طراحی و ارزیابی آزمایشات

شبیه‌سازی



فصل نهم

الگوهای شبیه‌سازی سیستمهای گسسته

گفتیم شبیه‌سازی فرایندی از مدلسازی و اجرای مدل است و ابتدا مسئله به یک مدل تبدیل می‌گردد. که مدلهای براساس گذار مانند شبکه پتری به عنوان ابزار قوی مدلسازی ارائه شد. دیدیم که چگونه مسائل مختلف را می‌توان با آن مدلسازی و سپس اجرا کرد. در این بخش به مدلهای انجام شبیه‌سازی سیستمهای گسسته می‌پردازیم و می‌بینیم که چگونه یک برنامه شبیه‌سازی انجام می‌شود. البته لازم به ذکر است که این نگرش در بیان زیانه‌های شبیه‌سازی نیز مد نظر قرار می‌گیرد و اساس تقسیم‌بندی زیانه‌های شبیه‌سازی را تشکیل می‌دهد.

۹.۱- گامهای شبیه‌سازی کامپیوتری

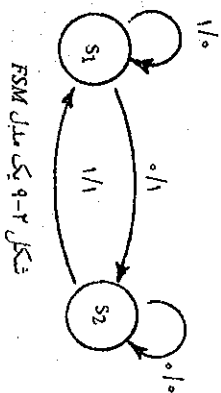
مدل شبیه‌سازی کامپیوتری دارای مراحل بسیاری است که می‌توان آنرا به سه گام اصلی و جدا تقسیم کرد. گامهای حرکت از مرحله آغازین تا ارائه گزارشات موردنظر به شرح زیر است:

- ۱- مرحله شناخت اجزاء (مقداردهی اولیه)
- ۲- مرحله حرکت یا عملیات (تزریق داده/تایج)
- ۳- مرحله گزارشات (ارائه گزارشات)
- مرحله شناخت اجزاء یا تعیین و فراهم آوردن شرائط اولیه جهت شروع مرحله شبیه‌سازی می‌باشد. این گام جهت تعیین اشیاء، متغیرهای سیستم و مقداردهی اولیه آنها است.
- مرحله حرکت یا مرحله دوم شبیه‌سازی جهت ارائه داده‌ها به سیستم و انجام مراحل مربوطه (حلقه اجرایی) و اجرای گام عملیات تکراری است. در این مرحله تمامی عملیات زمان‌بندی، چرخش عملیات و جمع‌آوری آمارهای موردنظر انجام می‌گیرد که از زمان آغازین (زمان صف) تا انتهای مرحله شبیه‌سازی (پایان شبیه‌سازی) عملیات پی‌گیری و دنبال می‌شود.
- مرحله گزارشات یا ارائه خروجی (تهیه گزارشات) مورد نیاز شبیه‌سازی که نتایج در قالب نمودارهای آماری یا جداول عملکردی به نمایش در می‌آید. حتی ممکن است انجام و اجرای عملیات

۹.۲.۱- مدل کودن جریان کنترل

اگر در یک سیستم به دنبال جریان کنترل باشیم، عواملی که جریان کنترل را تغییر می‌دهند پیش آمدها هستند. لذا "پیش آمدها" اساس کار را تشکیل می‌دهند و رخداد آنها مستیر کنترل یا جریان آنرا تحت تأثیر قرار می‌دهد. در این مدل حالتها یا وضعیتهای سیستم و عواملی که روی روند آنها نقش پایه‌ای دارند، تعیین و زمان‌بندی انجام آنها صورت می‌گیرد. لذا در این الگو پیش آمدها را می‌توانیم و آنها را زمان‌بندی می‌کنیم. چون اساس زمان‌بندی پیش آمدها است به آن "روشن زمان‌بندی پیش آمدها" گویند و مدل به "مدل پیش آمدگرا" نامگذاری می‌شود.

مثال ۹-۱ مدل ماشین‌های متناهی (FSM) نگرش بر اساس جریان کنترل را فراهم می‌کند. شکل ۹-۲ یک مدل با ماشین FSM ارائه می‌کند.

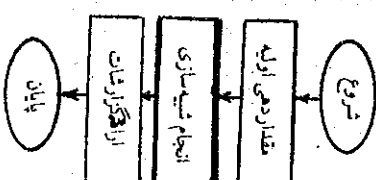


شکل ۹-۲ یک مدل FSM

پیش آمدها "ورود" یا "ورود ۱" هستند. عامل اصلی که از وضعیت S1 به S2 یا بالعکس جریان کنترل را عوض می‌کند، پیش آمدهای ورود "۰" یا "۱" می‌باشند. اگر در وضعیت S1 باشیم، پیش آمد "ورود ۰" به S2 تغییر وضعیت می‌دهد. در وضعیت S2، پیش آمد "ورود ۱" به S1 تغییر وضعیت می‌کند. پیش آمدهایی که تغییر وضعیت ایجاد می‌کنند به "پیش آمدهای اصلی" موسومند. پس برای تبدیل فضای مسئله به مدل FSM وضعیت‌های سیستم را می‌توانیم و پیش آمدها را فورست می‌کنیم و پیش آمدهایی که تغییر وضعیت را موجب می‌شوند علائقگذاری می‌نماییم (تاثیر پیش آمدهای اصلی). پس می‌توان گفت که مدل FSM مدلی است که جریان کنترل را ارائه می‌کند و بررسی‌های سیستم و مدل بر اساس پیش آمدها می‌باشد. این نگرش مدیران عملیاتی (ویانی) در سازمانهای اداری/عملیاتی می‌باشد که به مبادی

روی محیط مجازی اجراء شود (مانند نمایش عملیات اجرایی یک فرودگاه).

شکل ۹-۱ روند مراحل یک برنامه شبیه‌سازی را نشان می‌دهد.



شکل ۹-۱ گامهای یک برنامه شبیه‌سازی

۹.۲- مدل‌های شبیه‌سازی سیستمی گسترده

جهت انجام یک برنامه شبیه‌سازی گسترده الگوهای مختلفی وجود دارد. این الگوها موسوم به مدل‌های شبیه‌سازی گسترده‌اند که می‌توان به سه دسته کلی آنها را تقسیم‌بندی کرد:

- ۱- مدل پیش آمدگرا (event oriented model)
- ۲- مدل پروسه گرا (process oriented model)
- ۳- مدل فعالیت گرا (activity oriented model)

اساس تقسیم‌بندی را نگرش در شیوه پودارزش و انجام عملیات تشکیل می‌دهند. با توجه به آن که در سیستم به دنبال چه می‌گردیم، اصول و روش متفاوت است به دنبال چه هستیم؟ جریان کنترل^(۱)، جریان داده^(۲) و یا هر دو.

پاسخ به این سؤال الگوهای مختلف را پدید آورده است که ابتدا جهت تفهیم آن به شرح مقدمه‌ای می‌پردازیم و مدل‌های گرازیکی را جهت توجه و نگرش موردنظر مدلساز ارائه می‌نماییم. ابتدا مدل‌های مختلف ارائه کننده جریان کنترل، جریان داده و هر دو جریان داده و کنترل را بیان می‌داریم. و سپس فلورچارت‌های شبیه‌سازی هر سه مدل را ترسیم می‌کنیم.

کنترل عملیات توجه دارند و عوامل مؤثر بر جریان کنترل عملیات را تعیین می‌کنند. مثلاً در یک سیستم مکاتبات اداری، جریان مکاتبات مهم است. در این توجه پیش‌آمدهایی که جریان مکاتبات را دستخوش تغییر می‌کند، می‌بایم. پس یک نامه به عنوان موضوع سیستم مکاتبات مهم نیست که شماره شود، ثبت گردد یا مهر امضاء بر روی آن نقش بندها بلکه مسیرهایی که در سیستم مکاتبات طی می‌کند مدنظر قرار می‌گیرد. در این توجه "مبادی کنترل" و "پیش‌آمدهای مؤثر" بر مسیر حرکت یا توقف امور استخراج می‌شوند. "مبادی کنترل" وضعیاتی است که یک نامه می‌تواند داشته باشد. عامل تغییر وضعیتهای رخداد پیش‌آمدها مانند "شروع شماره شدن"، "پایان شماره شدن" "شروع ثبت"، "پایان ثبت" و غیره می‌باشند لذا عامل تعیین کننده "پیش‌آمدها" هستند.

در مثال ۹-۱ می‌توان گفت:

- وضعیت‌های سیستم: s1, s2

- پیش‌آمدهای سیستم: e0 (ورود) و e1 (ورود ۱)

- پیش‌آمدهای اصلی: e0 در e1s1

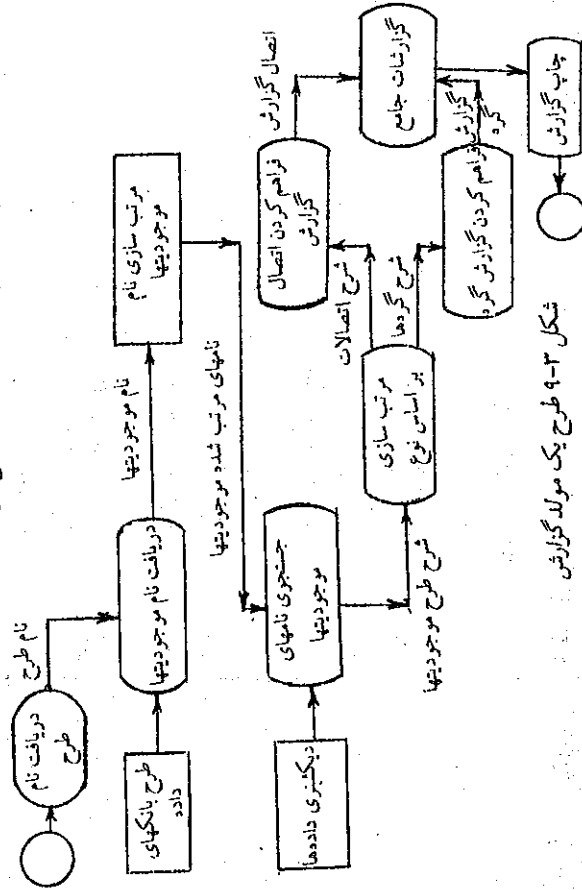
۹-۲-۲- مدل کردن جریان داده

چنانچه در سیستمی به دنبال "جریان داده" باشیم و مسیر عبور اشیاء را تعیین کنیم، عاملی که جریان داده را تغییر می‌دهد پروسه‌ها هستند. در واقع مسیر مدنظر قرار می‌گیرد و عواملی که بر داده مؤثرند و تغییرات آنرا موجب می‌شوند، انتخاب می‌گردند. پس مؤثر در این مدل پروسه‌ها (یا فرآیندها) هستند و آنها را زمان‌بندی می‌کنیم. چون روی داده مجموعه‌ای از پروسه‌ها مؤثرند به این دلیل باید به "زمان‌بندی پروسه‌ها" توجه شود و مدل را "پروسه‌گرا" نامند. پس گرایش اصلی آنرا پروسه تشکیل می‌دهند.

پروسه، مجموعه‌ای از پیش‌آمدها هستند که بطور متوالی یا همزمان باید رخ دهند تا نتیجه‌ای حاصل گردد. مدنظر منابع یا پردازش‌هایی است که عمل پردازشها یا پروسه‌ها را بوجود می‌آورد. این نگرش ارباب رجوع در یک سازمان است که صرفاً به مسیر عبور نامه در یک سیستم مکاتبات توجه دارد. او پروسه‌هایی که روی یک نامه انجام می‌شوند را یاد می‌گیرد و در تشکیلات سازمانی این عملیات را مرتب بررسی می‌کند. چنانچه تمامی پروسه‌ها را انجام یافته ببیند، مکاتبه خویش را پایان یافته تلقی می‌کند. در این مدل نقاطی که شیء را مجدداً فعال می‌کند و فرایند دیگری شکل می‌گیرد، به نقاط باز فعالی معروف است.

مثال ۹-۲-۱ مدل دیاکرام جریان داده DFD نگرشی براساس جریان داده را ارائه می‌کند و شکل ۹-۳ یک

مدل برای طراحی مولد گزارش توسط DFD را بیان می‌دارد.

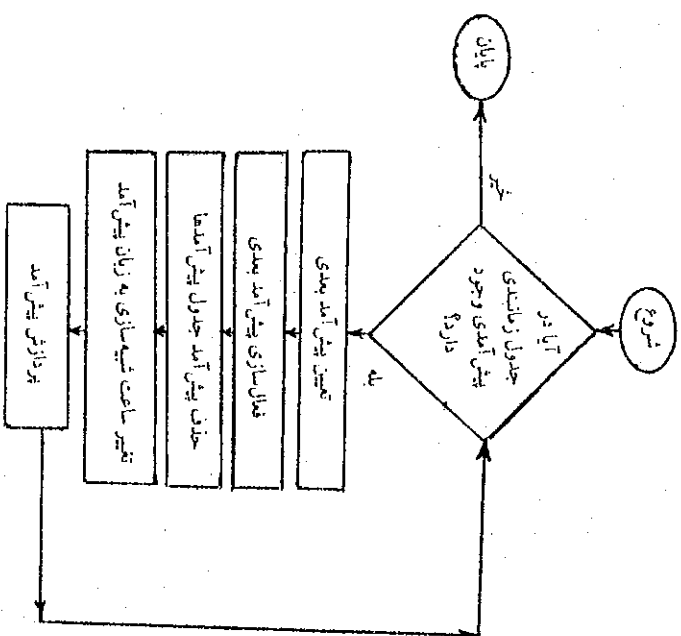


شکل ۹-۳ طرح یک مولد گزارش

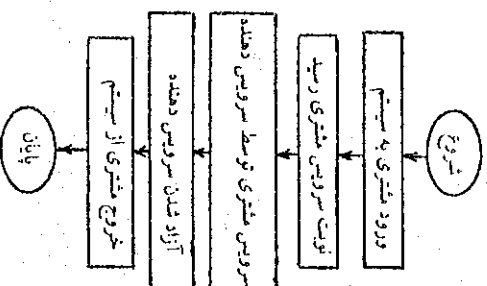
۹-۲-۳- مدل کردن جریان کنترل و داده

زمانی که در یک سیستم به دنبال جریان کنترل و داده باشیم، مدل فعالیت گرا جلوه می‌کند. در این مدل توجه به فعالیت‌های سیستم است و بررسی می‌شود کدام فعالیت قابل انجام است. اگر این فعالیت‌ها یافت شود پردازش آنها شروع می‌گردد و پس از اتمام، این روند برای فعالیت بعدی و اتمال آن تداوم می‌یابد. پس هدف جریان کنترل و جریان داده است و "فعالیتها" به عنوان برقرارکننده این جریان مدنظر قرار می‌گیرد زیرا شروع و اجرای یک فعالیت ممکن است به اجرای فعالیت‌های دیگر (همزمان یا غیرهمزمان) منجر شود.

در این مدل به دنبال یافتن فعالیتها و شرایط انجام آن هستیم. ضمناً اشیاء و منابع مورد نیاز انجام فعالیتها بررسی می‌گردند. اشیایی که باعث تغییر وضعیت در این مدل می‌شوند به اشیاء فعال موسومند. زمان‌بندی مدل روی اشیاء فعال صورت می‌گیرد و زمان تغییر وضعیت مدل براساس عمر این اشیاء است. این مدل معمولاً دید مدیران عالی تشکیلات سازمانی جهت بررسی روند عملیات سازمانی بوجود آمده است که مدل نگرش سازمانی جهت ایجاد و بهبود روشها می‌باشد. در تغییر روشها و بهبود آن ممکن است هر دو جریان کنترل و داده دستخوش تغییر قرار گیرند. لذا فعالیتها



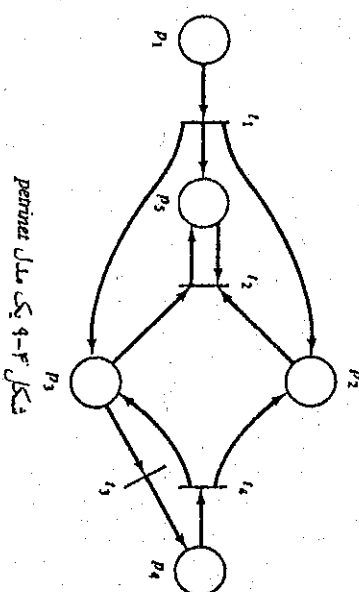
شکل ۵-۹-۵: فلوچارت مدل شبیه‌سازی پیش‌آمد گرا



شکل ۶-۹-۶: فلوچارت مدل پروسه گرا

جدیدی تعریف و مد نظر قرار می‌گیرند که مبادی کنترل و داده را دستخوش تغییر می‌کند.

مثال ۳-۹: مدل شبکه پتری نگارشی براساس جریان داده و کنترل را ارائه می‌نماید و شکل ۴-۹ مثال آنرا نشان می‌دهد. مدلی است براساس گذارد و وضیعت که در بخش مدل‌سازی ارائه گردید.



شکل ۴-۹: یک مدل Petri net

۹.۳-۹-۳: مدل‌های شبیه‌سازی ۹.۳.۱- مدل پیش‌آمد گرا

همانطور که در مدل‌سازی جریان کنترل بیان گردید پیش‌آمدها نقش اساسی دارند. لذا شبیه‌سازی بر این اساس را شبیه‌سازی پیش‌آمد گرا نامند. فلوچارت مدل شبیه‌سازی آن در شکل ۵-۹ نشان داده است که توجه به پیش‌آمدها و جریان زمان حرکت را سبب می‌گردند. این مدل مورد استفاده در بعضی از زبانهای شبیه‌سازی مانند GASP و SIMSCRIPT است.

در این روند ابتدا جدولی از پیش‌آمدها و زمان آنها را داریم و پیش‌آمدها جریان شبیه‌سازی را به زمان آنها جلو می‌بریم که در فصل بعد این مدل با جزئیات بیان می‌گردد.

۹.۳.۲- مدل پروسه گرا

در این مدل مراحلی که یک مشتری از بدو ورود به سیستم تا خروج از آن طی می‌کند، پروسه سرویس مشتری گویند و فرایندی که به ازاء هر مشتری انجام می‌گیرد، مراحل انتظار و سرویس را نیز در بر می‌گیرد. هر کدام از مراحل فلوچارت را یک بلاک گویند. طبق مطالب بیان شده در مدل‌سازی جریان داده عمل می‌کند. مدل مورد استفاده بسیاری از زبانهای شبیه‌سازی مانند GPSS و SIMULA است (شکل ۶-۹).

- ۱- سیستم (system)
 - ۲- حالت سیستم (stat)
 - ۳- فعالیتها (activity)
 - ۴- پدیده‌ها (موضوعات و اشیاء (object & subject)
 - ۵- صفات پدیده (Attribute)
 - ۶- پروسه (process)
 - ۷- پیش آمد (event)
- در واقع عاملی که شاید بطور واضح بیان نشده و به عنوان عامل هشتم محسوب می‌گردد همان "جریان زمان" در نگارش "جهان نگری" است که جهان شیء‌سازی ما را به انتهای فعالیت خود (پریود) می‌رساند. در شیء‌سازی سیستمهای گسسته مهمترین عامل تغییر زمان شیء‌سازی و جریان آن تا انتهای پریود شیء‌سازی می‌باشد. این امر توسط برنامه زمان‌بند صورت می‌گیرد که روشهای مختلفی دارد.

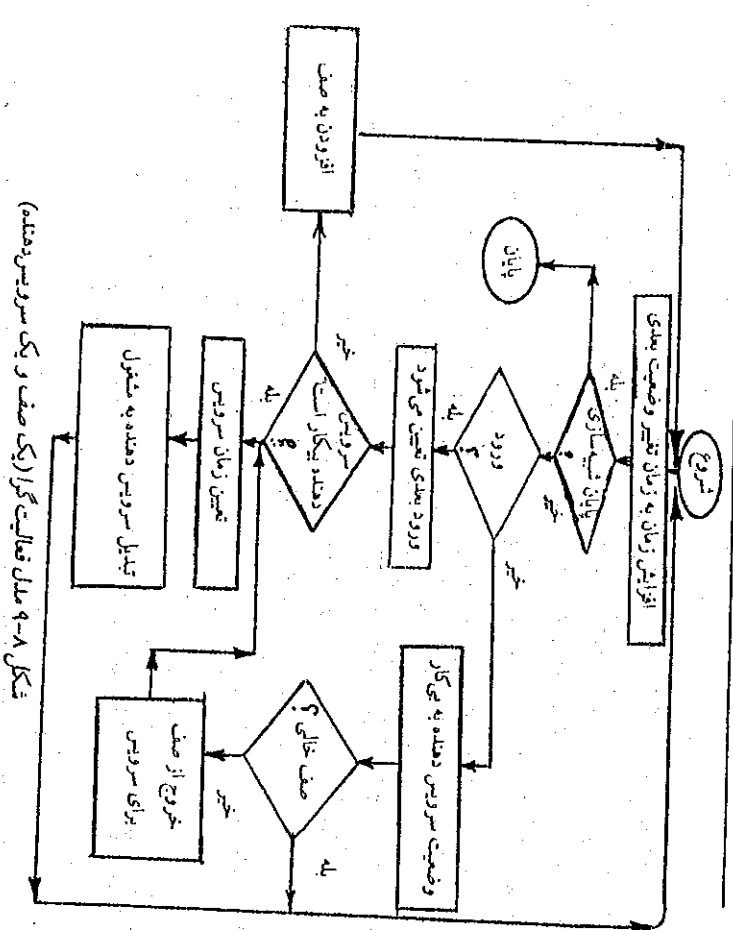
۹.۵- مکانیزم‌های جریان زمان (flow of control)

پروسه شیء‌سازی در یک مدت زمانی صورت می‌گیرد که این "دوره زمانی شیء‌سازی" را پریود شیء‌سازی می‌نامند. در یک سیستم شیء‌سازی باید مکانیزمی وجود داشته باشد که با توجه به عملکرد سیستم از زمان شروع شیء‌سازی به طرف انتهای زمان عملیات حرکت کنیم تا بتوانیم پریود شیء‌سازی را بپوشانیم.

سمبل گذر زمان در شیء‌سازی را ساعت شیء‌سازی (clock) می‌نامند. در بحث "جهان نگری" ساعت آن یک ساعت طبیعی است ولی در شیء‌سازی ممکن است ساعت شیء‌سازی کندتر یا سریعتر حرکت کند. "گذر زمان در پریود شیء‌سازی از طریق ساعت شیء‌سازی تصویر می‌گردد". در واقع جریان یا گذر زمان در ساعت شیء‌سازی با استفاده از گامهای افزایشی صورت می‌گیرد که این برهه‌های زمانی (گامهای زمانی) τ بستگی به بستر اجرای شیء‌سازی دارد ($\Delta t = \tau$)

در گام زمانی شیء‌سازی t دارای دو پارامتر اساسی می‌باشد:

- ۱- ثابت زمانی افزایشی Δt
 - ۲- زمان اجرای دستورات پردازش
- زمان اجرای دستورالعمل معمولاً مدنظر قرار نمی‌گیرد و این امر فرمول را به صورت زیر تبدیل



شکل ۹-۸ مدل فعالیت گرا (یک صف و یک سرور مهندسه)

۹.۶- شیء‌سازی کامپیوتری سیستمهای گسسته

"جهان نگری" اصطلاحی است که برای توصیف ارائه شیء‌سازی به چشم می‌خورد و بیان‌کننده "تصور سیستم مدل شده" است. نگارشی عمیق به سیستم‌ها جهت مدلسازی آنها می‌باشد و می‌گوید: الف - جهان به عنوان مجموعه‌ای از پدیده‌ها (اشیاء و موضوعات) است که به وسیله صفاتشان توصیف می‌گردند.

- ب- پدیده‌ها دارای رفتاری (فعالیتی) می‌باشد که به تناسب شرایط صورت می‌گیرد.
- ج- شرایط را می‌توان به عنوان مجموعه‌ای از پیش آمدها (وقایع) یا محرکها تصور نمود.
- د- پیش آمد و رخداد آن منجر به تغییراتی در وضعیت (حالت) سیستم می‌گردد.

در یک جمله می‌توان چنین بیان نمود: پیش آمدها تحت شرایطی رخ می‌دهند که برانر انجام فعالیتهایی باعث تغییر صفات پدیده‌ها می‌گردند و در نهایت این پروسه عامل تبدیل وضعیت سیستم از حالتی به حالت دیگر می‌شود. در این تعریف مفاهیم زیر توجیه است:

می‌کند. $clock = clock + \tau$ ($\tau = \Delta t$)

در شبیه‌سازی بحث زمان‌بندی یا مدیریت "زمان" مطرح می‌گردد. مدیر زمان یا زمان‌بند ۲ وظیفه اساسی دارد که عبارت از افزایش جریان زمانی یا به هنگام درآوردن آن و دیگری ایجاد همزمانی وقوع پیش آمدها است. زمان‌بند در واقع وظیفه "زمان‌بندی" را بر عهده دارد. چون زمان‌بندی پیش آمدها در کامپیوتر به صورت ترتیبی (یابجی) انجام می‌پذیرد لذا پیش آمدها در یک شبیه‌سازی می‌توانند پی‌درپی رخ دهند. وظیفه زمان‌بند، زمان‌بندی فرایندها و کنترل رخداد آنها است. مکانیزمهای اساسی زمان‌بندی (مدل جریان زمان) در نمو زمان به دو صورت است:

۱- مدل گام ثابت زمانی - (نمو زمانی ثابت)

۲- مدل گام متغیر زمانی - (نمو زمانی متغین)

در مدل گام ثابت زمانی با ثابت زمانی

ثابت Δt حرکت آغاز و تداوم می‌یابد:

| t | State | events |
|--------------------|----------------|---------------------------------|
| = 0 | S ₁ | e ₁ |
| = 0 + Δt | S ₂ | e ₂ , e ₃ |
| = 0 + 2 Δt | S ₃ | e ₄ , e ₅ |
| = 0 + 3 Δt | S ₄ | |
| = 0 + 4 Δt | S ₅ | |
| = 0 + 5 Δt | | e ₆ |

روش ساده‌ای است ولی به علت "زمان مرور یافتن وقایع اتفاقی" در برهه زمانی Δt بروز مشکل می‌کند. وقتی در برهه زمانی Δt هیچ پیش‌آمدی نداشته باشیم زمان مرور تلف می‌شود و سرعت شبیه‌سازی را کاهش می‌دهد. ضمناً انتخاب نامناسب Δt نتایج اشتباهی بدست می‌دهد. چنانچه Δt بزرگ انتخاب شود در گام حرکت به یک یا دو گام بسنده می‌شود و لذا فرمولی برای انتخاب Δt باید بدست آورد. در مدل "پیش‌آمد بعدی" جریان

زمان براساس پیش‌آمد بعدی تعیین می‌شود:

| t | State | events |
|-------------------------|----------------|----------------|
| = 0 | S ₁ | e ₁ |
| = e ₁ . time | S ₂ | e ₂ |
| = e ₂ . time | S ₃ | e ₃ |
| = e ₃ . time | S ₄ | e ₄ |
| = e ₄ . time | S ₅ | e ₅ |
| = e ₅ . time | | e ₆ |

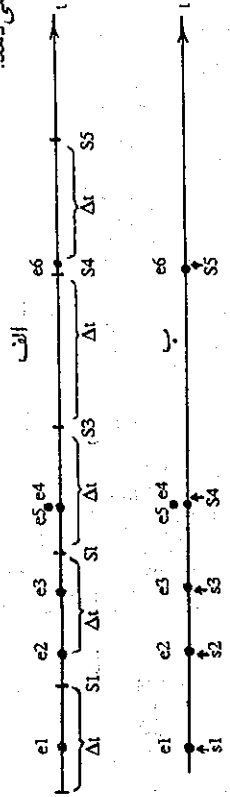
زمان شبیه‌سازی براساس زمان رخداد واقعه بعدی تنظیم می‌گردد و از نظر پیاده‌سازی بسیار مشکل می‌باشد. ولی سرعت شبیه‌سازی را بسیار افزایش می‌دهد و همچنین متداول‌ترین مدل در شبیه‌سازی است. در شکل ۹-۱۰ فلوچارت دو روش گام ثابت و پیش‌آمد بعدی را نشان داده‌است. در روش گام ثابت زمانی یا نمو ثابت زمانی، پرورد شبیه‌سازی به بخشهای مساوی زمانی و با طول ثابت k تقسیم می‌گردد. ساعت شبیه‌سازی با پرورد ثابت زمانی افزایش می‌یابد ($t=k$) و $clock = clock + k$ و در واقع با گامهای ثابت ($t=k$) زمان سپری می‌شود.

در روش گام متغیر یا نمو زمانی متغیر، پرورد شبیه‌سازی به بخش‌های مساوی زمانی تقسیم‌بندی نمی‌شود و ساعت شبیه‌سازی می‌تواند براساس گامهای با طول متغیر اتفاقی (random) حرکت کند. یعنی زمان تصادفی است یا آن‌که گامهای تغییر زمان براساس زمان رخداد پیش‌آمدها تعیین شود.

یعنی زمان پیش‌آمد بعدی است. انتخاب، به صورت تصادفی معمولاً برای جریان شبیه‌سازی صورت نمی‌گیرد و جهت کنترل سیستم‌های مدیریت توسط مدیران کنترل یا بازارسان در سیستم کنترل منابع انسانی کاربرد دارد. در شبیه‌سازی کامپیوتری هنوز استفاده مؤثر و چندانی از آن صورت نگرفته است. البته این امر در قطع زمان شبیه‌سازی کاربرد دارد چون در برنامه‌سازی زبانهای شبیه‌سازی کاربر می‌تواند زمان قطع برنامه شبیه‌سازی را تعیین کند یا هر زمان که می‌خواهد از خارج برنامه شبیه‌سازی آن را قطع نماید. ولی در واقع در پرورد تغییر زمان کاربرد ندارد.

انتخاب، به عنوان زمان پیش‌آمد بعدی به نام "مدل پیش‌آمد بعدی" (next event) موسوم است و در بسیاری از زبانهای شبیه‌سازی از این مدل استفاده می‌شود. پس دو مدل اساسی داریم که یکی مدل گام ثابت و دیگری مدل پیش‌آمد بعدی است. با مثالی این دو روش را بیان می‌کنیم.

پیش‌آمدهای یک سیستم به ترتیب: $e_1, e_2, e_3, e_4, e_5, e_6$ فرض می‌شود. شکل ۹-۹ جریان زمان در ۲ مدل را نشان می‌دهد:



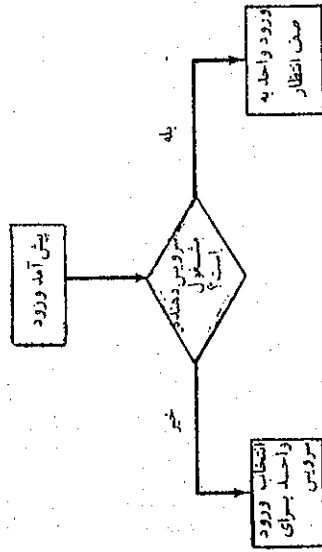
شکل ۹-۹ جریان زمان (الف) مدل با گام ثابت (ب) مدل پیش‌آمد بعدی

۹-۱- مثالهای شبیه‌سازی

هدف این بخش ارائه چندین مثال شبیه‌سازی است که مستقیماً و بدون کامپیوتر قابل انجام هستند. این مثالها نگرشی به مدل‌ریزی شبیه‌سازی سیستم‌های گسسته است و بعضی ملاحظات و تحلیل‌های مربوطه را در اختیار شبیه‌سازی قویار می‌دهد. با انجام این مثالها خواننده نکات و موارد مورد نیاز در فصول آینده را مورد توجه بیشتری قرار خواهد داد که به منظور شبیه‌سازی، طی سه گام زیر لازم است:

۱- تعیین ویژه گیهای هر ورودی شبیه‌سازی که به صورت توزیعهای آماری پیوسته یا گسسته قابل ارائه می‌باشند.

دومین پیش‌آمد وقتی رخ می‌دهد که یک واحد به سیستم وارد شود و دیاگرام جریان این حالت در شکل ۹-۱۲ نشان داده شده است. واحد ورودی ممکن است سرویس دهنده را به کار یا مشغول بیاورد. لذا در حالتی که وارد می‌شود و سرویس دهنده مشغول است برای سرویس به صف انتظار می‌رود. در غیر آن فعالیت سرویس آن آغاز می‌گردد.



شکل ۹-۱۲ دیاگرام جریان ورود واحد به سیستم

اعمال را در شکل ۹-۱۳ نشان داده‌ایم. اگر سرویس دهنده مشغول باشد واحد به صف وارد می‌شود ولی اگر بی‌کار (آزاد) باشد، واحد به سرویس دهنده (شروع سرویس) وارد می‌شود غیر ممکن است که سرویس دهنده آزاد باشد و صف خالی نباشد.

| وضعیت صف | | | |
|-------------------|------------|-------|-------------------|
| خالی | غیر خالی | مشغول | وضعیت سرویس دهنده |
| ورود به صف | ورود به صف | آزاد | دورود سرویس دهنده |
| دورود سرویس دهنده | غیر ممکن | | |

شکل ۹-۱۳ عملیات منقضي به ازای ورود یک ورودی

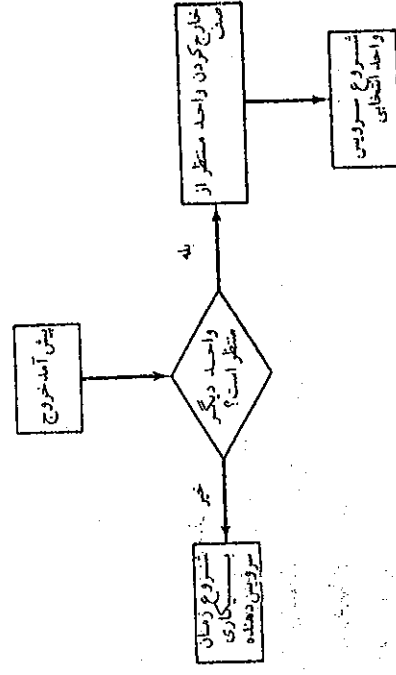
پس از تمام شدن یک سرویس، سرویس دهنده ممکن است آزاد شود یا به وسیله واحد بعدی مشغول بماند. ارتباط این نتایج و وضعیت صف در شکل ۹-۱۴ ارائه شده است. اگر صف خالی نیست واحد دیگری می‌تواند به سرویس دهنده وارد شود و لذا مشغول بماند. اگر صف خالی است، سرویس دهنده پس از اتمام یک سرویس آزاد خواهد ماند. غیر ممکن است اگر صف خالی است پس از اتمام سرویس، سرویس دهنده مشغول بماند. بطور مشابه آزاد بودن سرویس دهنده پس از اتمام یک

خارج می‌شوند. بعلاوه، اغلب زمانهای سرویس با طول تصادفی نیز از توزیع‌های آماری پیروی می‌کند که در طول زمان تغییر ناپذیرند. در اینجا گنجایش سیستم نامحدود است. ورودیها ممکن است برای سرویس از صف براساس سیاست خاصی (مانند FIFO) توسط یک سرویس دهنده یا کانال برگزیده شوند.

ورودیها و سرویس‌ها به وسیله توزیع‌های زمان بین ورودیها و زمان سرویس بیان می‌شوند. نرخ مؤثر ورودیهای باید از نرخ ماکزیمم سرویس کمتر باشد و گرنه طول خط انتظار بدون محدودیت رشد می‌کند. وقتی که صفها بدون محدودیت رشد وضعیت بی‌ثباتی پیش می‌آید. اگر نرخ ورودیها بیشتر از زمان سرویسها باشد، رشد نمایی می‌دهد.

اولیتهای مهم در فراهم آوردن شرائط شبیه‌سازی سیستم صف عبارت است از مفاهیم وضعیت سیستم، پیش‌آمدها و ساعت شبیه‌سازی وضعیت سیستم تعداد واحدهای سیستم و حالتیهای سرویس دهنده (یعنی بی‌کار یا مشغول) می‌باشد. یک پیش‌آمد مجموعه شرائطی است که سبب تغییر وضعیت سیستم می‌شود. در یک صف یا یک کانال فقط دو پیش‌آمد ممکن وضعیت سیستم را عوض می‌کند. این پیش‌آمدها عبارتند از: ورود یک واحد به سیستم (پیش‌آمد ورود) و اتمام سرویس یک واحد (پیش‌آمد خروج). سیستم صف شامل سرویس دهنده برای سرویس یک واحد (اگر یکی سرویس داده شود) و واحدهای منتظر داخل صف می‌باشد.

اگر سرویس کامل شده باشد، شبیه‌سازی از طریق دیاگرام جریان شکل ۹-۱۱ انجام می‌شود. توجه کنید که سرویس دهنده فقط دو وضعیت ممکن در شکل ۹-۱۲ یعنی بی‌کار یا مشغول را دارد.



شکل ۹-۱۱ دیاگرام جریان سرویس کامل شده

| زمان بین ورود | زمان سروس | مشتری |
|---------------|-----------|-------|
| 0 | 2 | 1 |
| 2 | 2 | 2 |
| 6 | 4 | 3 |
| 7 | 1 | 4 |
| 9 | 2 | 5 |
| 15 | 6 | 6 |

جدول ۹-۲ زمانهای بین ورود و ساعت شبیه‌سازی

زمانهای سروس ممکن یک، دو، سه و چهار واحد زمانی است. فرض کنید که تمامی ۴ مقدار شانس وقوع یکسان دارند.

| زمان سروس | مشتری |
|-----------|-------|
| 2 | 1 |
| 1 | 2 |
| 3 | 3 |
| 2 | 4 |
| 1 | 5 |
| 4 | 6 |

جدول ۹-۳ زمانهای سروس

حالا زمانهای ورود و سروس برای شبیه‌سازی سیستم صف با یک کانال، مخلوط می‌گردند. در جدول ۹-۴ اولین مشتری در ساعت زمانی صفر وارد می‌شود و بلافاصله سروس می‌یابد که دو دقیقه نیاز دارد. سروس او در ساعت زمانی ۲ پایان می‌یابد. دومین مشتری در ساعت زمانی ۲ وارد ساعت زمانی ۳ سروسش پایان می‌یابد. توجه این که چهارمین مشتری در ساعت زمانی ۷ وارد می‌شود ولی تا ساعت زمانی ۹ سروسش شروع نمی‌شود زیرا مشتری ۴ تا ساعت زمانی ۹ سروسش تمام نشده است.

جدول ۹-۴ بطور خاص برای یک صف با یک کانال و با سیاست سروس اولین ورودی - اولین خروجی (FIFO) طراحی شده است. این جدول ساعت زمانی که هر پیش‌آمد رخ می‌دهد را ردیفی کرده است. ستون دوم جدول ساعت زمانی هر پیش‌آمد ورود را ثبت می‌کند و ستون آخر ساعت زمانی هر پیش‌آمد خروج را در بر دارد. رخدادهای هر دو نوع پیش‌آمدها در جدول ۹-۵ و شکل ۹-۱۵ نشان داده شده است.

سروس وقتی که صف خالی نیست، غیرممکن است.

| وضعیت صف | تایم سروس | |
|----------|-----------|-----------|
| | مغفول | تایم سروس |
| خالی | غیرممکن | غیرممکن |
| غیرممکن | غیرممکن | غیرممکن |
| غیرممکن | غیرممکن | غیرممکن |

شکل ۹-۱۴ تایم سروس دهنده بعد از اتمام سروس

حالا چگونگی پیش‌آمدهای بیان شده در زمان شبیه‌سازی می‌تواند رخ دهد؟ معمولاً شبیه‌سازی یک سیستم صف نیاز دارد که یک فهرست پیش‌آمدها (event list) را برای تعیین رخ داد بعدی نگهداری کند. فهرست پیش‌آمدها زمانی را برای پیش‌آمدهای هر واحد در سیستم صف تعیین می‌کند. زمانها مطابق مقیاس ساعت شبیه‌سازی نگهداری و ارائه می‌شوند. در شبیه‌سازی، پیش‌آمدها به صورت تصادفی رخ می‌دهند زیرا ماهیت رخدادها در دنیای این چنین است.

تصادفی بودن نیاز به اجازه امکان استفاده از "اعداد تصادفی" دارد. اعداد تصادفی بطور یکتواخت (۰ و ۱) توزیع شده‌اند و بین (۰، ۱) مستقل رخ می‌دهند. ارقام تصادفی بطور یکتواخت در سری (۹ و ۱۰) توزیع می‌شوند. از ارقام تصادفی برای تعیین اعداد تصادفی استفاده می‌شود که تعداد مقتضی ارقام برای هر عدد انتخاب و سپس یک نقطه اعشاری در سمت چپ مقدار انتخابی قرار می‌گیرد. تعداد ارقام مقتضی بستگی به داده ورودی ما دارد. اگر توزیع ورودیها متفاوتی با دو رقم اعشار دارند، دو رقم از جدول ارقام تصادفی انتخاب می‌شود و نقطه اعشاری سمت چپ آن قرار می‌گیرد تا عدد تصادفی را تشکیل دهد.

در یک سیستم صف با یک کانال زمانهای بین ورود و زمانهای سروس با این توزیع این مشتریهای تصادفی تولید می‌شود. این مثالها چگونگی تولید زمانها را نشان می‌دهد و برای سادگی فرض می‌کنیم که مدت زمان بین وردها تولید شده است. جدول ۹-۲ شامل پنج زمان بین ورود تولید شده به این شیوه می‌باشد. این پنج زمان برای محاسبه زمانهای ورود شش مشتری به یک صف استفاده می‌شود. اولین ورودی در ساعت زمانی صفر وارد می‌شود. این ساعت شروع عملیات است، مشتری دوم دو واحد زمانی بعد یعنی در ساعت زمانی ۲ وارد می‌شود. سوبین مشتری ۴ واحد زمانی بعد از دومی یعنی در ساعت زمانی ۶ وارد می‌گردد و مانند آن. دومین زمان موردنظر زمان سروس است و جدول ۹-۳ شامل زمان سروس‌ها می‌باشد که به صورت تصادفی تولید شده‌اند.

در جدول ۹-۵ سلسله پیش‌آمدها براساس هدف شبیه‌سازی پیش‌آمد گسسته بیان شده است. مثال ۹-۱ منطق بالا را دنبال می‌کند و مثال ۹-۲ با یک سیستم صف با ۲ کانال است. دیاگرام‌های جریان یک سیستم صف چند کاناله با سیستم یک کاناله تفاوت دارد.

مثال ۹-۲ - صف با یک سرویس‌دهنده
یک فروشگاه کوچک فقط دارای یک صندوقدار است و مشتریان به این صندوق با فواصل زمانی از ۸ تا ۸ دقیقه بطور تصادفی مراجعه می‌کنند.

هر مقدار ممکن زمان بین ورود، احتمال وقوع یکسان مطابق جدول ۹-۶ دارند. زمان سرویس تصادفی از ۱ تا ۶ دقیقه در جدول ۹-۷ نشان داده شده است. دو ستون آخر جدول ۹-۷ و ۹-۶ مبدأ شرح داده خواهد شد. با زمان ورود و سرویس ۲۰ مشتری شبیه‌سازی، انجام می‌گیرد با یک فروشگاه خالی شروع می‌شود.

| حوزه رقم تصادفی | احتمال نسبی | احتمال | زمان بین ورود (دقیقه) |
|-----------------|-------------|--------|-----------------------|
| 001-125 | 0.125 | 0.125 | 1 |
| 126-250 | 0.250 | 0.125 | 2 |
| 251-375 | 0.375 | 0.125 | 3 |
| 376-500 | 0.500 | 0.125 | 4 |
| 501-625 | 0.625 | 0.125 | 5 |
| 626-750 | 0.750 | 0.125 | 6 |
| 751-875 | 0.875 | 0.125 | 7 |
| 876-000 | 1.000 | 0.125 | 8 |

جدول ۹-۶ توزیع زمان بین ورودها

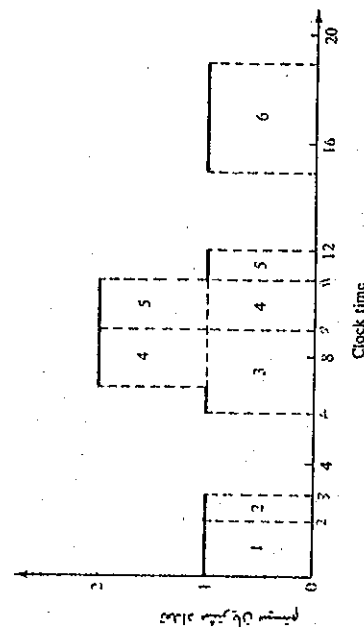
به یک سری اعداد تصادفی با توزیع یکتواخت جهت تولید مراجعات به صندوق نیاز می‌باشد. اعداد تصادفی خواص زیر را دارد:

| پایان زمان سرویس (ساعت) | زمان سرویس (ساعت) | زمان شروع سرویس (ساعت) | زمان ورود (ساعت) | شماره مشتری |
|-------------------------|-------------------|------------------------|------------------|-------------|
| 2 | 2 | 0 | 0 | 1 |
| 3 | 1 | 2 | 2 | 2 |
| 9 | 3 | 6 | 6 | 3 |
| 11 | 2 | 9 | 7 | 4 |
| 12 | 1 | 11 | 9 | 5 |
| 19 | 4 | 15 | 15 | 6 |

جدول ۹-۴ جدول شبیه‌سازی براساس ساعت شبیه‌سازی

| ساعت زمانی | کساره مشتری | نوع پیش آمد |
|------------|-------------|-------------|
| 0 | 1 | ورود |
| 2 | 1 | خروج |
| 2 | 2 | ورود |
| 3 | 2 | خروج |
| 6 | 3 | ورود |
| 7 | 4 | ورود |
| 9 | 3 | خروج |
| 9 | 5 | ورود |
| 11 | 4 | خروج |
| 12 | 5 | خروج |
| 15 | 6 | ورود |
| 19 | 6 | خروج |

جدول ۹-۵ سلسله زمانی پیش‌آمدها



شکل ۹-۱۵ نمودار تعداد مشتریان سیستم

شرح داده شده فوق و با استفاده از جدول ۹-۷ تولید شده است.

| شتری | ارقام تصادفی | زمان سروس (دقیقه) | شتری | ارقام تصادفی | زمان سروس (دقیقه) |
|------|--------------|-------------------|------|--------------|-------------------|
| 1 | 84 | 4 | 11 | 32 | 3 |
| 2 | 10 | 1 | 12 | 94 | 5 |
| 3 | 74 | 4 | 13 | 79 | 4 |
| 4 | 53 | 3 | 14 | 05 | 1 |
| 5 | 17 | 2 | 15 | 79 | 5 |
| 6 | 79 | 4 | 16 | 84 | 4 |
| 7 | 91 | 5 | 17 | 52 | 3 |
| 8 | 67 | 4 | 18 | 55 | 3 |
| 9 | 89 | 5 | 19 | 30 | 2 |
| 10 | 38 | 3 | 20 | 50 | 3 |

جدول ۹-۹ زمانهای سروس تولید شده

زمان سروس اولین مشتری ۴ دقیقه است زیرا عدد تصادفی ۸۴ در طبقه ۸۵-۹۱ جدول ۹-۷ قرار گرفته است.

اساس روش شبیهسازی را جدول شبیهسازی تشکیل می دهد. این جدول برای پاسخ به سئوالات ساختاری طراحی شده است. جدول شبیهسازی برای این مسئله در جدول ۹-۱۰ نشان داده شده است و یک توسعه از جدول مشاهده شده ۴-۹ می باشد. اولین مشتری در زمان ۰ وارد و پس از آن سروس شروع می گردد که در دقیقه ۴ خاتمه می یابد. مشتری برای ۴ دقیقه در سیستم بود. دومین مشتری در دقیقه ۸ وارد شد بنابراین (صندوقدار) سروس دهنده ۴ دقیقه بقیه کار بود. مشابه آن چهارمین مشتری در دقیقه ۱۵ وارد می شود ولی تا دقیقه ۱۸ نمی تواند سروس ببیند. این مشتری در صف ۳ دقیقه منتظر بوده است. این پروسه برای ۲۰ مشتری ادامه می یابد. کل اینها برای نمایش زمانهای سروس، زمان صرف شده مشتری در سیستم، زمان بیکاری سروس دهنده و زمان انتظار مشتریان دو صف انجام گرفته است.

برخی نتایج از شبیهسازی فوق عبارتند از:

۱- میانگین زمان انتظار هر مشتری ۲/۸ دقیقه است. که به صورت زیر تعیین می شود:

$$\text{دقیقه } ۲/۸ = \frac{۵۶}{۲۰} = \frac{\text{مجموع زمان انتظار مشتریان در صف (به دقیقه)}}{\text{میانگین زمان انتظار مشتری}}$$

کل مشتریان

| حوزه رقم تصادفی | احتمال تجمعی | احتمال | زمان سروس (دقیقه) |
|-----------------|--------------|--------|-------------------|
| 01-10 | 0.10 | 0.10 | 1 |
| 11-30 | 0.30 | 0.20 | 2 |
| 31-60 | 0.60 | 0.30 | 3 |
| 61-85 | 0.85 | 0.25 | 4 |
| 86-95 | 0.95 | 0.10 | 5 |
| 96-00 | 1.00 | 0.05 | 6 |

جدول ۹-۷ توزیع زمان سروس

۱- سری اعداد تصادفی دارای توزیع یکپارخت بین ۰ و ۱ دارند.

۲- اعداد تصادفی مستقل هستند.

دو ستون آخر جدول ۹-۶ و ۹-۷ برای تولید زمان تصادفی سروس و زمان بین ورود به کار می رود و در ستون قبل آنها احتمال تجمعی توزیع را در بر دارند. حوزه ارقام تصادفی در جدول ۹-۶ می رقت است که احتمال یک زمان بین ورود یک دقیقه ای، ۱/۲۵ است ۱۲۵ تا از ۱۰۰ رقم تصادفی مقدار گذاری می گردید. زمان بین ورود به ازای ۲۰ مشتری در جدول ۹-۸ آمده است. توجه اینکه اولین عدد تصادفی ۹۱۳ است. چون عدد تصادفی ۹۱۳ در محدوده ردیف هشتم جدول ۹-۶ قرار دارد لذا از ستون اول جدول ۹-۶ زمان ۸ دقیقه بدست می آید.

| شتری | ارقام تصادفی | زمان بین ورودها (دقیقه) | شتری | ارقام تصادفی | زمان بین ورودها (دقیقه) |
|------|--------------|-------------------------|------|--------------|-------------------------|
| 1 | — | — | 11 | 109 | 1 |
| 2 | 913 | 8 | 12 | 093 | 1 |
| 3 | 727 | 6 | 13 | 607 | 5 |
| 4 | 015 | 1 | 14 | 738 | 6 |
| 5 | 948 | 8 | 15 | 359 | 3 |
| 6 | 309 | 3 | 16 | 888 | 8 |
| 7 | 922 | 8 | 17 | 106 | 1 |
| 8 | 753 | 7 | 18 | 212 | 2 |
| 9 | 235 | 2 | 19 | 493 | 4 |
| 10 | 302 | 3 | 20 | 535 | 5 |

جدول ۹-۸ تعیین زمان بین ورودها

زمان سروس مشتریان در جدول ۹-۶ نشان داده شده است. زمان این سروسها بر اساس مدلوی

۲- احتمال اینکه یک مشتری در صف انتظار بکشد $0/65$ است که از طریق زیر تعیین می شود:

$$\text{احتمال (انتظار)} = \frac{\text{تعداد مشتریان منتظر}}{\text{کل مشتریان}} = \frac{13}{20} = 0/65$$

۳- درصد زمان بی کاری سروس دهنده $0/21$ است که به روش زیر تعیین می شود:

$$\text{احتمال اشتغال سروس دهنده} = \frac{\text{مجموع زمان بی کاری سروس دهنده (دقیقه)}}{\text{کل زمان شبیهسازی (دقیقه)}} = \frac{18}{86} = 0/21$$

۴- میانگین زمان سروس $3/4$ دقیقه است که به صورت زیر تعیین می شود:

$$\text{میانگین زمان سروس} = \frac{\text{مجموع زمان سروس (دقیقه)}}{\text{تعداد کل مشتریان}} = \frac{68}{20} = 3/4 \text{ دقیقه}$$

این نتیجه می تواند با امید ریاضی زمان سروس به وسیله میانگین توزیع زمان سروس از طریق زیر مقایسه شود:

$$E(S) = \sum_{s=0}^{\infty} SP(s)$$

با استفاده از جدول ۹-۷

$$\text{دقیقه } 3/2 = (0/05) + (0/10) + 3(0/3) + 4(0/25) + 1(0/10) + 2(0/20) = 1 \text{ امید زمان سروس}$$

مقدار امید ریاضی زمان سروس کمتر از میانگین زمان سروس شبیهسازی است که با افزایش زمان به امید ریاضی نزدیک می شود.

۵- میانگین زمان بین ورود $4/3$ دقیقه است که به روش زیر تعیین می شود:

$$\text{دقیقه } 4/3 = \frac{\text{مجموع تمامی زمانهای بین ورود (دقیقه)}}{\text{تعداد ورودها}} = \frac{82}{19} = 4/3$$

این نتیجه با امید ریاضی زمان رسیدن از طریق یافتن میانگین توزیع یکنواخت $a=1$ و $b=8$ مقایسه می شود.

$$E(A) = \frac{a+b}{2} = \frac{1+8}{2} = 4/5 \text{ دقیقه}$$

امید ریاضی میانگین زمان بین ورود بیشتر از میانگین شبیهسازی است در صورت افزایش زمان شبیهسازی به امید ریاضی نزدیک می گردد.

۶- میانگین زمان انتظار $4/3$ دقیقه می باشد که به طریق زیر معین می شود:

$$\text{دقیقه } 4/3 = \frac{\text{مجموع زمان انتظار مشتریان در صف (دقیقه)}}{\text{کل مشتریان منتظر در صف}} = \frac{52}{13} = 4/3$$

۷- میانگین زمان صرف شده یک مشتری در سیستم $6/2$ دقیقه است که به طریق زیر معین می گردد:

$$\text{دقیقه } 6/2 = \frac{\text{مجموع زمان صرف شده مشتریان در سیستم (دقیقه)}}{\text{تعداد کل مشتریان}} = \frac{124}{20} = 6/2$$

در سیستم (دقیقه)

دومین راه محاسبه این نتیجه تشخیص ارتباط زیر است:

$$\text{میانگین زمان صرف شده} + \text{میانگین زمان صرف شده مشتری} = \text{میانگین زمان صرف شده مشتری با یک سروس (دقیقه)} + \text{بابت انتظار در صف (دقیقه)}$$

$$= 2/8 + 3/4 = 6/2 \text{ دقیقه}$$

(دقیقه) شبیه سازی صف سه سازه جدول ۹-۱۰ جدول شبیه سازی

| زمان بین ورود | زمان ورود | زمان سروس | زمان انتظار | زمان پایان سروس | زمان صرف شده مشتری در سیستم | زمان میانگین سروس |
|---------------|-----------|-----------|-------------|-----------------|-----------------------------|-------------------|
| 20 | 1 | 1 | 0 | 4 | 4 | 0 |
| 19 | 2 | 8 | 1 | 15 | 15 | 8 |
| 18 | 3 | 6 | 1 | 10 | 10 | 6 |
| 17 | 4 | 3 | 1 | 8 | 8 | 3 |
| 16 | 5 | 3 | 1 | 7 | 7 | 3 |
| 15 | 6 | 5 | 1 | 9 | 9 | 5 |
| 14 | 7 | 3 | 1 | 8 | 8 | 3 |
| 13 | 8 | 1 | 1 | 6 | 6 | 1 |
| 12 | 9 | 3 | 1 | 7 | 7 | 3 |
| 11 | 10 | 3 | 1 | 8 | 8 | 3 |
| 10 | 11 | 3 | 1 | 7 | 7 | 3 |
| 9 | 12 | 3 | 1 | 8 | 8 | 3 |
| 8 | 13 | 6 | 1 | 10 | 10 | 6 |
| 7 | 14 | 5 | 1 | 9 | 9 | 5 |
| 6 | 15 | 3 | 1 | 8 | 8 | 3 |
| 5 | 16 | 3 | 1 | 7 | 7 | 3 |
| 4 | 17 | 3 | 1 | 8 | 8 | 3 |
| 3 | 18 | 3 | 1 | 7 | 7 | 3 |
| 2 | 19 | 3 | 1 | 8 | 8 | 3 |
| 1 | 20 | 3 | 1 | 7 | 7 | 3 |
| 0 | 21 | 3 | 1 | 8 | 8 | 3 |
| 0 | 22 | 3 | 1 | 7 | 7 | 3 |
| 0 | 23 | 3 | 1 | 8 | 8 | 3 |
| 0 | 24 | 3 | 1 | 7 | 7 | 3 |
| 0 | 25 | 3 | 1 | 8 | 8 | 3 |
| 0 | 26 | 3 | 1 | 7 | 7 | 3 |
| 0 | 27 | 3 | 1 | 8 | 8 | 3 |
| 0 | 28 | 3 | 1 | 7 | 7 | 3 |
| 0 | 29 | 3 | 1 | 8 | 8 | 3 |
| 0 | 30 | 3 | 1 | 7 | 7 | 3 |
| 0 | 31 | 3 | 1 | 8 | 8 | 3 |
| 0 | 32 | 3 | 1 | 7 | 7 | 3 |
| 0 | 33 | 3 | 1 | 8 | 8 | 3 |
| 0 | 34 | 3 | 1 | 7 | 7 | 3 |
| 0 | 35 | 3 | 1 | 8 | 8 | 3 |
| 0 | 36 | 3 | 1 | 7 | 7 | 3 |
| 0 | 37 | 3 | 1 | 8 | 8 | 3 |
| 0 | 38 | 3 | 1 | 7 | 7 | 3 |
| 0 | 39 | 3 | 1 | 8 | 8 | 3 |
| 0 | 40 | 3 | 1 | 7 | 7 | 3 |
| 0 | 41 | 3 | 1 | 8 | 8 | 3 |
| 0 | 42 | 3 | 1 | 7 | 7 | 3 |
| 0 | 43 | 3 | 1 | 8 | 8 | 3 |
| 0 | 44 | 3 | 1 | 7 | 7 | 3 |
| 0 | 45 | 3 | 1 | 8 | 8 | 3 |
| 0 | 46 | 3 | 1 | 7 | 7 | 3 |
| 0 | 47 | 3 | 1 | 8 | 8 | 3 |
| 0 | 48 | 3 | 1 | 7 | 7 | 3 |
| 0 | 49 | 3 | 1 | 8 | 8 | 3 |
| 0 | 50 | 3 | 1 | 7 | 7 | 3 |
| 0 | 51 | 3 | 1 | 8 | 8 | 3 |
| 0 | 52 | 3 | 1 | 7 | 7 | 3 |
| 0 | 53 | 3 | 1 | 8 | 8 | 3 |
| 0 | 54 | 3 | 1 | 7 | 7 | 3 |
| 0 | 55 | 3 | 1 | 8 | 8 | 3 |
| 0 | 56 | 3 | 1 | 7 | 7 | 3 |
| 0 | 57 | 3 | 1 | 8 | 8 | 3 |
| 0 | 58 | 3 | 1 | 7 | 7 | 3 |
| 0 | 59 | 3 | 1 | 8 | 8 | 3 |
| 0 | 60 | 3 | 1 | 7 | 7 | 3 |
| 0 | 61 | 3 | 1 | 8 | 8 | 3 |
| 0 | 62 | 3 | 1 | 7 | 7 | 3 |
| 0 | 63 | 3 | 1 | 8 | 8 | 3 |
| 0 | 64 | 3 | 1 | 7 | 7 | 3 |
| 0 | 65 | 3 | 1 | 8 | 8 | 3 |
| 0 | 66 | 3 | 1 | 7 | 7 | 3 |
| 0 | 67 | 3 | 1 | 8 | 8 | 3 |
| 0 | 68 | 3 | 1 | 7 | 7 | 3 |
| 0 | 69 | 3 | 1 | 8 | 8 | 3 |
| 0 | 70 | 3 | 1 | 7 | 7 | 3 |
| 0 | 71 | 3 | 1 | 8 | 8 | 3 |
| 0 | 72 | 3 | 1 | 7 | 7 | 3 |
| 0 | 73 | 3 | 1 | 8 | 8 | 3 |
| 0 | 74 | 3 | 1 | 7 | 7 | 3 |
| 0 | 75 | 3 | 1 | 8 | 8 | 3 |
| 0 | 76 | 3 | 1 | 7 | 7 | 3 |
| 0 | 77 | 3 | 1 | 8 | 8 | 3 |
| 0 | 78 | 3 | 1 | 7 | 7 | 3 |
| 0 | 79 | 3 | 1 | 8 | 8 | 3 |
| 0 | 80 | 3 | 1 | 7 | 7 | 3 |
| 0 | 81 | 3 | 1 | 8 | 8 | 3 |
| 0 | 82 | 3 | 1 | 7 | 7 | 3 |
| 0 | 83 | 3 | 1 | 8 | 8 | 3 |
| 0 | 84 | 3 | 1 | 7 | 7 | 3 |
| 0 | 85 | 3 | 1 | 8 | 8 | 3 |
| 0 | 86 | 3 | 1 | 7 | 7 | 3 |
| 0 | 87 | 3 | 1 | 8 | 8 | 3 |
| 0 | 88 | 3 | 1 | 7 | 7 | 3 |
| 0 | 89 | 3 | 1 | 8 | 8 | 3 |
| 0 | 90 | 3 | 1 | 7 | 7 | 3 |
| 0 | 91 | 3 | 1 | 8 | 8 | 3 |
| 0 | 92 | 3 | 1 | 7 | 7 | 3 |
| 0 | 93 | 3 | 1 | 8 | 8 | 3 |
| 0 | 94 | 3 | 1 | 7 | 7 | 3 |
| 0 | 95 | 3 | 1 | 8 | 8 | 3 |
| 0 | 96 | 3 | 1 | 7 | 7 | 3 |
| 0 | 97 | 3 | 1 | 8 | 8 | 3 |
| 0 | 98 | 3 | 1 | 7 | 7 | 3 |
| 0 | 99 | 3 | 1 | 8 | 8 | 3 |
| 0 | 100 | 3 | 1 | 7 | 7 | 3 |

| زمان دو صف | زمان سرویس | شروع | آشپز | | | پیشخدمت | | | ارقام تصادفی بین ورود | مشتري |
|------------|------------|------|------------|-------------|------|------------|-------------|------|-----------------------|-------|
| | | | زمان سرویس | پایان سرویس | شروع | زمان سرویس | پایان سرویس | زمان | | |
| 0 | | | | | 5 | 5 | 0 | 95 | 0 | 1 |
| 0 | 5 | | 3 | | 9 | 3 | 6 | 21 | 2 | 2 |
| 0 | | | | 5 | 15 | 5 | 10 | 51 | 4 | 3 |
| 0 | 18 | | 6 | | 12 | | 12 | 92 | 4 | 4 |
| 1 | | | | | 18 | 3 | 15 | 89 | 2 | 5 |
| 1 | 27 | | 4 | | 20 | 2 | 17 | 38 | 2 | 6 |
| 0 | | | | | 24 | 4 | 20 | 13 | 3 | 7 |
| 0 | 32 | | | | 27 | 3 | 24 | 61 | 3 | 8 |
| 0 | | | | 23 | 30 | | 27 | 50 | 3 | 9 |
| 1 | 35 | | 3 | | 35 | 5 | 30 | 49 | 1 | 10 |
| 0 | | | | | 39 | 3 | 31 | 39 | 2 | 11 |
| 0 | 39 | | 4 | | 43 | | 33 | 53 | 2 | 12 |
| 0 | | | | | 45 | 4 | 35 | 88 | 2 | 13 |
| 2 | 45 | | 5 | | 48 | | 37 | 01 | 1 | 14 |
| 0 | | | | | 51 | 2 | 40 | 81 | 2 | 15 |
| 1 | 51 | | 3 | | 52 | 4 | 42 | 53 | 2 | 16 |
| 0 | | | | | 56 | 4 | 44 | 81 | 2 | 17 |
| 0 | 56 | | 5 | | 57 | | 48 | 64 | 3 | 18 |
| 0 | | | | | 62 | 6 | 51 | 01 | 2 | 19 |
| 1 | 62 | | 6 | | 62 | | 54 | 67 | 2 | 20 |
| 0 | | | | | 62 | 3 | 55 | 01 | 4 | 21 |
| 0 | | | | | 62 | 3 | 57 | 47 | 1 | 22 |
| 1 | | | | | 62 | 3 | 59 | 75 | 2 | 23 |
| 0 | | | | | 62 | 3 | 59 | 57 | 3 | 24 |
| 0 | | | | | 62 | 3 | 59 | 87 | 1 | 25 |
| 0 | | | | | 62 | 3 | 59 | 47 | 4 | 26 |
| 11 | | | 43 | | 56 | | | | | |

جدول ۹-۱۴ شبیه‌سازی

تصمیم‌گیری یا استفاده از نتایج فوق ممکن است اما شبیه‌سازی وقت کار را افزایش داده است. بیشتر مشتریان باید منتظر باشند زیرا هزینه میانگین انتظار گران نیست ولی سرویس‌دهنده مدیریت زیادی بی‌کار نیست. اینها تعداد بین هزینه و زمان انتظار را برقرار می‌کند.

مثال ۹-۳ مسئله رستوران ماشین سواران

هدف این مثال تعیین روزین شبیه‌سازی برای وقتی است که بیش از یک سرویس‌دهنده داریم. مشتریان با اتوبوس به رستوران وارد می‌شوند. سفارش آنها توسط پیشخدمت دریافت و غذایشان در داخل ماشین تحویل می‌گردد. ماشین‌ها مطابق جدول ۹-۱۱ وارد می‌شوند. در سرویس‌دهنده پیش خدمت و آشپز وجود ندارد. پیش خدمت سریعتر از آشپز کار می‌کند. توزیع زمان سرویس در جدول ۹-۱۲ و ۹-۱۳ نشان داده شده است.

| زمان بین ورودها (دقیقه) | احتمال | احتمال تجمعی | حوزه رقم تصادفی |
|-------------------------|--------|--------------|-----------------|
| 1 | 0.25 | 0.25 | 01-25 |
| 2 | 0.40 | 0.65 | 26-65 |
| 3 | 0.20 | 0.85 | 66-85 |
| 4 | 0.15 | 1.00 | 86-00 |

جدول ۹-۱۱ توزیع بین ورود ماشینها

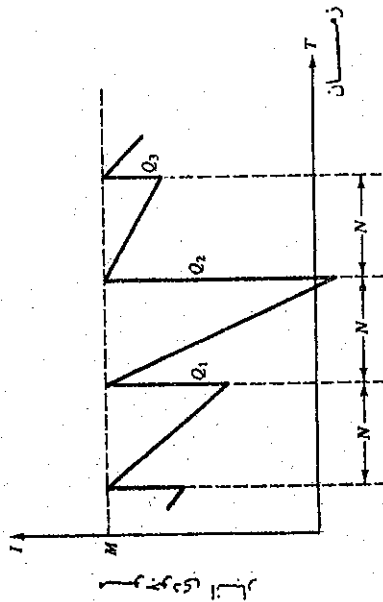
| زمان سرویس (دقیقه) | احتمال | احتمال تجمعی | حوزه رقم تصادفی |
|--------------------|--------|--------------|-----------------|
| 2 | 0.30 | 0.30 | 01-30 |
| 3 | 0.28 | 0.58 | 31-58 |
| 4 | 0.25 | 0.83 | 59-83 |
| 5 | 0.17 | 1.00 | 84-00 |

جدول ۹-۱۲ توزیع سرویس پیش خدمت

| زمان سرویس (دقیقه) | احتمال | احتمال تجمعی | حوزه رقم تصادفی |
|--------------------|--------|--------------|-----------------|
| 3 | 0.35 | 0.35 | 01-35 |
| 4 | 0.25 | 0.60 | 36-60 |
| 5 | 0.20 | 0.80 | 61-80 |
| 6 | 0.20 | 1.00 | 81-00 |

جدول ۹-۱۳ توزیع سرویس آشپز

مشهود است که پس از دریافت سفارش، ابتدا تقاضایی که باعث رخ داد کمبود کالا می‌گردد سفارش داده می‌شود. برای اجتناب از کمبود کالا، یک بافر به عنوان نقطه اطمینان یا مقداری کالا به عنوان موجودی انبار باید در نظر گرفته شود.



شکل ۹-۱۶ - احتمالات سطح - سفارش در انبار

نگهداری موجودی انبار مستلزم هزینه‌ای است که باید به آن توجه کرد و در صورت کمبود کالا تقاضای خرید کالاهای مورد نیاز صورت گیرد. هزینه‌های دیگر نگهداری کالا عبارتند از: هزینه انبارداری، حفاظت انبار، حفاظت از آتش سوزی و مانند آن است. یک راه نگهداری موجودی، بررسی موجودیها و جایگزینی سریع می‌باشد که هزینه سفارش بیشتر می‌گردد و هزینه کمبود کالا نیز افزایش می‌یابد. انبارهای بزرگتر احتمال کمبود کالا را کاهش می‌دهند که این هزینه‌ها باید کل هزینه سیستم انبار را کاهش می‌دهد.

هزینه کل انبار به عنوان ملک کارآیی محسوب می‌گردد که سیاستهای جایگزینی می‌تواند در آن مؤثر واقع شود. در شکل ۹-۱۶ تقسیم‌گیرنده می‌تواند حداکثر سطح انبار (M) و طول سیکل آن (N) را کنترل کند.

در یک سیستم انبار (MN) پیش‌آمدهایی که رخ می‌دهند عبارتند از: دریافت تقاضای کالای انبار، بررسی وضعیت انبار، تأیید دریافت کالاهای یک سفارش در پایان پریود بررسی. اگر مانند شکل ۹-۱۶ زمان تأخیر صفر شود، در پیش‌آمد آخر همزمان رخ می‌دهند. جزئیات بیشتر در فصل بعدی بیان می‌گردد و در اینجا به مثالهایی جهت نمایش شبیه‌سازی انبار می‌پردازیم.

شبیه‌سازی مطابق مثال ۹-۱ انجام شده و انتظار می‌رفت که پیچیده‌تر باشد. یک راه ساده سازی این است که اگر هر دو نفر بی‌کار باشد، پیشخدمت به علت سرعت عمل مشغول سروس او گردد. مسئله یافتن چگونگی کارایی عملیات فعلی است و از یک شبیه‌سازی با زمان یک ساعت استفاده می‌شود. بدیهی است که شبیه‌سازی طولانی‌تر دقت بیشتری می‌دهد ولی به علت چگونگی بیان مسئله زمان یک ساعت انتخاب شده است. شبیه‌سازی به روش مثال ۹-۱ صورت می‌گیرد. پیش‌آمدهای بیشتری با این شبیه‌سازی داریم زیرا یک سروس دهنده دوم نیز وجود دارد. پیش‌آمدهای ممکن عبارتند از: ورود مشتریان، شروع سروس مشتری با پیشخدمت، پایان سروس پیشخدمت، شروع سروس به وسیله آشپز، پایان سروس مشتری توسط آشپز جدول ۹-۱۴ مبین جدول شبیه‌سازی است. با تحلیل این جدول نتایج زیر بدست می‌آید.

۱- در تمام پریود ۶۲ دقیقه، پیش خدمت ۹۰٪ زمان مشغول کار است.

۲- آشپز فقط ۶۹٪ مشغول است.

۳- ۲۵٪ ورودیها منتظر می‌مانند. متوسط زمان انتظار هر مشتری ۰/۴۲ دقیقه یا ۲۵ ثانیه است که بسیار کوچک می‌باشد (قطر ۹ نفر انتظار).

۴- ۹ نفر منتظر بطور متوسط ۱/۲۲ دقیقه انتظار می‌کشند که ناچیز است.

۵- بطور خلاصه به نظر می‌آید که این سیستم بسیار متعادل است. یک سروس دهنده برای غذا دادن همه کافی نیست و سه سروس دهنده غیر اقتصادی می‌باشد. البته یک سروس دهنده اضافی زمان انتظار را به صفر می‌رساند گر چه کارگر اضافی هزینه را افزایش می‌دهد.

مثال ۹-۱۶ شبیه‌سازی سیستم‌های انبار

یکی از مهمترین طبقه‌های شبیه‌سازی شامل سیستم‌های انبار می‌باشد. یک سیستم انبار در شکل ۹-۱۶ نشان داده شده است. این سیستم انبار دارای دوره سفارش با طول N می‌باشد که همان زمان بازرسی موجودی انبار است. با هر بازرسی و سفارش کالا موجودی انبار به سطح M خواهد رسید. در پایان اولین دوره سفارش، یک سفارش با مقدار O انجام می‌گیرد. در این سیستم انبار زمان تأخیر (طول زمان بین ارسال و دریافت یک سفارش) صفر است. از آنجا که تقاضاها معمولاً با نظم‌میان قابل شناخت نیست، حد سفارش تصادفی است. تقاضا به صورت یکخواهت در شکل ۹-۱۶ نشان داده شده است (عملاً تقاضای توزیع یکخواهت ندارد). یک احتمال آنکه تقاضاها در شروع سیکل بازرسی رخ دهد و امکان دیگر تصادفی و غیر صفر بودن تقاضاها است.

در سومین سیکل، مقدار انبار به زیر صفر می‌رود و یک کمبود ایجاد می‌گردد. در شکل ۹-۱۶

| حوزه رقم تصادفی | احتمال نسبی | احتمال | نوع تقاضا |
|-----------------|-------------|--------|-----------|
| 01-35 | 0.35 | 0.35 | خوب |
| 36-80 | 0.80 | 0.45 | متوسط |
| 81-00 | 1.00 | 0.20 | ضعیف |

جدول ۹-۱۶ - حوزه عدد تصادفی نوع تقاضای روزانه

| تقاضا | احتمال نسبی | | | حوزه رقم تصادفی | | |
|-------|-------------|-------|------|-----------------|-------|-------|
| | خوب | متوسط | ضعیف | خوب | متوسط | ضعیف |
| 40 | 0.03 | 0.10 | 0.44 | 01-03 | 01-10 | 01-44 |
| 50 | 0.08 | 0.28 | 0.66 | 04-08 | 11-28 | 45-66 |
| 60 | 0.23 | 0.68 | 0.82 | 09-23 | 29-68 | 67-82 |
| 70 | 0.43 | 0.88 | 0.94 | 24-43 | 69-88 | 83-94 |
| 80 | 0.78 | 0.96 | 1.00 | 44-78 | 89-96 | 95-00 |
| 90 | 0.93 | 1.00 | 1.00 | 79-93 | 97-00 | |
| 100 | 1.00 | 1.00 | 1.00 | 94-00 | | |

جدول ۹-۱۷ - حوزه رقم تصادفی تقاضای روزانه

جدول شبیه‌سازی خرید ۷۰ روزانه در جدول ۹-۱۸ نشان داده شده که در روز اول تقاضای ۶۰ روزانه و در آمد حاصل از فروش ۶۰ روزانه ۱۲۰۰ ریال است. ۱۰ روزانه در انتهای روز فروش نمی‌رود و در آمد کاغذ باطله آن ۲۰ ریال از ۲۰۰ ریال است. سود اولین روز می‌شود: $ریال\ ۳۱۰ = ۲۰۰ - ۱۰۰ - ۱۴۰ = ۱۴۰ = سود$

در روز پنجم تقاضا بیشتر از موجودی است. درآمد فروش ۱۴۰ ریال می‌باشد و این روز فقط ۷۰ روزانه با این سیاست موجود است. تعداد ۲۰ روزانه می‌تواند فروش رفته باشد و بنابراین سود از دست رفته (۷۸۲۰) ۱۴۰ ریال می‌شود. سود روزانه به صورت زیر می‌باشد:

$ریال\ ۳۵۰ = ۱۴۰ - ۱۰ - ۱۴۰ = ۱۴۰ = سود$
سود ۲۰ روز مجموع سود روزانه، ۷۲۶۰ ریال به صورت زیر تعیین می‌شود:

$ریال\ ۳۵۰ = ۱۴۰ - ۱۰ - ۱۴۰ = ۱۴۰ = سود$
 $ریال\ ۷۲۶۰ = ۲۲۰ + ۵۶۰ - ۱۸۲۰ = ۲۵۸۰ = سود\ کل$

تعیین تعداد بهینه روزانه‌های قابل تهیه را به صدها خواننده می‌گذاریم.

مثال ۹-۵ - مسئله روزانه فروش

در این مثال یک پرورد زمانی با طول متغیر مد نظر است و فقط یک فروشنده روزانه دارد. موجودی انبار روزانه‌ها در انتهای پرورد زمانی تخلیه می‌شود و ممکن است از ارائه تخفیف فروش رود. از روشهای نوین انبارداری ارتباط بین خرید و فروش را برقرار می‌کند و در این مثال نیز مطرح می‌شود. فرض، روزانه فروش هر روزانه را ۱۳۰ ریال خریداری و ۲۰۰ ریال آن را می‌فروشد. روزانه فروش روزانه‌ها را به صورت بسته‌های ده‌تایی خریداری می‌نماید. روزانه‌هایی که در انتهای هر روز فروش نمی‌روند با قیمت ۲۰ ریال به عنوان کاغذ باطله باطله می‌فروش می‌رسد. خرید روزانه‌ها می‌تواند ۵۰ تایی یا ۶۰ تایی انجام پذیرد. به تجربه نشان داده شده است که تقاضاهای روزانه تحت سه عنوان "خوب"، "متوسط" و "ضعیف" با احتمالات نظیر ۳۵/۱۰۰، ۴۵/۲۰ و یا ۲۰/۵۰ دسته‌بندی می‌گردند. توزیع تقاضای روزانه برای این روزها در جدول ۹-۱۵ ارائه شده است. مسئله یافتن بهینه‌ترین تعداد روزانه‌هایی است که فروشنده باید خریداری نماید. این امر با شبیه‌سازی تقاضاها برای ۲۰ روز و ثبت نتایج فروش هر روز صورت می‌گیرد.

| تقاضا | توزیع احتمالی تقاضا | | |
|-------|---------------------|-------|------|
| | خوب | متوسط | ضعیف |
| 40 | 0.03 | 0.10 | 0.44 |
| 50 | 0.05 | 0.18 | 0.22 |
| 60 | 0.15 | 0.40 | 0.16 |
| 70 | 0.20 | 0.20 | 0.12 |
| 80 | 0.35 | 0.08 | 0.06 |
| 90 | 0.15 | 0.04 | 0.00 |
| 100 | 0.07 | 0.00 | 0.00 |

جدول ۹-۱۵ - توزیع تقاضای روزانه

سود روزانه به طریق رابطه زیر بیان می‌شود:

درآمد مصرف مجدد از + سود از دست رفته - هزینه روزانه‌ها - درآمد از فروش = سود فروش کاغذهای نگه‌داره تقاضای اضافی

عایدی فروش ۲۰۰ ریال و هزینه خرید ۱۳۰ ریال برای هر روزانه می‌باشد. در صورت تقاضای اضافی سود از دست رفته تقاضای اضافی ۷۰ ریال برای هر تقاضای روزانه است. عایدی حاصل از کاغذهای باطله ۲۰ ریال به ازای هر روزانه می‌باشد. جدول ۹-۱۶ و ۹-۱۷ - حوزه عدد تصادفی برای نوع و تقاضای روزانه را دارد. برای حل مسئله از طریق شبیه‌سازی خرید تعداد معینی روزانه برای هر روز مد نظر قرار می‌گیرد و سپس شبیه‌سازی را تکرار ۲۰ روز برای تعیین سود کل انجام می‌گیرد.

مثال ۹-۶. شبیه سازی یک سیستم اتیار (M, N)

| تفاضل | احتمال | احتمال تجمعی | استدلال تصادفی | حوزه رقم |
|-------|--------|--------------|----------------|----------|
| 0 | 0.10 | 0.10 | | 01-10 |
| 1 | 0.25 | 0.35 | | 11-35 |
| 2 | 0.35 | 0.70 | | 36-70 |
| 3 | 0.21 | 0.91 | | 71-91 |
| 4 | 0.09 | 1.00 | | 92-00 |

جلدول ۹-۱۹ حوزه ارقام تصادفی برای تقاضا

| ارقام تصادفی | احتمال تجمعی | احتمال | تاخیر (روز) |
|--------------|--------------|--------|-------------|
| 1-6 | 0.6 | 0.6 | 1 |
| 7-9 | 0.9 | 0.3 | 2 |
| 0 | 1.0 | 0.1 | 3 |

جدول ۹-۲۰ حوزه ارقام تصادفی برای زمان تأخیر (مهلت دریافت کالا)

در جداول ۹-۱۹ و ۹-۲۰ حوزه ارقام تصادفی برای تقاضا روزانه و زمان تأخیر نشان داده شده است. نتایج جدول شیشه‌سازی در جدول ۹-۲۱ آمده است. شیشه‌سازی با سطح انبار ۳ واحد و یک سفارش ۸ واحدی که در ۲ روز دریافت می‌شود، آغاز می‌گردد.

چندین روز مشخص جدول شیشه‌سازی را برای چگونگی انجام پروسه‌ها دنبال می‌کنیم. سفارش ۸ واحدی در صبح روز سوم در اولین سیکل موجود است و سطح انبار از ۱ واحد به ۹ واحد می‌رسد. تقاضا در انتهای باقیمانده اولین سیکل موجودی انبار را به سطح ۲ واحد در چهارمین روز رساند.

۷۰ روزه ۸۰ روزه ۹۰ روزه ۱۰۰ روزه ۱۱۰ روزه ۱۲۰ روزه ۱۳۰ روزه ۱۴۰ روزه ۱۵۰ روزه ۱۶۰ روزه ۱۷۰ روزه ۱۸۰ روزه ۱۹۰ روزه ۲۰۰ روزه ۲۱۰ روزه ۲۲۰ روزه ۲۳۰ روزه ۲۴۰ روزه ۲۵۰ روزه ۲۶۰ روزه ۲۷۰ روزه ۲۸۰ روزه ۲۹۰ روزه ۳۰۰ روزه ۳۱۰ روزه ۳۲۰ روزه ۳۳۰ روزه ۳۴۰ روزه ۳۵۰ روزه ۳۶۰ روزه ۳۷۰ روزه ۳۸۰ روزه ۳۹۰ روزه ۴۰۰ روزه ۴۱۰ روزه ۴۲۰ روزه ۴۳۰ روزه ۴۴۰ روزه ۴۵۰ روزه ۴۶۰ روزه ۴۷۰ روزه ۴۸۰ روزه ۴۹۰ روزه ۵۰۰ روزه ۵۱۰ روزه ۵۲۰ روزه ۵۳۰ روزه ۵۴۰ روزه ۵۵۰ روزه ۵۶۰ روزه ۵۷۰ روزه ۵۸۰ روزه ۵۹۰ روزه ۶۰۰ روزه ۶۱۰ روزه ۶۲۰ روزه ۶۳۰ روزه ۶۴۰ روزه ۶۵۰ روزه ۶۶۰ روزه ۶۷۰ روزه ۶۸۰ روزه ۶۹۰ روزه ۷۰۰ روزه ۷۱۰ روزه ۷۲۰ روزه ۷۳۰ روزه ۷۴۰ روزه ۷۵۰ روزه ۷۶۰ روزه ۷۷۰ روزه ۷۸۰ روزه ۷۹۰ روزه ۸۰۰ روزه ۸۱۰ روزه ۸۲۰ روزه ۸۳۰ روزه ۸۴۰ روزه ۸۵۰ روزه ۸۶۰ روزه ۸۷۰ روزه ۸۸۰ روزه ۸۹۰ روزه ۹۰۰ روزه ۹۱۰ روزه ۹۲۰ روزه ۹۳۰ روزه ۹۴۰ روزه ۹۵۰ روزه ۹۶۰ روزه ۹۷۰ روزه ۹۸۰ روزه ۹۹۰ روزه ۱۰۰۰ روزه

[illegible]

| روزها
تا رسیدن سفارش برای زمان تاخیر | رقم تصادفی | حد سفارش | حد کمبود کالا | انتهای انبار | تقاضا | رقم تصادفی تقاضا | شروع انبار | روز | دوره |
|---|------------|----------|---------------|--------------|-------|------------------|------------|-----|------|
| 1 | — | — | 0 | 2 | 1 | 24 | 3 | 1 | 1 |
| 0 | — | — | 0 | 1 | 1 | 35 | 2 | 2 | |
| — | — | — | 0 | 7 | 2 | 65 | 9 | 3 | |
| — | — | — | 0 | 4 | 3 | 81 | 7 | 4 | |
| 1 | 5 | 9 | 0 | 2 | 2 | 54 | 4 | 5 | |
| 0 | — | — | 0 | 2 | 0 | 03 | 2 | 1 | 2 |
| — | — | — | 0 | 8 | 3 | 87 | 11 | 2 | |
| — | — | — | 0 | 7 | 1 | 27 | 8 | 3 | |
| — | — | — | 0 | 4 | 3 | 73 | 7 | 4 | |
| 3 | 0 | 9 | 0 | 2 | 2 | 70 | 4 | 5 | |
| 2 | — | — | 0 | 0 | 2 | 47 | 2 | 1 | 3 |
| 1 | — | — | 2 | 0 | 2 | 45 | 0 | 2 | |
| 0 | — | — | 4 | 0 | 2 | 48 | 0 | 3 | |
| — | — | — | 0 | 4 | 1 | 17 | 9 | 4 | |
| 1 | 3 | 7 | 0 | 4 | 0 | 09 | 4 | 5 | |
| 0 | — | — | 0 | 2 | 2 | 42 | 4 | 1 | 4 |
| — | — | — | 0 | 6 | 3 | 87 | 9 | 2 | |
| — | — | — | 0 | 5 | 1 | 26 | 6 | 3 | |
| — | — | — | 0 | 3 | 2 | 36 | 5 | 4 | |
| 1 | 4 | 10 | 0 | 1 | 2 | 40 | 3 | 5 | |
| 0 | — | — | 0 | 1 | 0 | 07 | 1 | 1 | 5 |
| — | — | — | 0 | 9 | 2 | 63 | 11 | 2 | |
| — | — | — | 0 | 8 | 1 | 19 | 9 | 3 | |
| — | — | — | 0 | 5 | 3 | 88 | 8 | 4 | |
| 2 | 8 | 10 | 0 | 1 | 4 | 94 | 5 | 5 | |
| | | | | 87 | | | | | |

جدول ۹-۳۱ جدول شبیه‌سازی سیستم انبار (M و M)

بنابراین یک سفارش ۹ واحدی انجام می‌شود. زمان تاخیر این سفارش یک روز است. سفارش ۹ واحدی صبح روز دوم سیکل دوباره اضافه می‌شود.

توجه اینکه شروع انبار در دومین روز سیکل سوم صفر بود. یک سفارش دو واحدی در آن روز وضعیت کمبود کالا را باعث می‌شود. این واحد در آن روز باعث عقب افتادن سفارش می‌شود.

همچنین در روز بعد.

در صبح روز ۴ از سیکل ۳ شروع انبار با ۹ واحد بود، ۴ واحد سفارش عقب افتاد و تقاضای ۱ واحدی آن روز، موجودی انتهایی انبار را به ۴ واحد کاهش داد. براساس پنج سیکل شبیه‌سازی میانگین انتهایی انبار تقریباً ۳/۵ واحد (۸۷/۲۵) است. در ۲ روز از ۲۵ روز حالت کمبود کالا وجود دارد.

فصل دهم

سیستمهای صف

صف یک خط انتظار برای سروس گرفتن مشتریان است. ماشینها، کشتیها، تلویزیونها و مانند آن به یک مکان مطمئن وارد می شوند تا به هر طریقی سروس داده شوند. مردم در بانکها، ادارات پست، کشتی ها و راه آهن تشکیل صف می دهند تا سروس دریافت کنند. وقتی که نرخ ورود و نرخ سروس ارتباط واقعی با یکدیگر نداشته باشند، صف توسعه می یابد یا سروس دهنده بیکار می ماند. صف که طولانی می شود میانگین زمان تلف شده زیاد است و افزایش سروس دهنده تنها راه حل این مشکل نیست. یک مسئله صفی اساساً تعادل هزینه زمان انتظار و هزینه زمان اتلاف یا بیکاری سروس دهنده می باشد. این تعادل نیاز به آنالیز سیستم صف دارد که آمارهای مختلفی از قبیل زمان بیکاری، میانگین زمان انتظار، طول صف و غیره برای حالت های مختلف مد نظر است. زمان سروس مشتریان نیز در صف مهم است (مشتری در صف اساساً فرد نیست یک مشتری برای مثال می تواند ماشین، قطعه و غیره باشد).

از آنجایی که خط انتظار و سروس دیدن یک قانون مدارم در زندگی روزانه است، آنالیزهای بسیاری براساس مطالعه صف انجام می دهیم. منابع جدیدهای در تئوری صف موجود است و با ترکیب پارامترهای صف فرمولهای زیادی را برای مطالعه آن ایجاد می کند.

پارامترهای مهم در یک سیستم صف عبارتند از:

- (۱) الگوی ورود مشتریان، فروانی توزیع زمان ورودها (مولد زمان و ورود مشتریان)
 - (۲) الگوی سروس، فروانی توزیع زمان سروسها (مولد زمان سروس)
 - (۳) تعداد سروس دهنده ها (کانتالها یا شمارنده های سروس)
 - (۴) سیاست صف (انتخاب برای سروس LIFO، FIFO و غیره)
 - (۵) مدل سروس دهنده ها (موازی، سری، ترکیبی)
- قدیمی ترین کار روی تئوری ریاضی صف به وسیله ارلنگ حدود سال ۱۹۱۸ درباره ترافیک

بسیار به تعداد مشتریان دارد. مثلاً وقتی یک هواپیمای به وسیله یک تیم مکانیک (۲۴ ساعته) سرویس می‌پیشد، نرخ ورود صفر است زیرا دیگر جایی برای هواپیمای دیگر نیست. یعنی در این مدل هواپیمایی نمی‌تواند به صف وارد شود اما وقتی هواپیمای سریع سرویس داده شود، نرخ ورودی ماکزیمم می‌گردد. پس زمان سرویس نقش اساسی در تعیین نرخ ورودی سیستم دارد. گنجایش یا محدودیت تعداد مشتریان جهت انتظار در صف عامل محدودکننده دیگر صف است.

مثلاً جهت شستشوی ماشین محلی برای ده ماشین جا دارد. لذا وقتی صف داریم طول صف نیز عامل محدودکننده بدی است. حتی این که نرخ ورود به سیستم (صف) نرخ عادی یا نرخ مؤثر است عامل دیگری می‌باشد. چون ممکن است به سیستم مشتری وارد نشود ولی به علت آنکه صف گنجایش ندارد از ورود آنها جلوگیری می‌کنیم. مشتریان در سیستم می‌تواند شروع کنند نرخ ورود مؤثر مطرح می‌گردد. سیاست صف نیز عامل تعیین کننده در زمان انتظار می‌باشد، اینکه سیاست صف PRIORITY, FIFO, است.

پس در یک صف عوامل زیر به عنوان پارامترهای اختصاصی آن است:

- نام صف (اسم صف): (QName name)
- حداکثر طول صف (ظرفیت صف) (QCapacity customers)
- مقدار اولیه صف (در بدو کار) (QInitial queue)
- سیاست صف (خروج برای سروس) (QPolicy of queue)
- مکان نگهداری صف (گد نام فایل صف) (QFLD file queue)
- رفتار صف (ورود به صف پُر) (نیی اثر یا BLOCK یا BALK)
- کارایی صف (حداکثر کارایی صف) (Performances)

۱۰.۱.۲ - رفتار و سیاست صف

رفتار صف، رفتار مشتری در ورود به سیستم و قرائگیری در صف است (اگر صف پُر باشد

مشتری در صف وارد شود یا نشود؟). عکس‌العملهای مختلفی دارد:

- چنانچه صف طولانی شود، مشتری صف را ترک می‌کند.
- چنانچه وارد شود و صف گنجایش نداشته باشد سیستم را ترک نمی‌کند.

مکالمات تلفن انجام گیرد. اگر یک نرخ تلفنها، زمان مکالمه و چگونگی ارتباط انتظار با تعداد خطوط را تعیین زد و سپس کارهای تحلیلی زیادی وی تئوری صف انجام شد. شیبه‌سازی اغلب در تحلیل مدل‌های صف به کار می‌رود لذا این مسئله به عنوان یک بخش عمده‌ای نمود یافته است و اغلب باهای شیبه‌سازی با قابلیت پشتیبانی شیبه‌سازی صفها طراحی می‌گردند.



شکل ۱۰-۱ یک صف و اجرای آن

در مثال ۱۰-۱ یک صف و بعضی اجرای آن نشان داده شده است. در آن، انتظار برای سروس و سروس دیدن به وضوح مشخص است. سیستم صف بسته به افزایش تعداد خط‌های انتظار و تعداد سروس دهنده‌ها پیچیده‌تر می‌شود و سیاست یا شیوه انتخاب از صف برای سروس، بر این امر می‌افزاید. توجه به مواردی از قبیل کارایی صف، درصد اشغال سروس دهنده، طول صف، تأخیر مشتریان برای سروس و مانند آن مطرح می‌گردد. تئوری صف یا شیبه‌سازی آن، تعیین این پارامترها و امثال آن به عنوان یک تابع از پارامترهای ورودی است. پارامترهای ورودی شامل نرخ ورود و مشتریان، نیاز به سروس مشتریان، نرخ کار سروس دهنده، تعداد و ترتیب سروس دهنده و مانند آن است.

۱۰.۱.۱ - صف

۱۰.۱.۱ - انواع سیستمهای صف

سیستمهای صف را می‌توان از جهت ظرفیت مشتریان به دو دسته کلی تقسیم‌بندی کرد:

سیستم نامعین-نیمین: با پتانسیل بالای مشتریان است. مانند فرودگاه، قطار، رستوران و مانند آن. سیستم معین: سیستم با تعداد کم و مشخص مشتریان است. مانند سیستمهای صف برای تعمیر

هواپیمای.

فروق این سیستم‌ها در نرخ ورودی آنهاست. در یک سیستم نامعین، نرخ ورود با تعداد مشتریان بستگی

| سیستم | مشتری‌ها | سرویس دهنده‌ها |
|----------------|------------|---------------------|
| تعمیرات | ماشین‌ها | تعمیرکار |
| گازاژ | کامیون‌ها | مکانیک |
| بیمارستان | بیماران | پرستار |
| فروشگاه | هواپیماها | راه حرکت |
| فروشگاه کتاب | کتاب‌ها | فروشنده |
| کامپیوتر | کارها | پردازنده، نور، دیسک |
| استادیوم ورزشی | تماشاچی‌ها | مستول فروش بلیط |

جدول ۱۰-۱: مثالهایی از صف

۱۰.۲ - پروسه ورود

فرایند ورود برای مدل‌های جمعیت نامعین به صورت زمان بین ورود مشتریان ارائه می‌گردد. فرایند ورود مشتریان به صف (سیستم) ممکن است زمان‌بندی شود یا با زمان تصادفی وارد شوند. وقتی زمان تصادفی باشد، زمانهای ورود معمولاً به وسیله توزیع احتمال دسته‌بندی می‌گردند. به علاوه مشتریان ممکن است، هرکدام در یک زمان یا بصورت یکجا (دسته‌ای) وارد گردند. دسته‌ها ممکن است اندازه ثابت یا متغیر داشته باشند.

مهمترین مدل فرایند ورود تصادفی براساس A_n است. اگر A_n زمان بین ورود مشتری $(n-1)$ و مشتری n A_1 زمان واقعی رسیدن اولین مشتری)، باشد پس برای یک فرایند بواسون A_n به صورت نمایی با میانگین $\frac{1}{\lambda}$ واحد زمانی توزیع شده است. نرخ ورود λ مشتری در واحد زمان است و تعداد ورودی در یک زمان با طول t $N(t)$ گریند که توزیع بواسن با میانگین λt مشتری می‌باشد.

فرایند بواسن به عنوان مدل ورود مردم به رستوران، عملیات در بانک و سرویسهای دیگر، مکالمات تلفن، نیاز به یک سرویس یا محصول، ورود عناصر خراب یا ناخوش برای سرویس است. دومین حالت ورود، ورود زمان‌بندی شده مانند رسیدن مواد به فرایند تولید محصول است. در این حالت، زمانهای $\{A_n, n=1, 2, \dots\}$ می‌تواند ثابت باشد یا به صورت ثابت افزایش یا کاهش با یک مقدار تصادفی باشد.

سومین حالت وقتی که حداقل یک مشتری (فرض شده) همیشه در صف باشد، همچنین

● چنانچه وارد شود و صف گنجایش نداشته باشد به محل دیگری هدایت^(۱) می‌شود.

● از ورود آن جلوگیری می‌شود^(۲) تا زمانی که صف خروجی داشته باشد؛ کل فعالیتهای مربوط به تولید مشتری متوقف یا بلوکه شود. لذا می‌توان گفت که صف باید وضعیت مشتری را نیز در این حالتها معین کند، آیا فعالیتهای منجر به صف را بلوکه می‌کند یا آنکه نسبت به ورود و خروج آن بی‌تفاوت است و آیا به محل دیگر آتفا ارسال می‌نماید.

سیاست صف:

سیاست صف یعنی خروج از صف جهت دریافت سرویس چگونه تعریف شده است. صف کدامین مشتری از صف را به عنوان خروجی بفرستد؟ که سیاستهای موجود انتخاب برای سرویس به صورت زیر است:

● اولین ورود اولین خروج: LIFO: First In First Out، اولین ورود اولین خروجی است (انتخاب اولین ورودی).

● اولین ورود آخرین خروج: LIFO: Last In First Out، آخرین ورود-اولین خروجی است (انتخاب آخرین ورودی).

● انتخاب تصادفی SIRO: Service In Randomly، انتخاب نقرات به صورت تصادفی است

(تصادفی).

● انتخاب کوتاه‌ترین زمان سرویس SPT: Shortest Processing Time، کوتاه‌ترین زمان برای سرویس اولین است (کوتاه‌ترین زمان).

● انتخاب براساس اولویت PRI: Priority، سرویس براساس اولویت آنست (اولویت).

● انتخاب بیشترین زمان پردازش LPT: Longest Processing Time، بزرگترین زمان برای سرویس اولین است (بزرگترین زمان).

معمولاً زمانهای شبه‌سازی پس از تعیین صفها در محل دیگری صفها و سیاستهایشان را تعریف می‌کنند.

مثالهایی از سیستم صف در جدول زیر آمده است.

ورودی در واحد زمان) و نرخ ورود مؤثر (تعدادی که به صف وارد می‌شوند) تفاوت است.

مکانیزم‌های سرویس دهی

زمان سرویس ورودیها به وسیله S_1, S_2, S_3, \dots ارائه می‌شود که می‌تواند مقدار ثابت یا تصادفی داشته باشد. مقدار سلسله مستقل $\{S_1, S_2, \dots\}$ بطور تصادفی توزیع شده است. مدلهای ویل نمائی و توزیع نرمال به عنوان زمان سرویس می‌تواند بکار رود و همبطور در بعضی سیستم‌ها، زمان سرویس بستگی به طول خط انتظار دارد. برای مثال سرویس دهند هها وقتی خط انتظار طولانی می‌شود سریعتر کار می‌کنند.

یک سیستم صف شامل یک تعدادی مرکز سرویس و صفهای متصل است. هر مرکز سرویس دارای چندین سرویس دهنده (C) است که بطور موازی کار می‌کنند. در مکانیزم سرویسهای موازی ممکن است یک سرویس دهنده ($C=1$) چند سرویس دهنده ($C>1$) یا نامحدود ($C=\infty$) باشند. پارامترهای مهم:

۱- سرویس دهند ههای موازی:

۱- ($C=1$) یک سرویس دهنده

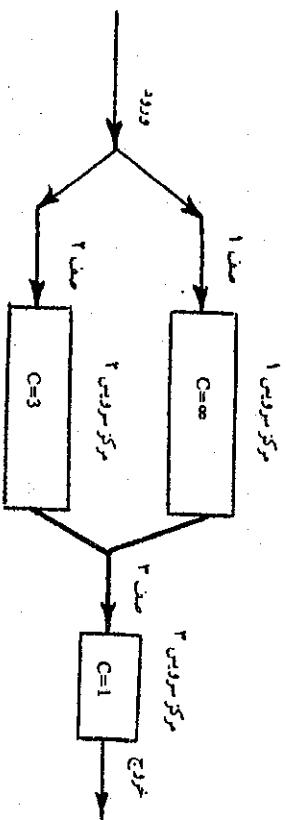
۲- ($1 < C < \infty$) چندین سرویس دهنده

۳- ($C=\infty$) سرویس دهنده نامحدود

۲- وضعیت سرویس دهند ه:

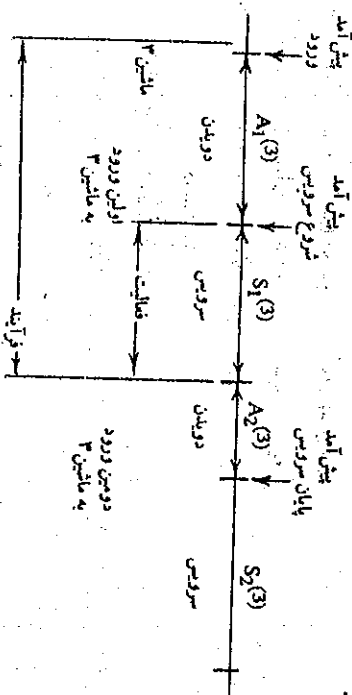
۱- یک کار (idle) - ۲- مشغول (busy)

مثال ۱-۰۱ سرویس دهند ههای موازی با سه مرکز سرویس



سرویس دهند ه هجوقت یک کار نیست. برای مدلهای جمعیت معین، فرایند ورود کاملاً متفاوت دستبندی می‌شود. وقتی که مشتری از سیستم صف خارج می‌شود، یک عضو جمعیت را صدا می‌زنند $A_1^{(1)}, A_2^{(1)}, \dots, A_n^{(1)}$ زمانهای حرکت مشتری است و $S_1^{(1)}, S_2^{(1)}, \dots, S_n^{(1)}$ و دفعات سرویس سربرطه آن در سیستم است. $S_k^{(1)}$ کل زمان صرف شده در سیستم به وسیله مشتری ادر انثای تاامین ملاقات است. مثلاً اگر فرض شود که تمامی ماشین‌های شکل ۲-۱۰ در زمان صف حرکت کنند اولین ورود به سیستم در زمان A_1 رخ می‌دهد که: $A_1 = \min(A_1^{(1)}, A_1^{(2)}, A_1^{(3)}, A_1^{(4)}, A_1^{(5)}, A_1^{(6)})$

اگر $A_1 = A_1^{(2)}$ یعنی ماشین ۲ اولین ورودی بعد از زمان صفر است. نرخ ورود ثابت نیست و تابع مشتریان است و یکی از کاربردهای مدل جمعیت معین، مسئله تغییر ماشین است. ماشینها همان مشتریان هستند و یک زمان اجراء، زمان تا خرابی ماشین است، وقتی که ماشین خراب می‌شود، به صف وارد می‌گردد و دستپر می‌ماند تا "سرویس" (تعمیر) داده شود. زمانی خرابی ماشین‌ها به صورت توزیع نمائی، ویل و گاما است. مدلهای با یک زمان اجرای نمائی معمولاً قابل ردیابی هستند، مانند خرابی دستگاهها که با استقلال آماری به عوامل دیگر بستگی دارد. مانند عمر ماشین (که مهمترین عامل است).



شکل ۲-۱۰ فرایند ورود برای یک مدل جمعیتی - معین

در سیستم صف نیاز به تابع مولد مشترها با توزیع‌های گوناگون داریم و جریان صف با آن حیات می‌یابد. در یک سیستم صف که محدودیت تعداد مشتری در خط انتظار دارد، مشتری ورودی که صف پر را می‌بیند نمی‌تواند وارد شود سریع به جمعیتی که از آنجا صدا زده شده برمی‌گردد. بعضی سیستمها مانند ثبت نام دانش آموزان به عنوان سیستمهای با گنجایش نامحدود هستند و محدودیت تعداد شاگردان دستپر ثبت نام ندارد. همانطور که می‌دانید در چنین سیستمهایی، بین نرخ ورود (تعداد

موازی، گنجایش نامحدود و پتانسیل نامحدود جمعیت.

بعضی علائم و نتایج مورد نیاز در سیستم صف به صورت زیر است:

P_n : احتمال وجود n مشتری در حالت ثابت

$P_n(t)$: احتمال وجود n مشتری در سیستم در زمان t

۱: نرخ ورود

λ_e : نرخ مؤثر ورود

μ : نرخ سرویس یک سرویس دهنده

μ_e : نرخ مؤثر یک سرویس دهنده

A_n : زمان بین ورود مشتری $n-1$ و مشتری n

S_n : زمان سرویس n امین مشتری

W_n : کل زمان مشتری n ام در سیستم

Q : کل زمان مشتری n ام در خط انتظار

$L(t)$: تعداد مشتریان سیستم در نقطه t

$LQ(t)$: تعداد مشتریان در صف در نقطه t

W : میانگین زمان صرف شده در سیستم به ازای هر مشتری

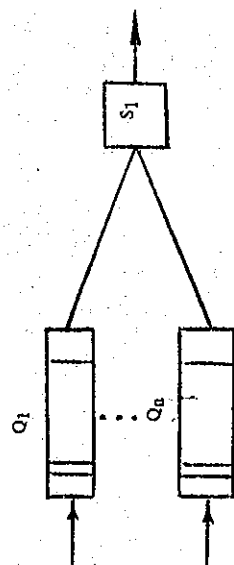
WQ : میانگین زمان صرف شده در صف به ازای هر مشتری

P : اشغال سرویس دهنده

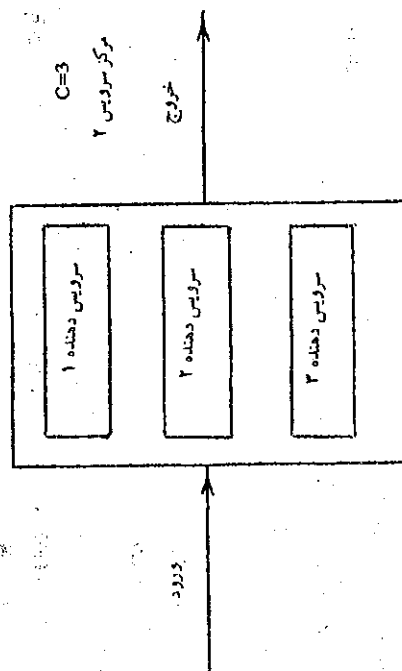
و پارامترهای بسیاری می توان از یک صف بدست آورد و جهت تحلیل وضعیت آن استفاده کرد.

۱۰.۳ - انواع مدل های صف

صف دارای مدل های مختلف است که براساس ساختاری تعداد صف های موجود در سیستم و سرویس دهنده های آن تقسیم بندی می شود و مدل های آن بشرح زیر است:



مثال ۱۰-۲ سرویس دهنده های موازی با ۳ سرویس دهنده در مرکز سرویس شماره ۲ مثال قبل



۱۰.۳ - بیان صف

کنдал در سال ۱۹۵۳ یک سیستم عمومی برای سیستم های با سرویس دهنده موازی ارائه کرد و گسترش یافت یک نگارش آن به این شکل است.

$A/B/C/N/K$

A: توزیع زمان ورودی

B: توزیع زمان سرویس

A/B شامل: توزیع M نمائی، D ثابت یا قطعی، E_k ارلنگ با k ، G عمومی یا حلقوی

C: تعداد سرویس دهنده های موازی

N: گنجایش سیستم صف جمعیت ورودی

K: پتانسیل فضای جمعیت ورودی

مثال ۱۰.۳ - $M/M/\infty/\infty$: ورودی با توزیع نمائی، زمان سرویس با توزیع نمائی، یک سرویس دهنده

مثال آن لانگرگاه است و لانگرگاه مکانی برای بارگیری یک کامیون دارد و جایی برای انتظار کامیون‌های دیگر نیست. کامیونها با توزیع پواسن و با میانگین ۴ در هر ساعت وارد می‌شوند. در حالی که زمان بارگیری/تخلیه مدل متغیر تصادفی نمایی با میانگین $\mu=1=120$ دقیقه است. همچنین نرخ سروس $M/M/1/\infty$ $\mu=0.5$ در هر ساعت و از پتانسیل جمعیت زیاد کامیونها برخوردار است که به صورت صف $M/M/1/\infty$ بیان گردید.

فرض در زمان لانگرگاه خالی است. زمان بین ورود به دقیقه:

$$A_1=10, A_2=25, A_3=5, A_4=15, A_5=20$$

و زمان سروسها به دقیقه:

$$S_1=35, S_2=20, S_3=60, S_4=15, S_5=134$$

زمان شبیه‌سازی از 0 تا 75 است. لذا پرورد شبیه‌سازی 75 دقیقه است. فرض این است که در A_1 دقیقه ۱۰ وارد شده و سروس می‌گیرد دومی و سومی (A_3, A_2) یعنی سروس S_2 و S_3 کنار گذاشته می‌شود و سروس به آنها داده نمی‌شود و چون (A_4) آمد و سروس دهنده مشغول نبود به آن با زمان S_4 سروس داده شد.

پس در زمان دقیقه $t=75$ اشتغال سروس دهنده به صورت \bar{P} است. یعنی 67% دقیقه اشتغال سروس دهنده است:

$$\bar{p} = \frac{35 + 15}{10 + 25 + 5 + 15 + 20} = \frac{50}{75} = 0.67$$

در واقع به صورت زیر است:

$$\bar{p} = \rho \left(\frac{T_0}{T} \right) + 1 \left(\frac{T_1}{T} \right) = \rho \left(\frac{25}{75} \right) + 1 \left(\frac{50}{75} \right) = 0.67$$

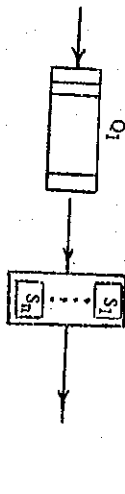
$\frac{T_1}{T}$ کسر زمانی است که به مشتری اسروس داده شده است.

\bar{P} زمان اندازه‌گیری شده با میانگین تجمعی زمان گویند. بسیاری تحلیل‌ها سیستم را در زمانهای طولانی (ثبات) یا در بخش گذرا سیستم را مد نظر قرار می‌دهند.

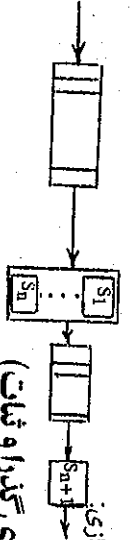
برای صف $M/M/1/\infty$ زمان اشتغال سروس دهنده به صورت ریاضی می‌شود:

$$\rho = \frac{\lambda}{\lambda + \mu} = \frac{2}{2 + 0.5} = 0.80$$

۳- یک صف و چند سروس دهنده متوالی:



۴- یک صف و چند سروس دهنده موازی:



۵- چند صف و چند سروس دهنده موازی:

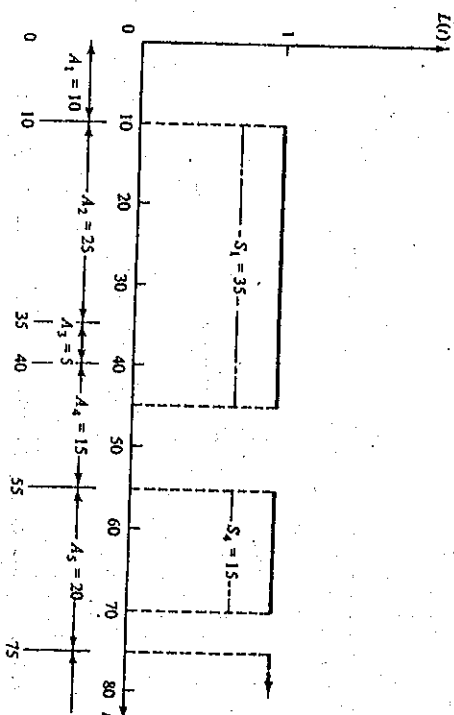
۱-۴- آنالیز صف (در حالت‌های گذرا و ثبات)

با مثالی به شرح یک صف $M/M/1/\infty$ که در لحظه زمانی صفر خالی و سروس دهنده آن بیکار

است می‌پردازیم.

مثال ۴-۱ یک صف یک سروس دهنده با نرخ ورودی پواسن و پارتر ورودی ۴ در هر واحد زمان، زمان سروس نمایی با میانگین ۲۰ در واحد زمان، گنجایش سیستم ۱ یعنی خط انتظار نیست و انتخاب از یک جمعیت نامعین است در صورتی که سروس دهنده مشغول باشد مشتری خارج می‌شود. گرچه این سیستم به صورت ریاضی حل می‌شود ولی برای رفتار گذرای صف شبیه‌سازی می‌شود (سیستم صف با یک سروس دهنده).

یک مثال از $M/M/1/\infty$ صف شکل ۴-۱۰ است. جایی که $L(t)$ تعداد مشتریان سیستم در زمان t است و در طول زمان ترسیم شده است. برای این سیستم همیشه $L(t)$ مقدار ۱ یا ۰ دارد.



شکل ۴-۱۰ سابقه بارگیری لانگرگاه

بعلاوه P مستقل از وضعیت مقداردهی اولیه است. توجه اینکه در صد اشتغال P عددی ثابت بین 0 و 1 می‌باشد. در حالی که \hat{P} متغیر تصادفی (یا آماری) که بستگی به ورود تصادفی و زمان سرویس دارد، بعلاوه \hat{P} وابسته به T طول زمان مشاهده و L وضعیت اولیه (وضعیت سیستم در زمان 0) است. هرچه T بزرگتر شود مقدار آماری \hat{P} به P نزدیکتر می‌شود:

$$\hat{P} \rightarrow P \quad T \rightarrow \infty$$

برای این سیستم \hat{P} کم است زیرا وضعیت مقدار اولیه سیستم هیچ و وضعیت بیکاری است. $B(\hat{P}) < P$ یعنی که بخاطر وضعیت بیکاری و غیره میانگین اشتغال کمتر از میانگین P است. با استفاده از تئوری ریاضی صف، احتمال حضور تعداد نفرات $(L(t))$ در زمان t می‌شود:

$$P(L(t) = 0) = P_0(t) = \frac{\mu}{\lambda + \mu} + a_0 e^{-(\lambda + \mu)t}$$

$$P(L(t) = 1) = P_1(t) = \frac{\lambda}{\lambda + \mu} + a_1 e^{-(\lambda + \mu)t}$$

a_0 و a_1 مستقل از زمان هستند و به مقدار اولیه بستگی دارند.

$$a_0 = P_0(0) - \frac{\mu}{\lambda + \mu}$$

$$a_1 = P_1(0) - \frac{\lambda}{\lambda + \mu}$$

$$t \rightarrow \infty, e^{-(\lambda + \mu)t} \rightarrow 0 \quad \lambda > 0 \quad \mu > 0,$$

$$P_0(t) \rightarrow P_0 = \frac{\mu}{\lambda + \mu}$$

$$P_1(t) \rightarrow P_1 = \frac{\lambda}{\lambda + \mu}$$

همیشه $p_1 + p_0 = 1$ است. احتمالات $\{P_i(t)\}$ را احتمالات گذرا می‌نامند چایی که احتمال، وضعیت سیستم برای زمان t را می‌دهد. سیستم در حالت اولیه L در زمان $t=0$ است و L به وسیله $P_1(0)$ و $P_0(0)$ تعریف شده است. احتمالات P_1 و P_0 احتمالات حالت - ثابت نامیده می‌شوند. در پرورد طولانی سیستم را دارای حالت - ثابت گویند. یک سیستم از نظر آماری در زمان T :

$$P_0 = \mu / (\lambda + \mu)$$

احتمال خالی بودن سیستم:

$$P_1 = \lambda / (\lambda + \mu)$$

احتمال اشتغال سیستم:

مثال ۵-۱۰ در مدل صف $M/M/1/\infty$ با ورود $\lambda=2$ در هر ساعت و میانگین زمان سرویس $\mu^{-1}=2$ ساعت است و فرض در زمان ۰ سیستم خالی می‌باشد. بنابراین:

$$P_0(0) = 1, \quad P_1(0) = 0$$

$$P_0(t) = \frac{0.5}{2 + 0.3} + \left(1 - \frac{0.5}{2 + 0.3}\right) e^{-2.5t}$$

$$= 0.2 + 0.8e^{-2.5t}$$

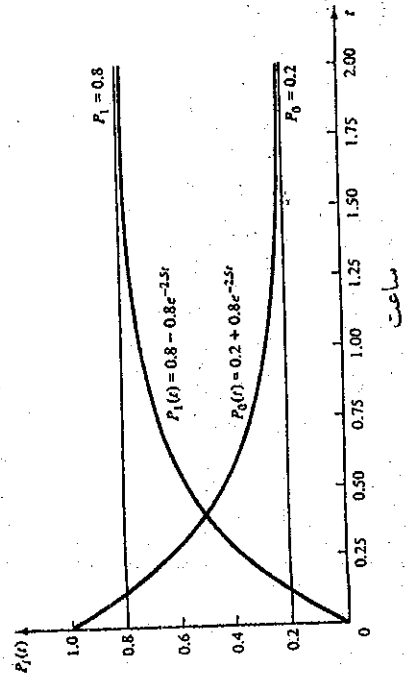
$$P_1(t) = \frac{2}{2 + 0.3} + \left(0 - \frac{2}{2 + 0.3}\right) e^{-2.5t}$$

$$= 0.8 - 0.8e^{-2.5t}$$

در حالت کلی $t \rightarrow \infty$ احتمال حالت - ثابت می‌شود:

$$P_0 = 0.2, \quad P_1 = 0.8$$

که $P_0(t) > P_0$ و $P_1(t) < P_1$ است $t \geq 0$.



شکل ۵-۱۰ احتمالات گذرا و احتمالات حالت ثابت صف $M/M/1/\infty$

نرخ مؤثر ورود، میانگین تعداد ورودیها است که به سیستم در هر ساعت وارد می‌شوند. مؤثر ورود در

۱۱۰ میانگین زمانی نفرت در خط انتظار.

$$TQ_0 = 1 + 2 = 3$$

$$TQ_1 = 2 + 2 + 4 + 4 = 12$$

$$TQ_2 = 2 + 2 = 4$$

$$TQ_3 = 1$$

$$LQ = (0(3) + 1(2) + 4(2) + 1(3)) / 20 = (21/20) = 1.05$$

میانگین زمانی نفرت در خط انتظار (بیکل ب. ۱۰-۶)

● میانگین زمان صرف شده در سیستم به ازای هر مشتری: (۳)

N تعداد مشتریان در انیای $[0, T]$ است.

W_i زمانی که وقت مشتری i در سیستم صرف می‌کند.

$$\bar{w} = \frac{1}{N} \sum_{i=1}^N W_i$$

$$\bar{w} = \frac{2 + 5 + 5 + 7 + 4}{5} = \frac{23}{5} = 4.6$$

(شکل ۱۰-۶)

● میانگین زمان صرف شده در خط انتظار به ازای هر مشتری

$$\bar{w}_0 = \frac{1}{N} \sum_{i=1}^N W_i \rho \rightarrow w_0 \quad N \rightarrow \infty$$

(شکل ۱۰-۶)

$$\bar{w}_0 = \frac{0 + 0 + 3 + 3 + 0}{5} = 1.2$$

$$\hat{\rho} \rightarrow \rho \quad T \rightarrow \infty$$

● اشتغال سرورس دهده:
(شکل ۱۰-۶)

$$\lambda_i = \lambda(1 - P_i) = 0.4$$

هر ساعت

$$\mu_i = \mu(1 - P_i) = (0.5)(0.8) = 0.4$$

در ساعت

نرخ مؤثر سرورس در واحد زمان:

آنالیز روی صف دارای تئوری بسیاری است و نتایج فشیه‌سازی صف را بررسی و ارزیابی می‌کند.

میانگین زمانی نفرت در سیستم (L)

تعداد مشتریان داخل سیستم در زمان t است. فرض در مثال زیر T کل زمان در انیای $[0, T]$ که

$$T_0 = 3, T_1 = 12,$$

$$\sum_{i=0}^{\infty} T_i = T$$

سیستم شامل مشتریان است.

$$T_2 = 4, T_3 = 1$$

که:

$$L = \frac{\sum_{i=0}^{\infty} iT_i}{T} = \sum_{i=0}^{\infty} i \left(\frac{T_i}{T} \right)$$

میانگین زمانی تعداد نفرت هست:

$$L = (0(3) + 1(12) + 2(4) + 3(1)) / 20 = 23/20 = 1.15$$

در شکل (الف ۱۰-۶)

T زمانی که سیستم دارای مشتریان است. در شکل (الف ۱۰-۶) $L(t)$ مجموع ارتفاع i و طول T

است. برای مثال ناحیه T_3 از زمان $t=7$ تا $t=8$ می‌باشد پس $T_3=1$.

$$\sum_{i=0}^{\infty} iT_i = \int_0^T L(t) dt$$

یعنی:

$$L = \frac{1}{T} \sum_{i=0}^{\infty} iT_i = \frac{1}{T} \int_0^T L(t) dt$$

$$L = \frac{1}{T} \int_0^T L(t) dt \rightarrow \bar{L} \quad T \rightarrow \infty$$

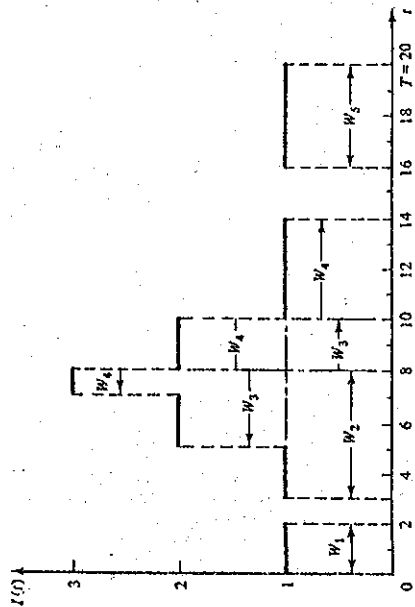
\bar{L} تقریبی از L است.

\bar{L}_0 میانگین زمانی تعداد مشتریان در خط انتظار L است:

$$\bar{L}_0 = \frac{1}{T} \sum_{i=0}^{\infty} iT_i \rho = \frac{1}{T} \int_0^T L_0(t) dt \rightarrow L_0 \quad T \rightarrow \infty$$

$$L_0(t) = \begin{cases} 0 & \text{if } L(t) = 0 \\ L(t) - 1 & \text{if } L(t) \geq 1 \end{cases}$$

L_0 : تعداد مشتریان در خط انتظار و T_0 کل زمان در انیای $[0, T]$ که مشتری i در خط انتظار است.



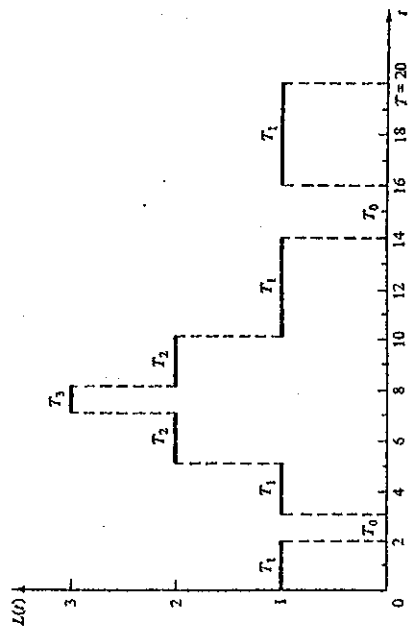
شکل ج ۱۰-۶ زمانهای سیستم W_i برای سیستم صف و یک سرویس دهنده

۱۰.۵ - شبیه سازی پیش آمدهای - گسسته

در اینجا قالبی برای مدلسازی سیستم های پیچیده و تعیین مزایای شبیه سازی پیش آمدهای گسسته ارائه می کنیم. این نوع مدلسازی برای شبیه سازی پیش آمد گسسته می باشد که پیش آمدهای وضعیتهای یک سیستم را تغییر می دهد. مدلسازی پیش آمد گسسته برای سیستمهایی مقتضی است که تغییر وضعیت سیستم در زمانهای گسسته رخ می دهد.

هر زمان شبیه سازی پیش آمد گسسته دید خاص خود را دارد. زمانی ممکن است با نگرش زمان بندی - پیش آمد یا نگرش ارتباط پروسه با مدلسازی پیش آمد گسسته باشد. در نگرش زمان بندی - پیش آمد نیاز به آن دارد که تحلیل گر روی یک موجودیت (مانند مشتری) متمرکز گردد و با توانایی پیش آمدها و فعالیتهای موجودیت از طریق سیستم عبور داده می شود زمان استفاده از یک زبان همه منظوره FORTRAN و C و BASIC یا ALGOL یا PASCAL یک شبیه ساز نگرش زمان بندی پیش آمد را انتخاب می کند. زبانهای مانند GASP از نگرش زمان بندی پیش آمد استفاده می کند در حالی که GPSS روش ارتباط پروسه را انتخاب کرده است. بعضی زبانهای جدید مانند SIMSCRIPT و SLAM اجازه می دهد که شبیه سازی با یکی از دوروش یا ترکیبی و بطور مقتضی استفاده کند.

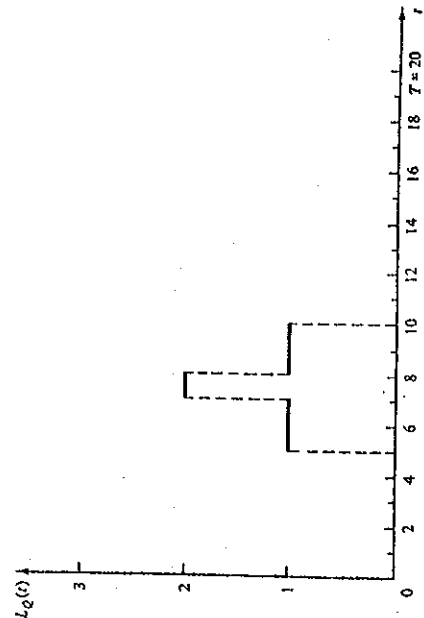
در این مرحله اساس زمان بندی - پیش آمد و نگرش ارتباط - پروسه بیان می شود و مثالهایی برای شبیه سازی آنها ارائه می کنیم.



شکل الف ۱۰-۶ تعداد مشتریان در سیستم $L(t)$ در زمان t

آنالیز صف با توزیعهای مختلف دارای فرمولهای ارزیابی می باشد که به عنوان نمونه برای یک صف $M/M/1/\infty$ ارائه گردید.

$$\bar{L}_q = \frac{\rho(1-\rho)}{1-\rho} = \frac{\rho(1-\rho)}{1-\rho} = 0.3$$



شکل ب ۱۰-۶ تعداد تفرقات خط انتظار $L_q(t)$ در زمان t

می باشد. شروع و پایان یک تاخیر یک پیش آمد شرطی نام دارد (شروع یک فعالیت با یک پیش آمد موتقی یا شرطی ممکن است باشد). واژه "پیش آمد" در این متن یک پیش آمد موتقی می باشد.

سیستمهایی که بیان می شوند پویا هستند و در طی زمان تغییر می کنند. بنابراین وضعیت سیستم، صفات موجودیتها و تعداد موجودیتهای فعال، محتوای سری ها و فعالیتها و تاخیرات موجود تماماً تابعی از زمان هستند و در طی زمان تغییر می کنند. زمان پویای یک متغیر که ساعت شبیه سازی Clock نام دارد بیان می شود.

مثال ۱-۰۶ - بررسی رستوران ماشین سواران

یک مدل پیش آمد - گسسته برای شبیه سازی آن یکار می گیریم. لذا اول باید مفاهیم اساسی را تعیین کنیم:

● حالت سیستم $LQ(t)$ ، تعداد ماشینهای منتظر که باید در زمان t سروس داده شوند.

$L_A(t)$ ، h یا تعیین آن است که پیش خدمت در زمان t یک کار یا مشغول می باشد.

$L_B(t)$ ، h یا تعیین آن است که آشپز در زمان t یک کار یا مشغول می باشد.

● موجودیتها، نه تنها مشتریان (یا ماشینها) بلکه سروس دهندگان نیز نیاز به ارائه واضح و روشن آنها است، غیر آن بصورت متغیرهای وضعیت بیان می شوند.

● پیش آمدها، پیش آمد ورود

پایان سروس بوسیله پیش خدمت

پایان سروس بوسیله آشپز

● فعالیتها، زمان بین ورود در جدول ۱-۱ (فصل قبلی) تعریف شده است

زمان سروس بوسیله پیش خدمت در جدول ۱۲-۹ تعریف شده است

زمان سروس آشپز در جدول ۱۳-۹ تعریف شده است
تاخیر، انتظار در صف تا پیش خدمت و آشپز آزاد شوند.

تعریف مفاهیم مدل یک شرح ثالثی از آن را فراهم می آورد. به علاوه، شرح ارتباطات دینامیکی و محاوره ای بین عناصر نیاز است. بعضی سوالهایی که نیاز به پاسخ دارند عبارتند از:

۱- چگونگی اثر هر پیش آمد روی وضعیت سیستم، صفات موجودیتها و محتوای سری ها است؟

۲- چگونه فعالیتها تعریف می شوند (مانند قطعی، تصادفی یا بعضی معادلات ریاضی دیگری)؟ چه پیش آمدهائی در آغاز و پایان هر فعالیت مشخص می شوند؟ فعالیت بدون ارتباط با وضعیت سیستم می تواند شروع شود یا شروع آن در سیستم در یک وضعیت معین است؟ (برای مثال یک فعالیت ماشین بدون آزاد بودن ماشین نمی تواند شروع شود).

۱-۰۵-۱ - مفاهیم پایه ای در شبیه سازی پیش آمد - گسسته

در این بخش مفاهیم یک سیستم و مدل سیستم تصادفی دینامیک (شامل زمان و عناصر تصادفی) که به روش گسسته تغییر می کند را شرح می دهیم. مفاهیم اساسی تعیین می گردد و سپس ارائه می شود مانند:

● سیستم، مجموعه ای از موجودیتها (مانند جمعیت و ماشینها) است که در طی زمان، با یکدیگر اثر متقابل دارند و اهداف خاصی را دنبال می کنند.

● مدل، ارائه یا بیان مجرد یک سیستم که معمولاً شامل ارتباط منطقی یا ریاضی است و یک سیستم را به صورت وضعیت، موجودیتها و صفات آنها، سری های پیش آمدها، فعالیتها و تاخیرات بیان می کند.

● حالت سیستم، مجموعه ای از متغیرها که شامل تمامی اطلاعات ضروری برای بیان مستقیم در هر زمان است.

● موجودیت، هر شیء یا عضو سیستم که ارائه واضح و روشن آن در مدل میسر می باشد (مانند یک سروس دهنده، یک مشتری یا یک ماشین).

● صفات، ویژگیهای یک موجودیت مشخص است (مانند اولویت یک مشتری منتظر).

● مجموعه، مجموعه ای از موجودیتهای مجتمع (موت یا دائمی) که بدلائل منطقی مورد نظر هستند (مانند تمامی مشتریان فعلی منتظر در خط انتظار که بصورت اولین ورودی اولین سروس را بگیرد یا بوسیله اولویت).

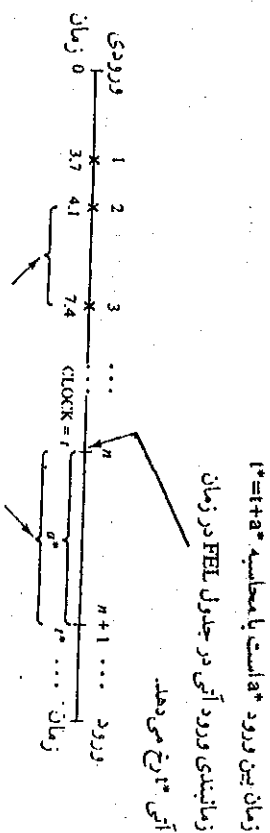
● پیش آمد، یک رخ داد آتی که وضعیت یک سیستم را عوض می کند (مانند ورود یک مشتری جدید).

● فعالیت، یک دوره زمانی با طول مشخص (زمان سروس) که طول آن همان مدت زمان انجام آن است (بصورت توزیعهای آماری تعریف می شود).

● تاخیر، یک دوره زمانی با طول نامشخص است که طول آن تا انتهایش ناشناخته می ماند (مانند تاخیر مشتری در خط انتظار که شروع آن بستگی به ورودی های بعدی دارد).

مجموعه ها اغلب لیست ها، صف ها یا رشته ها نام دارند. یک فعالیت می تواند قطعی (مانند زمان سروس که همیشه ۵ دقیقه است) یا تصادفی (مانند توزیع یکساخت 5 ± 3 دقیقه) یا دارای هر نوع تابع ریاضی باشد. گرچه دوره زمانی، یک فعالیت در شروع هر منظوری، داخل مدل قابل محاسبه است. یک تاخیر به عنوان نمونه وقتی که بعضی شرایط منطقی درست شوند، پایان می یابد که این شرایط منطقی معمولاً نتیجه پیش آمدهای بسیاری است. زمانی که یک مشتری در یک خط انتظار صرف می کند مثالی از تاخیر است. یک تاخیر اغلب یک انتظار شرطی نام دارد و در یک فعالیت انتظار غیر شرطی است. توجه این که پایان یک فعالیت یک پیش آمد است و اغلب یک پیش آمد موتقی

(در شکل ۸-۱۰ فرض شده که t^2 بین t^1 و t^2 است). تصویر سیستم در زمان صفر بوسیله تعریف و تولید مقدار اولیه است که پیش آمدهای بروزنا (Exogenous) نامیده می‌شود. مقدار دهی اولیه وضعیت سیستم در زمان صفر است. برای مثال در شکل ۸-۱۰ اگر $t=0$ باشد سپس وضعیت $(-)$ ممکن است تعداد اولیه مشتریان در سه نقطه مختلف سیستم را ارائه می‌کند. یک پیش آمد بروزنا یک رخداد "خارج سیستم" است که از سیستم برت می‌شود. یک مثال مهم ورود به یک سیستم صف است. در زمان صفر اولین پیش آمد ورود می‌شود و FEL زمانی می‌شود. زمان بین ورود یک مثال از یک فعالیت است. وقتی Clock به زمان اولین ورودی پیش می‌رود، دومین پیش آمد ورودی تولید می‌شود. اول یک زمان بین ورود a^2 تولید می‌گردد، این به عنوان پیش آمد ورود جدید در جدول FEL جای می‌گیرد. این روش تولید جویبار ورودی‌های خارجی، "boosting" نام دارد که در شکل ۸-۱۰ نشان داده شده است. اولین سه زمان ورودی $7/3$ و $4/3$ و $3/2$ واحد زمان است. شروع و پایان بین ورودها مثالهایی از پیش آمدهای حتمی هستند.



بین پیش آمدهای ورود
انواع دیگر پیش آمدها ممکن است رخ دهد.

شکل ۸-۱۰ تولید جویباری از ورودی‌های خارجی بوسیله روش bootstrapping

گام ۱. برداشتن پیش آمد (پیش آمد t^2 زمان t^1) از جدول FEL
گام ۲. پیشبرد زمان به زمان پیش آمد (clock از t^1 به t^2).

گام ۳. اجرای پیش آمد (۳) بروز آوردن وضعیت سیستم، تغییر صفات موجودیت و مرتب‌سازی اعضا طبق منطق.

گام ۴. تولید پیش آمدهای بعدی (در صورت نیاز) و قرار دادن در مکان مناسب FEL.

گام ۵. بروز آوردن آمارها و شمارنده‌ها. مثال پیش آمد t^2 در زمان t^1 رخ داده جایی که $t^2 < t^1$
یک تصویر جدید سیستم در زمان t^2

| وضعیت سیستم | ... | لیست پیش آمدهای آتی | |
|-------------|-----------|--|--|
| CLOCK | | | |
| ۱ | (5, 1, 5) | $(1, t^1)$ پیش آمد نوع ۱ در زمان t^1 رخ می‌دهد.
$(2, t^2)$ پیش آمد نوع ۲ در زمان t^2 رخ می‌دهد.
$(4, t^4)$ پیش آمد نوع ۴ در زمان t^4 رخ می‌دهد.
$(1, t^1)$ پیش آمد نوع ۱ در زمان t^1 رخ می‌دهد. | |
| | | پیش آمد نوع ۲ در زمان t^2 رخ می‌دهد. $(2, t^2)$ | |

شکل ۸-۱۰ پیشبرد زمان شیوه‌سازی و بروز آوردن تصویر سیستم

علاوه بر FEL تمامی سری‌ها در مدل به منظورهای خاصی نگهداری می‌شوند و عملیات افزودن و کاهش موجودیتها از سری، نیاز به تکنیکهای کارای پردازش حلست دارد.

در شکل ۸-۱۰ افزودن و کاهش پیش آمد از FEL آمده است. پیش آمد t^2 با زمان رخداد t^1 ارائه شده، می‌گوید که یک پیش آمد سروس در سروس دهده t^2 پایان یافته است. از این جهت پیش آمد حتمی در زمان t^2 از جدول FEL در گام ۱ الگوریتم پیشبرد زمان از مانبندی پیش آمد برداشته شده است (شکل ۸-۱۰). وقتی پیش آمد t^2 (پیش آمد ورود) با زمان t^2 در گام ۲ تولید می‌شود، یک راه تعیین مکان مناسب در جدول FEL جستجوی بالا به پایین است:

اگر $t^2 > t^1$ باشد پیش آمد t^2 راس لیست FEL قرار گیرد.

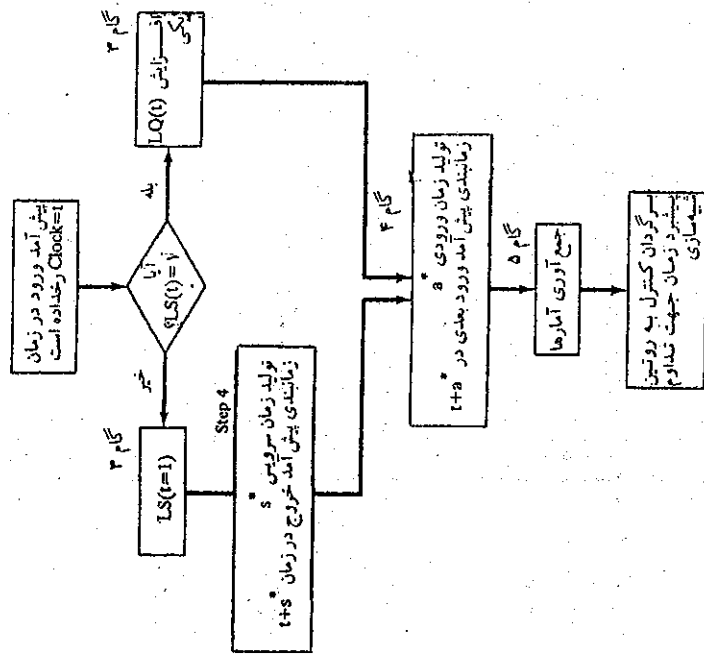
اگر $t^2 < t^1$ و $t^2 < t^3$ باشد پیش آمد t^2 دومین در فهرست FEL قرار گیرد.

اگر $t^2 < t^1$ و $t^2 < t^3$ و $t^2 < t^4$ باشد پیش آمد t^2 سومین در فهرست FEL قرار گیرد.

اگر $t^2 < t^1$ و $t^2 < t^3$ و $t^2 < t^4$ و $t^2 < t^5$ باشد پیش آمد t^2 آخرین در FEL قرار گیرد.

صف MQ در طول پیشرفت شبیهسازی جمع آوری می شود. (هر S^* زمانهای بین ورود و سرویس است.)

به همان سرعت که تصویر سیستم در زمان $Clock=0$ کامل شد، شبیهسازی شروع می شود. در زمان صفر پیش آمد حتمی (D,4) است. $Clock$ به زمان ۴ تغییر می کند و (D,4) از جدول FEL خارج می شود. $IS(1)=1$ و $0 < t < 4$ و سرویس دهنده ۴ دقیقه مشغول است لذا مجموع زمان از $B=0$ به $B=4$ افزایش می یابد. با استفاده از منطق شکل ۱۰-۱۱، $IS(4)=0$ می شود. (سرویس دهنده بی کار می گردد). جدول FEL با فقط دو پیش آمد آتی (A,8) و (E,60) می ماند. $Clock$ شبیهسازی به زمان ۸ تغییر و پیش آمد ورود اجرا می شود. تفسیر بقیه جدول ۱۰-۲ بعهدہ خواننده می باشد.



شکل ۱۰-۱۰ زیر برنامه پیش آمد ورود

۱۰-۶- راهنمای شبیهسازی با استفاده از زمانبندی پیش آمدها

سیستم صف شامل مشتریانی است که در خط انتظار قرار می گیرند سپس سرویس می یابند. مدل اجزای زیر را دارد:

وضعیت سیستم، زوج $(LQ(t), IS(t))$ چایی که $LQ(t)$ تعداد مشتریان در خط انتظار و $IS(t)$ تعداد آنهاایی که در زمان t سرویس دیده اند (0 یا 1).

موجودیتها، سرویس دهنده و مشتریان بطور واضح مدل نشده اند بلکه به صورت متغیرهای وضعیت بالا بیان می شوند.

پیش آمدها: - ورود (A)

- خروج (D)

- پیش آمد پایان شبیه سازی (E) زمانبندی شده که در زمان $t+60$ رخ می دهد.

فعالیتها:

- زمان بین ورود طبق جدول ۹-۶ تعریف شده است.

- زمان سرویس، طبق جدول ۹-۷ تعریف شده است.

- تأخیر، زمانی که مشتری در خط انتظار صرف می کند.

پیش آمدها در FEL بصورت (زمان پیش آمد و نوع پیش آمد) نوشته شده است. در این مدل همیشه FEL شامل حداقل دو یا سه پیش آمد می باشد. اثرات ورود و خروج در شکلهای ۱۰-۱۰ و ۱۰-۱۱ نشان داده شده است. جدول شبیهسازی این مسئله در جدول ۱۰-۲ آمده است. خواننده باید تمامی تصاویر انتظاری سیستم را پوشش دهد، با اولین شروع می شود و تصویر بعدی سیستم را از قبلی و منطق پیش آمد را از شکل های ۱۰-۱۰ و ۱۰-۱۱ بدست می آورد. زمان بین ورود و سرویس ها بصورت زیر است:

| | | | | | | | |
|---------------|---|---|---|---|---|---|-----|
| زمان بین ورود | 8 | 6 | 1 | 8 | 3 | 8 | ... |
| زمان سرویس | 4 | 1 | 4 | 3 | 2 | 4 | ... |

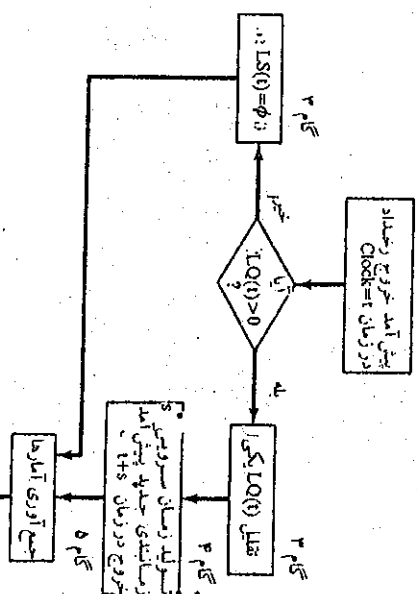
بافتادیر اولیه ورود مشتری در زمان صفر و سرویس آن عملیات شروع می گردد. این امر در جدول ۱۰-۲ منعکس شده است و با تصویر سیستم در زمان صفر ($Clock=0$) یا $IS(0)=1$ و $LQ(0)=0$ و یک پیش آمد خروج و پیش آمد ورود زوجی FEL انجام می شود. همچنین شبیهسازی برای توقف در زمان ۶۰ زمانبندی می شود.

فقط دو آمار اشغال سرویس دهنده و حداکثر طول صف جمع آوری خواهد شد. در صد اشغال سرویس دهنده بوسیله تقسیم زمان اشغال (B) بر کل زمان (TE) بدست می آید. کل زمان اشغال B و حداکثر طول

| آمارها | | | لیست پیش آمدهای آنی | وضعیت سیستم | | ساعت شبیه سازی |
|--------|---|----|-------------------------|-------------|-------|----------------|
| | B | MQ | | LS(t) | LQ(t) | |
| 0 | 0 | 0 | (D, 4) (A, 8) (E, 60) | 1 | 0 | 0 |
| 4 | 0 | 0 | (A, 8) (E, 60) | 0 | 0 | 4 |
| 4 | 0 | 0 | (D, 9) (A, 14) (E, 60) | 1 | 0 | 8 |
| 5 | 0 | 0 | (A, 14) (E, 60) | 0 | 0 | 9 |
| 5 | 0 | 0 | (A, 15) (D, 18) (E, 60) | 1 | 0 | 14 |
| 6 | 1 | 1 | (D, 18) (A, 23) (E, 60) | 1 | 1 | 15 |
| 9 | 1 | 1 | (D, 21) (A, 23) (E, 60) | 1 | 0 | 18 |
| 12 | 1 | 1 | (A, 23) (E, 60) | 0 | 0 | 21 |

جدول ۱۰-۲ جدول شبیه سازی صف با یک سرور پیوسته

کلیه مدل‌های شبیه سازی گسترده که با FORTRAN نوشته می شوند در بر دارنده اجزاء معرفی شده وضعیت سیستم، موجودیت، صفات، سری ها، پیش آمدها، فعالیتها و تأخیرها است، مضافاً اجزائی که فهرست وار در زیر آمده را شامل می شوند. جهت تسهیل طراحی مدل و رفع ایرادهای آن بهتر این است که مدل‌های فورتران را به صورت غیر متمرکز و با استفاده از زیر برنامه های متعدد به وجود



شکل ۱۰-۱۱ زیر برنامه پیش آمد خروج

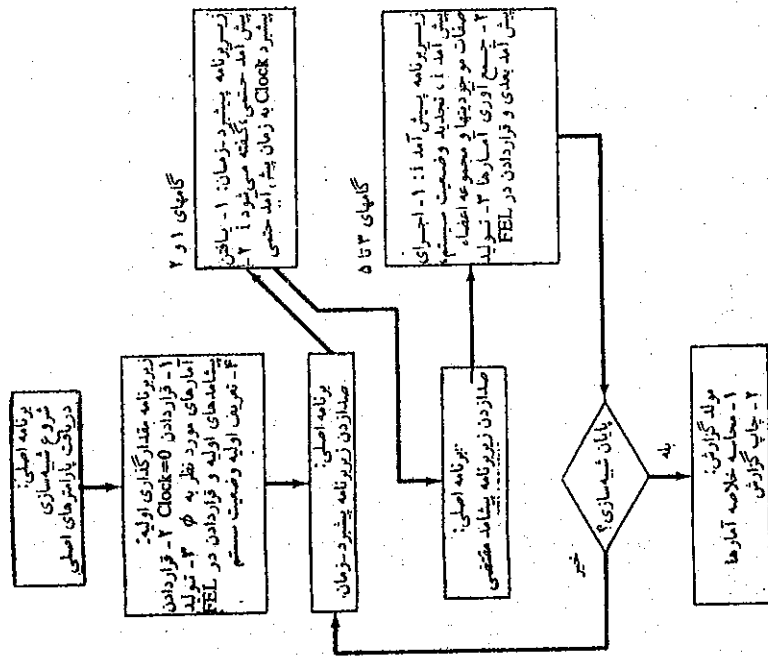
شبیه سازی در جدول ۱۰-۲ از زمان بین ۱۰،۲۱ تا ۱۰،۲۱ می پرشاند. در زمان شبیه سازی ۲۱، سیستم خالی است. اما ورود بعدی در زمان آنی ۲۲ رخ می دهد. سرورس دهنده برای ۱۲ واحد از ۲۱ واحد از ۲۱ واحد زمان شبیه سازی شده مشغول بوده و حداکثر طول صف یک بود.

۱۰.۷ - شبیه سازی با زبانهای مختلف

در این بخش جهت مقایسه و بررسی معایب و مزایای زبانهای عمومی با زبانهای شبیه سازی به مثالهای شبیه سازی با آنها می پردازیم.

۱۰.۷.۱ - شبیه سازی با استفاده از FORTRAN

FORTRAN از متداولترین و سهل الوصول ترین زبانهای عمومی است که در شبیه سازی نیز مورد استفاده قرار می گیرد. این زبان هیچگونه امکاناتی را به طور مستقیم در زمینه تسهیل شبیه سازی در اختیار شبیه ساز قرار نمی دهد به عبارت دیگر، شبیه ساز باید کلیه برنامه های مربوط به زمان بندی پیش آمدها، پیشبرد زمان، تولید نمونه های مورد نیاز توزیع مختلف آماری، جمع آوری اطلاعات و ارائه گزارش ها را خود پیروید. از این رو استفاده از FORTRAN برای شبیه سازی سیستمهای پیچیده بسیار مشکل است. استفاده از FORTRAN (یا سایر زبانهای عمومی) در زمینه شبیه سازی سیستمهای کوچک میتواند عامل مهمی جهت فراگیری جزئیات طراحی الگوریتم زمان بندی پیش آمدها و پیشبرد زمان باشد. در واقع زبانهای شبیه سازی به خاطر طبیعت شان بسیاری از ظرایف کارهای مورد بحث را از دیده شبیه سازینهان نگه می دارند.



شکل ۱۰-۱۲ ساختار کلی برنامه شبیه‌سازی زمانبندی پیش‌آمدها

توجه این که تمامی احوال در زیربرنامه پیش‌آمد در یک لحظه زمانی شبیه‌سازی انجام می‌شود. مقدار متغیر Clock در زیربرنامه یک پیش‌آمد تغییر نمی‌کند. اگر شبیه‌سازی تمام نشده باشد، کنترل مجدداً به زیربرنامه پیش‌آمد زمان می‌شود. سپس اجرای زیربرنامه پیش‌آمد مقتضی و مانند آن. وقتی شبیه‌سازی تمام می‌شود کنترل به مولد گزاش می‌شود تا خلاصه آمارهای مورد نیاز یک گزارش چاپ شود. کاراتی یک مدل شبیه‌سازی کامپیوتری بستگی به تکنیک استفاده و پیاده‌سازی FEL و دیگر مجموعه‌ها دارد. برداشتن پیش‌آمد و افزودن پیش‌آمد جدید، دو عمل اصلی روی FEL است.

مثال ۱۰-۶ شبیه‌سازی مسئله صف و یک سرویس دهنده با استفاده از FORTRAN -
در اینجا به شبیه‌سازی مسئله صف می‌پردازیم. حالت خاصی از این مساله است. پیش‌آمد ورود به سیستم و خروج آن را به ترتیب پیش‌آمدهای نوع ۲۱ نامگذاری می‌شوند. متغیر IMVET تعیین‌کننده نوع پیش‌آمد وقوع است. زیربرنامه‌های این مدل و جریان کنترل برنامه به وسیله شکل ۱۰-۱۳

آورد. تقریباً کلیه مدل‌هایی که به فرتون نوشته می‌شوند، دارای اجزاء زیر هستند.

ساعت شبیه‌سازی (Clock)، یک متغیر برای تعریف ساعت شبیه‌سازی است و زمان را تعریف می‌کند. زیربرنامه مقداردهی اولیه، یک زیربرنامه برای تعیین وضعیت سیستم در زمان صفر است.

زیربرنامه پیش‌آمد زمان، یک زیربرنامه که جدول FEL را برای تعیین پیش‌آمد بعدی (که نام پیش‌آمد حتمی دارد و بوسیله IMEVT ارائه می‌شود) جستجو می‌کند و کلاک را به زمان وقوع پیش‌آمد حتمی

پیش می‌برد.

زیربرنامه زمانبندی، زیر برنامه‌ای که پیش‌آمدهای تولیدی آتی را در جدول FEL قرار می‌دهد (در اینجا نشان داده نشده است).

زیربرنامه‌های پیش‌آمدها، برای هروقت پیش‌آمد یک زیر برنامه است که وقتی پیش‌آمد رخ می‌دهد، وضعیت سیستم را تجدید می‌کند.

مولد‌های تصادفی، روش‌های تولید نمونه‌های تصادفی با توزیع مورد نیاز است.

برنامه اصلی، تمامی کنترل‌های الگوریتم زمانبندی -پیش‌آمدها را فراهم می‌کند.

مولد گزاش، روشی که خلاصه آمارها برای جمع‌آوری و چاپ یک گزارش در انتهای شبیه‌سازی را محاسبه می‌کند.

ساختار برنامه اصلی شبیه‌سازی فرتون در شکل ۱۰-۱۲ نشان داده شده است. این فلوچارت توسعه‌ای از الگوریتم پیش‌آمد در زمان/زمانبندی -پیش‌آمدها است.

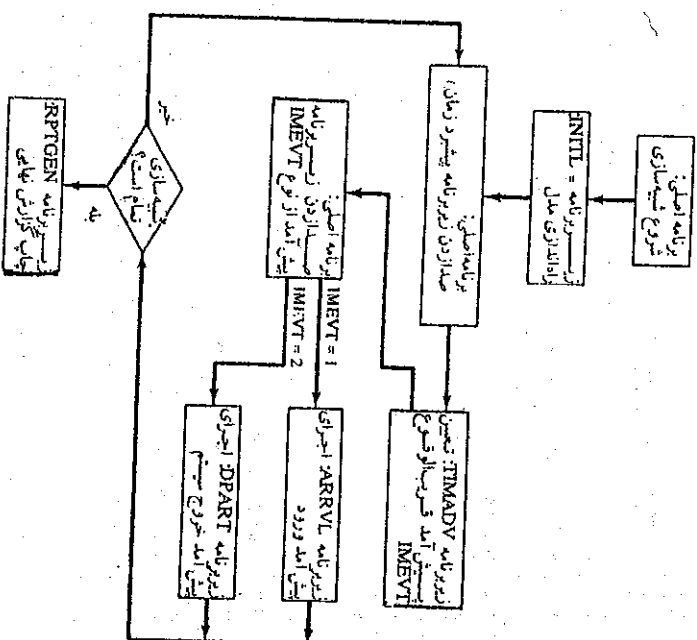
شبیه‌سازی با مقداردهی Clock به صفر، مقدار اولیه آمارها به صفر، تولید پیش‌آمدهای اولیه (حداقل یکی) و قرار دادن در جدول FEL و تعریف وضعیت سیستم در زمان صفر شروع می‌شود. برنامه شبیه‌سازی سپس بین زیربرنامه‌های پیش‌آمد زمان و زیربرنامه پیش‌آمد مقتضی، بصورت دوره‌ای حرکت می‌کند تا شبیه‌سازی پایان یابد. زیربرنامه پیش‌آمد در زمان، فهرست FEL را برای یافتن پیش‌آمد جستجو می‌کند که یک پیش‌آمد نوع گزاش می‌شود. ساعت شبیه‌سازی Clock به زمان پیش‌آمد حتمی جلو می‌رود و سپس زیربرنامه پیش‌آمد اجرا صدا زده می‌شود، آمارها بروز می‌شوند و پیش‌آمدهای آینده را تولید می‌کند (در فهرست FEL قرار می‌گیرد). با اجرای پیش‌آمد حتمی وضعیت سیستم، صفات موجودیت و مجموعه عناصر تغییر می‌کنند که این اثر مطابق رخداد پیش‌آمد است.

| متغیرها | متغیرهای وضعیت سیستم: | شرح |
|--|--|--|
| LOT
LST | متغیرهای صفات موروثیها و خصوصیات:
CHROUT0
CHROUT1
CHROUT2
CHROUT3 | مقدار مقادیری که در لحظه جاری شیبه سازی در صف انتظار قرار دارند.
مقدار مقادیری که در لحظه جاری شیبه سازی در حال دریافت سرورس هستند (۱۰ بار) |
| (FEB)
FEB0
FEB1
FEB2
FEB3 | متغیرهای فهرست پیش آمدهای آتی:
(FEB)
FEB0
FEB1
FEB2
FEB3 | لحظه ورود مشتری که در لحظه فعلی در حال دریافت سرورس است.
لحظه ورود (t-1) یعنی مشتری (تایم ترتیب CHROUT3) نشان دهنده زمان ورود یک مشتری است که فعلا در حال انتظار برای دریافت سرورس به سر میرود (t) |
| LAT
SYT | متغیرهای دوره فعالیها:
LAT
SYT | زمان بین ورود مشتری قبلی و ورود مشتری بعدی
طول زمان ارائه سرورس به آخرین مشتری |
| CLOCK
NUMEVS | متغیرهای شیبه سازی:
CLOCK
NUMEVS | لحظه فعلی زمان شیبه سازی
تعداد انواع وقایع (در اینجا ۲) |
| B
TILE
MQ
MS
S
ND
F | متغیرهای گردآوری اطلاعات:
B
TILE
MQ
MS
S
ND
F | مدت اشغال سرورس دهنده، تا لحظه فعلی
لحظه رویداد آخرین پیش آمد برای تجدید B مورد استفاده قرار می گیرد.
حداکثر طول صف انتظار تا لحظه فعلی
جمع مدت به سر بردن در سیستم توسط مشتری که تا این لحظه سیستم را ترک کرده اند
تعداد مشتریانی که تا این لحظه ۴ دقیقه یا بیشتر در سیستم به سر برده اند |
| RHQ=B/CLOCK
MQ
AVRG=(S/ND)
PQ4=F/ND | متغیرهای خلاصه کردن اطلاعات:
RHQ=B/CLOCK
MQ
AVRG=(S/ND)
PQ4=F/ND | درصد زمان اشغال سرورس دهنده (CLOCK) مقدار نهایی آن است).
حداکثر طول صف
متوسط مدت زمان راه اندازی یک مشتری
درصد مشتریانی که ۴ دقیقه یا بیشتر در سیستم به سر برده اند |
| M4T
MSVT
SIGMA
NCUST | پارامترهای ورودی:
M4T
MSVT
SIGMA
NCUST | میانگین مدت بین ورود (۴/۵ دقیقه)
میانگین مدت ارائه سرورس (۳/۷ دقیقه)
انحراف معیار مدت ارائه سرورس (۱/۰ دقیقه)
قطع شیبه سازی ۱۰۰۰۰ بار شدن تعداد مشتریان راه اندازی شده |
| EXRON (FMEAN)
NORMAL (COMU, SIGMA) | متغیرهای وضعیت سیستم:
EXRON (FMEAN)
NORMAL (COMU, SIGMA) | شرح |
| INTILE
TMADV
ARRVL
DPART
RPTGEN | متغیرهای صفات موروثیها و خصوصیات:
INTILE
TMADV
ARRVL
DPART
RPTGEN | این تابع، یک نمونه تصادفی با تابع نهایی با میانگین FMEAN تولید می کند.
این تابع برای تابع چگالی نرمال با میانگین XMD و انحراف معیار SIGMA مقدار تصادفی تولید می کند. |
| ZIRORNAME | متغیرهای دوره فعالیها:
ZIRORNAME | نام برنامه مربوط به راه اندازی مدل |

نشان داده شده است. این نمودار در واقع نحوه به کار بردن شکل ۱۰-۱ را عرضه می‌کند. جدول ۳-۱ نیز کلیه متغیرهای مربوط به وضعیت سیستم صفات مورچو دیتها و مجموعه‌ها، دوره فعالیتها، محله‌ای ذخیره و اطلاعات گردآوری شده را نشان می‌دهد.

این جدول نام کلیه زیربرنامه‌ها از جمله توابع تولید مقدار تصادفی از دو تابع چگالی نمائی منفی و نرمال نیز درج شده است.

برنامه اصلی که در شکل ۱۰-۱۴ نمایش داده شده است، نظارت کلی بر جریان گردش کار الگوریتم زمان‌بندی پیش آمدها و پیشبرد زمان برعهده دارد. متغیرهایی که مقادیر شان پیوسته برای کلیه زیربرنامه‌ها معلوم است در جملات COMMON با نامهای SIM و TIMEKP ارائه شده‌اند. برنامه اصلی زیربرنامه‌ها نمایش داده شده در شکل ۱۰-۱۳ پیروی میکند. ابتدا تعداد پیش آمدها NUNMEVS از مسطح نمایش داده شده در شکل ۱۰-۱۳ پیروی میکند. ابتدا تعداد پیش آمدها NUNMEVS و پارامترهای ورودی مانند SIGMA, MSVT, MLAT و NCUST مشخص میشوند. معیاراً اختیار به برنامه NINTEZ وگذار میشود تا عمل راه‌اندازی مدل را انجام دهد.



شکل ۱۳-۱۰ ساختار برنامه شبیه‌سازی صف و یک سرور دهنده به FORTRAN

```

PROGRAM FTESIM(OUTPUT,TAPE6=OUTPUT)
C
C
C MAIN PROGRAM
C
1-INITIALIZES MODEL
2-CALLS TIME-ADVANCE AND EVENT ROUTINES
3-CALLS REPORT GENERATOR TO TERMINATE SIMULATION NORMALLY
C
REAL MIAT,MSVT
COMMON /SIM/ MIAT,MSVT,SIGMA,NCST,LOT,LST,TLE,
1 CHECHT(100),B,MQ,S,P,NO
COMMON /TIMEKP/ CLOCK,INSETT,NUMEVS,PSL(2)
NUMEVS=2
C
C ASSIGN VALUES TO INPUT PARAMETERS
C (THESE VALUES COULD BE STORED IN A FILE
C AND READ INTO THE PROGRAM, FOR CONVENIENCE.)
C
MIAT = 4.5
MSVT = 3.2
SIGMA = .6
NCST = 1000
C
C CALL INITIALIZATION ROUTINE
C
CALL INITLZ
C
C CALL TIME-ADVANCE ROUTINE TO DETERMINE IMMINENT
C EVENT AND ADVANCE CLOCK TO THE IMMINENT
C EVENT TIME.
C
30 CALL TIMADV
C
C THE VARIABLE "INSETT" INDICATES THE IMMINENT
C EVENT. INSETT = 1 FOR AN ARRIVAL.
C INSETT = 2 FOR A DEPARTURE.
C
GO TO (40,50),INSETT
C
C CALL THE APPROPRIATE EVENT SUBROUTINE
C
40 CALL ARRVL
C
GO TO 30
C
50 CALL DPART
C
C CHECK TO SEE IF SIMULATION IS OVER.
C IF NOT, RETURN TO TIME-ADVANCE ROUTINE
C
IF (ND .LT. NCST) GO TO 30
C
C WHEN SIMULATION IS OVER, CALL REPORT GENERATOR
C
CALL RPTGEN
STOP
END

```

شکل ۱۰-۱۲ برنامه اصلی FORTRAN شبیه‌سازی مسئله صف در یک سرویس‌دهنده

سپس اختیار به زیربرنامه TIMADV و اگر کار می‌گردد تا به تعیین پیش آمد قریب‌الوقوع پرداخته و زمان شبیه‌سازی را به رویداد پیش آمد مزبور جلو ببرد. در مرحله بعد اختیار به زیربرنامه پیش آمد مورد نظر و اگر کار می‌شود. اگر مقدار عددی IMVET مساوی ۱ باشد، جمله GO TO اختیار را به CALL ARRVL و اگر کار می‌کند و در صورتیکه $IMEVT=2$ باشد اختیار به CALL DPART منتقل می‌شود.

پس از آنکه پیش آمد اجرا شد آیا باید شبیه‌سازی را پایان داد یا خیر؟ در صورتیکه شرایط متوقف کردن شبیه‌سازی هنوز فراهم نشده باشد، اختیار بر برنامه آفندر بین زیربرنامه‌های مربوط به پیشبرد زمان و اجرای پیش آمدها مبادله می‌شود تا شرایط به اتمام رساندن شبیه‌سازی فراهم شود. با اتمام شبیه‌سازی، اختیار به زیربرنامه RPTGEN منتقل می‌گردد تا گزارش نهایی تهیه و چاپ شود.

برنامه اصلی MAIN PROGRAM

۱- مدل را راه‌اندازی می‌کند

۲- زیربرنامه‌های پیشبرد زمان و اجرای پیش آمدها را فعال می‌کند.

۳- به منظور اتمام شبیه‌سازی زیربرنامه گزارش‌نویس را فعال می‌کند.

پارامترهای ورودی تعیین می‌شوند (این عمل متوالی از طریق ثبت مقادیر مزبور در یک فایل و خوانده شد نشان توسط برنامه صورت گیرد. زیربرنامه راه‌اندازی مدل فعال می‌شود.

زیربرنامه پیشبرد زمان برای تعیین پیش آمد قریب‌الوقوع و جلو بردن ساعت شبیه‌سازی به لحظه رویداد آن فعال می‌شود.

متغیر IMEVT نوع پیش آمد قریب‌الوقوع را مشخص می‌کند. اگر این برنامه مساوی ۱ باشد پیش آمد از نوع ورود و اگر مساوی ۲ باشد از نوع خروج از سیستم است. زیر برنامه پیش آمد مورد نظر فعال می‌شود. اگر شبیه‌سازی به اتمام نرسیده است زیربرنامه پیشرو زمان فعال می‌شود و اگر نه زیر برنامه گزارش نویس فعال می‌شود.

شکل ۱۰-۱۵ فهرست زیربرنامه INTILZ را نشان می‌دهد. در این زیربرنامه مقادیر اولیه ساعت شبیه‌سازی، وضعیت سیستم، و سایر متغیرها تعیین می‌گردد. باید توجه داشت که زمان رویداد پیش آمد اولین ورود به سیستم توسط تابع FUNCTION EXPON تولید و در محل FEL(1) ذخیره می‌شود.

```

C
C
C INITIALIZATION ROUTINE
C
SUBROUTINE INIZLZ
REAL NMT, NSVT
COMMON /SIM/ NMT, NSVT, SIGMA, NCST, LQT, LST, TLE,
1 CHECKOUT(100), B, NQ, S, F, ND
COMMON /TIMEEV/ CLOCK, IMEVT, NMEVTS, FEL(2)
C
C INITIALIZE SIMULATION
C 1-SET SIMULATION CLOCK TO ZERO
C 2-ASSURE SYSTEM IS EMPTY AND IDLE
C 3-INITIALIZE CUMULATIVE STATISTICS TO 0.
C
CLOCK=0.0
- NMT=0
- LQT=0
- LST=0
- TLE=0
- B=0
- NQ=0
- S=0
- F=0
- ND=0
C
C GENERATE TIME OF FIRST ARRIVAL, LQT, AND
C SCHEDULE FIRST ARRIVAL IN FEL(1).
C SET FEL(2) TO "IMPLICIT" TO INDICATE THAT
C A DEPARTURE IS NOT POSSIBLE WHILE THE
C SYSTEM IS EMPTY.
C
FEL(1)=CLOCK + EXPON(NMT)
FEL(2)=1.0E-30
RETURN
END

```

شکل ۱۵-۱- زیربرنامه راهاندازی مدل FORTRAN شبیه‌سازی

فرضهای دیگری در زمینه راهاندازی شبیه‌سازی را نیز میتوان به مدل وارد کرد.

شکل ۱۵-۱۶ فهرست زیربرنامه پیشبرد زمان را نشان می‌دهد. این زیربرنامه به منظور پردازش فهرست‌پیش‌آمدهای آنسی (FEL) به جستجوی عناصر زیردار FEL(1) یعنی FEL(2)، FEL(1)، FEL(2)، FEL(1) می‌پردازد تا زودترین پیش‌آمد را که دارای اندیس IMEVT است پیدا کند. پیش‌آمد IMEVT پیش‌آمد قریبالوقوع است. زمان رویداد آن FEL (IMEVT) است. متعاقباً، زمان شبیه‌سازی (CLOCK) به زمان رویداد پیش‌آمد قریبالوقوع جلو برده شده و اختیار به برنامه اصلی واگذار می‌گردد. اگر کلیه عناصر برودار FEL (1) مساوی بینهایت باشند، بیغایمی اگر مبتنی بر خالی بودن فهرست پیش‌آمدهای آنسی و عدم ادامه شبیه‌سازی توسط برنامه چاپ می‌شود و در چنین شرایطی اختیار برنامه به زیربرنامه گزارش نویسی واگذار می‌شود.

زیربرنامه مربوط به پیشبرد زمان (توضیحات برنامه)

- این زیربرنامه، واقعه بعدی را از فهرست و تابع آنسی یافته و زمان شبیه‌سازی را به جلو میرود. جستجوی فهرست و تابع آنسی به منظور یافتن واقعه بعدی صورت می‌گیرد.

زیربرنامه راهاندازی مدل

SUBROUTINE INIZLZ

شبیه‌سازی آغاز می‌شود:

۱- ساعت شبیه‌سازی صفر می‌شود.

۲- فرض می‌شود سیستم در لحظه صفر خالی از مشتری و متوقف است.

۳- محلهای گردآوری اطلاعات صفر می‌شوند.

زمان اولین ورود تولید شده (LAT) و در FEL(1) قرار می‌گیرد.

۲- مساوی با بینهایت فرض می‌شود تا به این معنی باشد که با یک سیستم خالی از مشتری امکان

رویداد واقعه خروج سیستم وجود ندارد.

به این ترتیب فرض خالی بودن سیستم از مشتری در لحظه $CLOCK=0$ است. از این نظر، زمان رویداد

پیش‌آمد بعدی خروج سیستم، یعنی FEL(2) بینهایت فرض می‌شود. در برنامه فوق، این مقدار $1.0E-30$ است. در نتیجه اولین پیش‌آمدی که روی میدهد یک پیش‌آمد ورود است. از سویی دیگر در صورتیکه فرض کنیم در لحظه صفر یک متقاضی در سیستم وجود دارد و ارائه سروس به او آغاز شده است. انجام تغییرات زیر در برنامه ضروری می‌شود:

LST=1

CHKOUT(1)=CLOCK

FEL(2)=CLOCK + NORMAL(MSVT, SIGMA)

زیر برنامه DPART از این صفت استفاده خواهد شد تا مدت به سرپرست بیشتری در سیستم محاسبه شود.

زیر برنامه پیش آمد ورود

آیا سرویس دهنده مشغول است؟

سرویس دهنده آزاد است. وضعیت سیستم تجدید شده و زمان ورود مشتری ثبت می گردد.

مدت ارائه سرویس به مقاضی تازه وارد تولید شده و زمان خروج از سیستم برای اوتعیین می شود.

اطلاعات گردآوری شده تجدید می شود.

سرویس دهنده مشغول است. وضعیت سیستم تجدید شده و زمان ورود مشتری جدید ثبت می شود.

اطلاعات گردآوری شده در قالب B و MQ تجدید می شود. با رویداد پیش آمد ورود F.S و ND

تجدید نمی شوند.

یک مقدار تصادفی برای فاصله بین دو ورود تولید شده و پیش آمد ورود بعدی برنامه ریزی می شود.

اشتباه از اندازه در نظر گرفته شده برای بردار CHKOUT تجاوز شده و اندازه این بردار را باید

افزایش داد.

نظر به اینکه ارائه خدمت به مشتری جدید شروع شده است، با استفاده از FUNCTION NORMAL در یک مقدار تصادفی برای مدت ارائه سرویس (SVT) تولید شده و با قرار دادن لحظه خروج سیستم در FEL(2) یک پیش آمد خروج سیستم برنامه ریزی می شود. توجه به این نکته نیز ضروری است که مقدار FEL(2) از جمع زمان فعلی شبیه سازی (CLOCK) و طول مدت زمان ارائه سرویس که تازه شروع شده (SVT) به دست می آید. در اینجا، اختیار برنامه به دستور شماره 100 (شکل ۱۰-۱۷) واگذار می شود تا از طریق FUNCTION EXPON یک مقدار تصادفی برای فاصله زمانی بین آخرین ورود تا ورود بعدی (IAT) تولید شده و پیش آمد ورود بعدی از طریق برآورد CLOCK+IAT و سپردن آن به FEL(1) برنامه ریزی شود. متعاقب این عمل کنترل به برنامه اصلی واگذار میگردد. از طرف دیگر اگر در لحظه رخ داد پیش آمد ورود، سرویس دهنده مشغول باشد (LST=1) اختیار برنامه به دستور شماره 20 (شکل ۱۰-۱۷) منتقل خواهد شد. در اینجا به تعداد مشتریان حاضر در صف انتظار (LOT) یک واحد افزوده شده و صفت "زمان ورود متقاضی در انتهای مجموعه CHKOUT ثبت می شود (تعداد) محلهای ذخیره اطلاعات برای بردار CHKOUT مساوی 100 در نظر گرفته شده است.

اشتباه - فهرست وقایع آتی خالی است.

ساعت شبیه سازی جلو برده میشود.

واقعه بعدی از نوع IMEVT است و در لحظه FEL(IMEVT) روی می دهد.

```

C
C
C TIME ADVANCE ROUTINE - FINDS NEXT EVENT ON FUTURE EVENT
  LIST AND ADVANCES CLOCK
C
C
C SUBROUTINE TIMEADV
  REAL MILT, MST
  COMMON /GEN/ MILT, MST, SIGMA, NECT, LOT, LST, FIL,
  1 CHKOUT(100), S, P, S, F, ND
  COMMON /TIMEZ/ CLOCK, IMEVT, NUMEVS, FEL(2)
  - P, FIL=1, S=29
  IMEVT=0
C
C SEARCH THE FUTURE EVENT LIST FOR THE NEXT EVENT
C
C DO 30 I=1, NUMEVS
  IF (FEL(I).GE. P) GO TO 30
  FROM=FEL(I)
  IMEVT=I
30 CONTINUE
  IF (IMEVT.GT.0) GO TO 50
C
C ERROR CONDITION - FUTURE EVENT LIST EMPTY
C
  WRITE(6,40)
40 FORMAT(1X,*****FUTURE EVENT LIST EMPTY*****
  1X,***SIMULATION CANNOT CONTINUE***
  )
  CALL RPTGEN
  STOP
C
C ADVANCE SIMULATION CLOCK
C
C NEXT EVENT IS TYPE "IMEVT", WHICH WILL
  C OCCUR AT TIME FEL(IMEVT)
C
  50 CLOCK = FEL(IMEVT)
  RETURN
  END

```

شکل ۱۰-۱۶ زیر برنامه مربوط به جلو بردن زمان

شکل ۱۰-۱۷ فهرست زیر برنامه مربوط به پیش آمد ورود را ارائه میکند. منطق اصلی پیش آمد ورود صف قبلا ارائه شد.

پیش از هر چیز باید وضعیت سرویس دهنده از نظر مشغول یا آزاد بودن تعیین شود. در زیر برنامه پیش آمد ورود صفر یا یک بودن متغیری منعکس کننده به ترتیب آزاد یا مشغول بودن سرویس دهنده است. در صورتیکه سرویس دهنده بیکار (آزاد) باشد.

نیازی به تجدید اطلاعات مربوط به F.S, MQ, B و ND نیست. وضعیت سرویس دهنده به وضعیت مشغول (LST=1) تغییر داده شده و زمان ورود در CHKOUT(1) ثبت می شود. باید توجه داشت که CHKOUT مبین مجموعه ای است که حاوی صفات "زمان ورود مشتری است. متعاقباً در

بنابراین تا جایی که تعداد مشتریان حاضر در سیستم $LOT + IST$ از ۱۰۰ تجاوز نکند هیچ شکلی ایجاد نخواهد شد. به منظور حصول اطمینان از این امر در صورت تجاوز $LOT + IST = I$ از ۱۰۰ زیر برنامه پیش آمد ورود پیام مناسبی در این زمینه چاپ کرده، اختیار را به زیر برنامه گزارش نویسی واگذار کرده و شبیه‌سازی متوقف می‌شود. اگر چنین وضعیتی به وجود آید، لزوماً به معنی غلط بودن نحوه شبیه‌سازی مسئله مورد بررسی نیست در واقع ممکن است سیستم پیش از آنچه که تصور می‌شد مشغول انجام فعالیت است و باید اندازه بردار $CHKOUT$ را بزرگتر و اندازه آئرا مثلاً به ۲۰۰ رسانید. جمله IF نیز در عوض مقایسه I با ۱۰۰ باید I را با ۲۰۰ مقایسه کند.

در گام بعد، به تجدید اطلاعات مربوط به B و MO اقدام می‌شود. مجموع زمان اشتغال B به طریق زیر تجدید می‌شود.

زمان B قبلی + طول مدت اشتغال بعد از رویداد آخرین پیش آمد = B جدید

چنین متنی به زبان فرتن به صورت $B = B + (CLOCK - TLE)$ نوشته شده است. به یاد داریم که TLE لحظه رویداد پیش آمد قبلی است. نظریه اینکه سرویس دهنده به طور مسلم طی فاصله زمانی از TLE تا CLOCK مشغول ارائه سرویس بوده است. مجموع زمان اشتغال سرویس دهنده را باید به میزان $CLOCK - TLE$ افزایش داد. پس از تجدید B باید به تجدید TLE و MO پرداخت. بعد از طی این مراحل، مجدداً باید به تولید پیش آمد ورود و برنامه ریزی آن در فهرست پیش آمدهای آتی اقدام کرد و این کنترل را به برنامه اصلی می‌سپارد.

شکل ۱۰-۱۸ فهرست زیر برنامه DPART را که برای اجرای پیش آمد ترک سیستم نوشته شده است عرضه می‌کند.

زیر برنامه پیش آمد خروج از سیستم
اطلاعات موجود در $DPART$ و FND تجدید می‌شود چون LOT کاهش می‌یابد MO تغییر نمی‌کند.
وضعیت صف انتظار بررسی می‌شود.

چون در صف انتظار هیچ مشتری وجود ندارد وضعیت بررسی دهنده به آزاد تبدیل شده و پیش آمد بعدی خروج از سیستم در زمان "بی‌نیاحت" روی خواهد داد.
حداقل یک مشتری در صف انتظار وجود دارد پس کلیه مشتریان حاضر در صف یک خانه به جلو

```

C
C
C ARRIVAL EVENT ROUTINE
C
C
C SUBROUTINE ARRVL
  REAL NLT, NST, LIT
  COMMON /SIM/ NLT, NST, SIGMA, NCST, LOT, LST, TLE,
  1 CHKOUT(100), B, MO, S, F, ND
  COMMON /TIMER/ CLOCK, NRETT, NUMRETS, FEL(2)
C
C DETERMINE IF SERVER IS BUSY
C
C . IF(LST .EQ. 1) GO TO 20
C
C SERVER IS FREE. UPDATE SYSTEM STATE AND
C RECORD ARRIVAL TIME OF NEW CUSTOMER
C
C . LST = 1
C . CHKOUT(1) = CLOCK
C
C GENERATE A SERVICE TIME FOR THE NEW ARRIVAL
C AND SCHEDULE THE DEPARTURE FOR THIS ARRIVAL
C
C . SRT = NORME(NST, SIGMA)
C . FEL(2) = CLOCK + SRT
C
C UPDATE CUMULATIVE STATISTIC, MO (AND ALSO TLE)
C
C . TLE = CLOCK
C . IF(LIT .GT. MO) MO = LIT
C GO TO 100
C
C
C SERVER IS BUSY. UPDATE SYSTEM STATE AND
C RECORD ARRIVAL TIME OF NEW CUSTOMER
C
C 20 . LOT = LOT + 1
C . I = LOT + LST
C . IF(I .GT. 100) GO TO 200
C . CHKOUT(1) = CLOCK
C
C UPDATE CUMULATIVE STATISTICS, B AND MO.
C (S, MO, AND F ARE NOT UPDATED WHEN AN
C ARRIVAL OCCURS.)
C
C . B = B + (CLOCK - TLE)
C . TLE = CLOCK
C . IF(LIT .GT. MO) MO = LIT
C
C GENERATE AN INTERARRIVAL TIME AND
C SCHEDULE THE NEXT ARRIVAL EVENT
C
C 100 . IAT = EXPON(NLAT)
C . FEL(1) = CLOCK + IAT
C RETURN
C
C
C ERROR CONDITION HAS OCCURRED. ARRAY CHKOUT
C HAS OVERFLOWED. INCREASE DIMENSION.
C
C 200 . WRITE(6,205)
C 205 . FORMAT('11,***OVERFLOW IN ARRAY CHKOUT. INCREASE',
  1 /, '11,***SIMULATION CANNOT CONTINUE***')
C
C CALL RTGCR
C STOP
END

```

جریان فعل و انفعالات مربوط به پیش آمد ورود در چهارچوب شکل ۱۰-۱۰ تشریح گردید. نخست، اطلاعات گردآوری شده در $PC4$ و $AVGR$ براساس روابط ارائه شده در جدول ۱۰-۳ انجام می گیرد. متعاقباً پارامترهای ورودی $AMAT$ ، $SIGMA$ ، $MSVT$ همراه خلاصه اطلاعات چاپ می شوند. چاپ پارامترهای ورودی به مثاله، در پایان شبیه سازی جهت اطمینان از صحت آنها و تغییر ناخواسته

زیر برنامه مربوط به گزارش نویسی در شکل ۱۰-۱۹ ارائه شده است. اطلاعات خلاصه شده به وسیله $AVGR$ ، $PC4$ و $PC4$ براساس روابط ارائه شده در جدول ۱۰-۳ انجام می گیرد. متعاقباً پارامترهای ورودی $AMAT$ ، $SIGMA$ ، $MSVT$ همراه خلاصه اطلاعات چاپ می شوند. چاپ پارامترهای ورودی به مثاله، در پایان شبیه سازی جهت اطمینان از صحت آنها و تغییر ناخواسته

زیر برنامه مربوط به گزارش نویسی در شکل ۱۰-۱۹ ارائه شده است. اطلاعات خلاصه شده به وسیله $AVGR$ ، $PC4$ و $PC4$ براساس روابط ارائه شده در جدول ۱۰-۳ انجام می گیرد. متعاقباً پارامترهای ورودی $AMAT$ ، $SIGMA$ ، $MSVT$ همراه خلاصه اطلاعات چاپ می شوند. چاپ پارامترهای ورودی به مثاله، در پایان شبیه سازی جهت اطمینان از صحت آنها و تغییر ناخواسته

زیر برنامه مربوط به گزارش نویسی در شکل ۱۰-۱۹ ارائه شده است. اطلاعات خلاصه شده به وسیله $AVGR$ ، $PC4$ و $PC4$ براساس روابط ارائه شده در جدول ۱۰-۳ انجام می گیرد. متعاقباً پارامترهای ورودی $AMAT$ ، $SIGMA$ ، $MSVT$ همراه خلاصه اطلاعات چاپ می شوند. چاپ پارامترهای ورودی به مثاله، در پایان شبیه سازی جهت اطمینان از صحت آنها و تغییر ناخواسته

زیر برنامه مربوط به گزارش نویسی در شکل ۱۰-۱۹ ارائه شده است. اطلاعات خلاصه شده به وسیله $AVGR$ ، $PC4$ و $PC4$ براساس روابط ارائه شده در جدول ۱۰-۳ انجام می گیرد. متعاقباً پارامترهای ورودی $AMAT$ ، $SIGMA$ ، $MSVT$ همراه خلاصه اطلاعات چاپ می شوند. چاپ پارامترهای ورودی به مثاله، در پایان شبیه سازی جهت اطمینان از صحت آنها و تغییر ناخواسته

زیر برنامه مربوط به گزارش نویسی در شکل ۱۰-۱۹ ارائه شده است. اطلاعات خلاصه شده به وسیله $AVGR$ ، $PC4$ و $PC4$ براساس روابط ارائه شده در جدول ۱۰-۳ انجام می گیرد. متعاقباً پارامترهای ورودی $AMAT$ ، $SIGMA$ ، $MSVT$ همراه خلاصه اطلاعات چاپ می شوند. چاپ پارامترهای ورودی به مثاله، در پایان شبیه سازی جهت اطمینان از صحت آنها و تغییر ناخواسته

زیر برنامه مربوط به گزارش نویسی در شکل ۱۰-۱۹ ارائه شده است. اطلاعات خلاصه شده به وسیله $AVGR$ ، $PC4$ و $PC4$ براساس روابط ارائه شده در جدول ۱۰-۳ انجام می گیرد. متعاقباً پارامترهای ورودی $AMAT$ ، $SIGMA$ ، $MSVT$ همراه خلاصه اطلاعات چاپ می شوند. چاپ پارامترهای ورودی به مثاله، در پایان شبیه سازی جهت اطمینان از صحت آنها و تغییر ناخواسته

زیر برنامه مربوط به گزارش نویسی در شکل ۱۰-۱۹ ارائه شده است. اطلاعات خلاصه شده به وسیله $AVGR$ ، $PC4$ و $PC4$ براساس روابط ارائه شده در جدول ۱۰-۳ انجام می گیرد. متعاقباً پارامترهای ورودی $AMAT$ ، $SIGMA$ ، $MSVT$ همراه خلاصه اطلاعات چاپ می شوند. چاپ پارامترهای ورودی به مثاله، در پایان شبیه سازی جهت اطمینان از صحت آنها و تغییر ناخواسته

رانده می شوند.

وضعیت سیستم تجدید می شود.

یک مقدار تصادفی برای مدت ارايه سرویس به آن مشتری که سرویس آن در حال آغاز است، تولید شده و پیش آمد بعدی خروج از سیستم برنامه ریزی می شود.

```

C
C DEPARTURE EVENT ROUTINE
C
C
C SUBROUTINE DEPART
C REAL K1AT,MSVT
C COMMON /SIN/ K1AT,MSVT,SIGMA,NCUST,LOT,LST,TLE,
C 1 CHKOUT(100),B,PA,S,F,ND
C COMMON /TIMERS/ CLOCK,DEPT,SUBSVS,FEL(2)
C
C
C UPDATE CUMULATIVE STATISTICS, B, S, ND, AND P
C (LOT IS DECREASING, SO NO DOES NOT CHANGE NOW)
C
C B = B + (CLOCK - TLE)
C TLE = CLOCK
C RT = CLOCK - CHKOUT(1)
C S = S + RT
C ND = ND + 1
C IF (RT .GE. 4.0) P = P + 1
C
C CHECK CONDITION OF WAITING LINE
C IF (LOT .GE. 1) GO TO 20
C
C NO CUSTOMERS IN LINE, SO SERVER BECOMES IDLE
C AND NEXT DEPARTURE TIME IS SET EQUAL TO "INFINITY"
C
C LST = 0
C FEL(2) = 1.0E+30
C RETURN
C
C AT LEAST ONE CUSTOMER IS IN LINE, SO MOVE EACH
C CUSTOMER IN LINE FORWARD ONE SPACE
C
C DO 30 I = 1,LOT
C I1 = I + 1
C CHKOUT(I1) = CHKOUT(I)
C 30 CONTINUE
C
C UPDATE SYSTEM STATE
C
C LOT = LOT - 1
C
C GENERATE NEW SERVICE TIME FOR CUSTOMER BEGINNING
C SERVICE, AND SCHEDULE THE NEXT DEPARTURE EVENT
C
C SVT = NORME(MSVT,SIGMA)
C FEL(2) = CLOCK + SVT
C RETURN
C END

```

شکل ۱۰-۱۸ زیر برنامه پیش آمد خروج از سیستم برنامه FORTRAN شبیه سازی صف و یک سرویس دهنده

با تغییر ضابطه اتمام شبیه‌سازی، ایجاد تغییراتی جزئی در برنامه اصلی و زیربرنامه راه‌اندازی مدل ضروری خواهد بود. به طور خاص، NUMEVS را باید مساوی با ۳ قرار داد و به تغییر دستور GO TO در برنامه اصلی اقدام کرد. ایجاد تغییراتی از این قبیل به عهده خواننده است.

۱۰.۷.۲ - شبیه‌سازی با استفاده از GASP

زبان GASP IV مجموعه‌ای از زیر برنامه‌ها به زبان FORTRAN است که آن را برای شبیه‌سازی زمان‌بندی - پیش‌آمدها با زبان FORTRAN نوشته‌اند. این زبان شامل ۳۰ زیر برنامه و تابع است که تسهیلات مورد نیاز شامل رویتین پیشبرد - زمان (بنام GASP)، رویتینهای مدیریت فهرست پیش‌آمدهای آتی (مانند افزودن پیش‌آمدهای جدید به فهرست پیش‌آمدهای آتی)، رویتینهای افزایش و کاهش موجودیتها از مجموعه‌ها، رویتینهای جمع‌آوری آمار، رویتینهای مولد تصادفی و یک مولد گزارش استاندارد است.

برنامه‌نویس باید یک برنامه اصلی، یک رویتین مقداردهی اولیه، رویتینهای پیش‌آمدها و در صورت نیاز یک مولد گزارش علاوه بر یک زیربرنامه بنام EVNTS فراهم کند. برنامه اصلی باید یک دستور CALL "GASP" برای شروع شبیه‌سازی داشته باشد. زیر برنامه GASP عمل تعیین پیش‌آمد بعدی را انجام می‌دهد و زیر برنامه نوشته شده کاربر EVNTS شاخص (مانند IMEVT در مدل FORTRAN بخش ۱۰.۷ نام NEXT IV در GASP) را صدا می‌زند.

این شاخص (NEXT) پیش‌آمدی که به عنوان پیش‌آمد بعدی به وسیله EVENTS صدا زده می‌شود را تعیین می‌کند. برای GASP IV بیشتر کامپیوترهایی که کامپایلر FORTRAN دارند، موجود است.

۱۰.۷.۳ - شبیه‌سازی با استفاده از SIMSCRIPT

زبان SIMSCRIPT II.5 یک زبان برنامه نویسی سطح بالا با تسهیلات مشخص طراحی شده جهت توسعه مدل شبیه‌سازی پیش‌آمدهای - گسسته است. به عنوان یک زبان شبیه‌سازی، اجازه زمان‌بندی پیش‌آمدها یا محاوره بروسه‌ها را می‌دهد. به عنوان یک زبان علمی، حداقل به قدرت زبانهای FORTRAN، ALGOL، PL/I، Pascal است.

```

C NORMAL RANDOM VARIATE GENERATOR
C
C FUNCTION NORMAL(MEAN, SIGMA)
C   REAL MEAN, SIGMA
C   DATA K/0, PI/3.14159/
C
C CHECK TO SEE WHICH N(0,1) RANDOM VARIABLE TO USE
C
C IF (1.EQ.1) GO TO 10
C
C GENERATE TWO UNIFORM(0,1) RANDOM NUMBERS
C
C   RONE=RAUN(DUMPT)
C   RTWO=RAUN(DUMPT)
C
C GENERATE TWO NORMAL(0,1) RANDOM VARIABLES
C
C   (SEE EQUATION (8-25).)
C
C   ZONE=SQRT(-2*ALOG(RONE)) * COS(2*PI*RTWO)
C   ZTWO=SQRT(-2*ALOG(RONE)) * SIN(2*PI*RTWO)
C
C COMPUTE NORMAL RANDOM VARIABLE WITH PARAMETERS
C (MEAN, SIGMA) FOR MEAN AND STANDARD DEVIATION
C
C   NORML = ZONE*SIGMA + MEAN
C
C   K = 1
C   RETURN
C
C COMPUTE NORMAL RANDOM VARIABLE, N(MEAN, SIGMA**2)
C
C   10 NORML = ZTWO*SIGMA + MEAN
C
C   K = 0
C
C   RETURN
C
C   END

```

شکل ۱۰-۲۱ مولد اعداد با توزیع چگالی نرمال صف یک سرویس‌دهنده

SINGLE SERVER QUEUE SIMULATION - GROCERY STORE CHECKOUT COUNTER

```

MEAN INTERARRIVAL TIME      4.50
MEAN SERVICE TIME           3.20
STANDARD DEVIATION OF SERVICE TIMES .50
NUMBER OF CUSTOMERS SERVED  1000

```

```

SERVER UTILIZATION          .60
MAXIMUM LINE LENGTH        5
AVERAGE RESPONSE TIME     4.59 MINUTES
PROPORTION WHO SPEND FOUR MINUTES OR MORE IN SYSTEM .48
SIMULATION RUNLENGTH      4460.68 MINUTES
NUMBER OF DEPARTURES      1000

```

شکل ۱۰-۲۲ نتایج بدست آمده از شبیه‌سازی مسئله صف و یک سرویس‌دهنده

می توانند از هر روتین پیش آمد یا برنامه اصلی صدا زده شوند.

مثال ۷-۱۰ شبیه سازی صف و یک سرورس دهنده با SIMSCRIPT

یک مدل SIMSCRIPT از یک فروشگاه شرح داده می شود. در راه حل FORTRAN زمان اولین ورود یک زمان تصادفی انتخاب شده از توزیع نمایی است و در مدل SIMSCRIPT فرض می شود که اولین ورود در زمان صفر روی می دهد.

نظامنامه (Preamble) در شکل ۲۳-۱۰ فهرست شده و دستورات این نظامنامه شرح و بیان می شود. مد متغیرها ممکن است صحیح یا اعشاری باشند، دستور "NORMALY, MODE IS" برای تمامی متغیرهای تعریف نشده یک زمینه را فراهم می آورد. گرچه نامها در SIMSCRIPT هر طریقی می تواند داشته باشد اما ۵ یا ۶ حرف اول آنها باید منحصر بفرد باشد، کلمه DEFINE عبارت MEAN را معنی می کند و برای هر کلمه در برنامه عبارت معادل آن را قرار می دهد. خط ۳ آن نام "DEPARTING CUSTOMER" را به عنوان "ATTRB1" (صفت ۱) کمیابل می کند. سپس دو پیش آمد ARRIVAL و DEPARTURE تعریف شده اند (هر پیش آمد مانند یک رکورد است) پیش آمدها در فهرست پیش آمد آتی بر می شوند:

```

PREAMBLE
NORMALY, MODE IS INTEGER
DEFINE DEPARTING,CUSTOMER TO MEAN ATTRB1
DEFINE ARRIVAL,TIME TO MEAN ATTRB2

EVENT NOTICES INCLUDE ARRIVAL
EVENT DEPARTURE HAS A DEPARTING,CUSTOMER
TEMPORARY ENTITIES
EVERY CUSTOMER HAS AN ARRIVAL,TIME
AND MAY BELONG TO THE QUEUE
THE SYSTEM OWNS THE QUEUE
DEFINE QUEUE AS FIFO SET
DEFINE REPORT,GENERATOR AS A ROUTINE

DEFINE MEAN, MEAN, SIGMA, ARRIVAL,TIME, AND RESPONSE,TIME
AS REAL VARIABLES
DEFINE SERVER, NCUST, IS, RT, 4 AND NUMBER,OF, DEPARTURES
AS INTEGER VARIABLES
DEFINE TIME TO MEAN 0
DEFINE BUST TO MEAN 1

ACCOMMODATE AND AS THE AVERAGE OF SERVER
TALLY MAX,LENGTH AS THE MAXIMUM OF N,QUEUE
TALLY AVG,RT AS THE AVERAGE OF RESPONSE,TIME
TALLY PROB,RT,GE,4 AS THE AVERAGE OF IS,RT,4
END
  
```

شکل ۲۳-۱۰ نظامنامه SIMSCRIPT برای مدل صف و یک سرورس دهنده

بسیاری از برنامه ها بطور کارآتر از زبان FORTRAN در SIMSCRIPT انجام می گیرند. یک برنامه SIMSCRIPT می تواند به صورت دستوراتی مانند زبان انگلیسی رها از فرمت نوشته شود. برخلاف FORTRAN SIMSCRIPT بطور خودکار نگهداری فهرست پیش آمدهای آتی و الگوریتم زمان بندی - پیش آمدها/پیشبرد- زمان را انجام می دهد، مجموعه ها را نگهداری می کند، عملیات افزایش و کاهش موجودیتها از مجموعه ها را برعهده دارد، جمع آوری آمارهای مورد تقاضا را تقلیل می کند و مولدهای اعداد تصادفی برای محدوده وسیعی از توزیعها را فراهم می آورد.

دنیای انجام عملیات توسط SIMSCRIPT براساس موجودیتها، صفات و مجموعه ها است. موجودیتها به صورت دائمی و موقت دسته بندی می گردند. موجودیتهای دائمی اشیایی از سیستم را نشان می دهد که برای دوره شبیه سازی در سیستم باقی می مانند. به عنوان مثال تعداد مشخص سرورس دهنده ها در یک مدل صف، کنتی ها در مدل کشتریانی می باشد. اشیاء موقتی ارائه کننده اشیایی مانند مشتریان در مدل صف است که به سیستم وارد می شوند، برای مدتی می مانند و سپس سیستم را ترک می کند. تعداد موجودیتهای موقتی در طی شبیه سازی متغیر است، موجودیتها دارای صفات می باشند و موجودیت مشخص شده به یک مجموعه تعلق دارد. یک برنامه SIMSCRIPT شامل یک نظامنامه، یک برنامه اصلی، روتینهای پیش آمد و زیر برنامه های مورد نظر است. به خاطر دانسته باشید که روتین پیشبرد - زمان، روتینهای مولد اعداد تصادفی و روتینهای جمع آوری آمار بطور خودکار وجود دارند. نظامنامه، شرح ثابت سیستم به وسیله تعریف تمامی موجودیتها، صفاتشان و مجموعه هایی که به آنها تعلق دارند، را می دهد. همچنین متغیرهای شناور (Global) برای تعیین وضعیت سیستم و سیستمهای آماری جمع آوری کننده از متغیرها را تعیین می کند. یک مجموعه بزرگی از متغیرها را خودکار نگه می دارد. به عنوان مثال TIME V ساعت شبیه سازی را ارائه می کند. اگر QUEUE نام یک صف باشد، N QUEUE تعداد موجودیتهای داخل این مجموعه را نشان می دهد. برنامه اصلی مقادیر ورودی را می خواند، وضعیت سیستم را مقداردهی می کند و اولین پیش آمد را تولید می نماید. روتینهای پیش آمدها بطور خودکار به وسیله زیر برنامه پیشبرد - زمان صدا زده می شود که به وسیله دستور "START SIMULATION" در برنامه اصلی فعال می گردد. به این ترتیب زیر برنامه ها

```

MAIN
  SCHEDULE AN ARRIVAL NOW
  LET MIAT = 4.5 "MINUTES, THE MEAN INTERARRIVAL TIME
  LET MSFT = 3.2 "MINUTES, THE MEAN SERVICE TIME
  LET SIGMA = 0.6 "MINUTE, THE STANDARD DEVIATION OF SERVICE TIME
  LET NCUST = 1000 "CUSTOMERS TO BE SERVED (THE STOPPING CRITERIA)

  LET SERVER = IDLE "SO THAT THE FIRST ARRIVAL WILL FIND THE SERVER IDLE
  LET NUMBER OF DEPARTURES = 0

  START SIMULATION

```

END

شکل ۱۰-۲۴ برنامه اصلی SIMSCRIPT برای مدل صف و یک سرویس دهنده

دستور START SIMULATION اجرای الگوریتم پیشبرد - زمان / زمان بندی - پیش آمد را شروع می کند. اگر بخواهد با زمان تصادفی رخ دهد دستور SCHEDULE با دستور زیر جایگزین می شود:

SCHEDULE AN ARRIVAL IN EXPONENTIAL.F(MIAT) MINUTES

مولد تصادفی نمایی با میانگین MIAT و آرگومان دوم عدد صحیح از ۱ تا ۱۰ در این زمان به صورت پیش ساخته وجود دارد. روتین پیش آمد ARRIVAL در شکل ۱۰-۲۵ نشان داده شده است. دستور "CREATE" یک کپی از موجودیهای موقتی ایجاد می کند. صفت "ARRIVAL.TIME" موجودیت ایجاد شده CUSTOMER با زمان جاری شبیه سازی مقداردهی می شود. SIMSCRIPT دارای ساختار IF به شکل زیر است:

```

IF Condition is true
  then do these Statements
  OTHER WISE (or ELSE)
  then do these Statements
  ALWAYS (or REGARDLESS)

```

که قابلیت خواندن یک مدل را بهبود می بخشد. اگر وضعیت SERVER IDLE باشد (مقدار صفر دارد)، آنرا BUSY می کند و یک خروجی از ورودی CUSTOMER برای رخ دادن درانتهای زمان سرویس، زمان بندی می شود. این زمان بندی یعنی پیش آمد DEPARTURE در فهرست پیش آمدها آتی قرار داده می شود و شامل اطلاعات خارجی (اشاره گر یا شاخص نام دارد) است که به CUSTOMER

علاوه بر این، پیش آمد ممکن است دارای صفات باشد. پیش آمد "ARRIVAL" صفاتی ندارد اما پیش آمد "DEPARTURE" دارای صفت "DEPARTING.CUSTOMER" برای هر دسته است. یک نوع از موجودیت موقت یک "CUSTOMER" با یک صفت "ARRIVAL.TIME" می باشد. یک نوع از موجودیت "CUSTOMER" ممکن است به "QUEUE" تعلق داشته باشد که در سیستم به صورت FIFO عمل می کند. زیر برنامه "REPORT GENERATOR" می تواند به وسیله برنامه نویس صدا زده شود. سپس متغیرهای شناور تعریف شده است.

منظور از دو دستور "DEFINE...TO MEAN" آنست که تمامی وقوع (BUSY) با اعداد (۱) جایگزین شوند. دستورات "TALLY" و "ACCUMULATE" تعیین کننده آمارهای مطمئن است که باید جمع آوری گردد. "SERVER" یک متغیر وضعیت سیستم با مقادیر (BUSY) یا (IDLE) مؤید وضعیت سرویس دهنده است. با دستور "ACCUMULATE" بطور خودکار SIMSCRIPT مجموع وزن زمان متغیر "SERVER" (در مدل FORTRAN نام "B" دارد) را نگهداری می کند. در انتهای شبیه سازی این جمع بر TIME.V برای تولید "RHO" تقسیم می شود. میانگین "RHO" بطور خودکار به وسیله دستور فوق ارائه می شود.

RHO برابر زمان اشتغال سرویس دهنده است. اولین "TALLY"، MAX.Q.LENGTH به عنوان ماکزیمم تعداد موجودیهای صف "QUEUE" می باشد و از آیترو "N.Set-name" بطور خودکار برای هر مجموعه به وسیله SIMSCRIPT نگهداری می شود. وقتی یک پیش آمد DEPARTURE رخ می دهد برنامه نویس "RESPONSE.TIME" از "DEPARTING.CUSTOMER" را محاسبه می کند و SIMSCRIPT بطور خودکار این مقدار را به مقدار آماری در انتهای شبیه سازی به عنوان میانگین زمان پاسخ "AVG.RT"، اضافه می کند. وقتی که یک پاسخ زمانی ۴ دقیقه یا بیشتر طول بکشد، متغیر "IS.RT.4" مقدار ۱ می گیرد و در غیر اینصورت ۰ می شود بنابراین PROBRT.GE.4 اگر پاسخ زمانی بیشتر از ۴ باشد عمل می کند.

برنامه اصلی در شکل ۱۰-۲۴ نشان داده شده است. از دستور SCHEDULE برای قرار دادن پیش آمدها در فهرست پیش آمدهای آتی استفاده می شود. اولین پیش آمد ARRIVAL به زمان صفر زمان بندی SCHEDULE می شود. پارامترهای ورودی MIAT، MSVT، SIGMA و NCUST (با همان معانی در مدل فورتان) مقداردهی می شوند.

صنعتی تمامی آنها می توانند از یک فایل ورودی با فرمت یا بدون فرمت خوانده شوند. متغیر وضعیت سیستم "SERVER" با "IDLE" (۰) مقدار می گیرد ولی بلافاصله به وسیله روتین پیش آمد ARRIVAL به "BUSY" مقداردهی می شود.

به دقیقه در (در دستور IF) با ضرب در ۶۰ (HOURS.V) و در ۶۰ (MINUTES.V) تبدیل می شود. به دقیقه در (در دستور IF) با ضرب در ۶۰ (HOURS.V) و در ۶۰ (MINUTES.V) تبدیل می شود. به دقیقه در (در دستور IF) با ضرب در ۶۰ (HOURS.V) و در ۶۰ (MINUTES.V) تبدیل می شود. به دقیقه در (در دستور IF) با ضرب در ۶۰ (HOURS.V) و در ۶۰ (MINUTES.V) تبدیل می شود.

به دقیقه در (در دستور IF) با ضرب در ۶۰ (HOURS.V) و در ۶۰ (MINUTES.V) تبدیل می شود. به دقیقه در (در دستور IF) با ضرب در ۶۰ (HOURS.V) و در ۶۰ (MINUTES.V) تبدیل می شود. به دقیقه در (در دستور IF) با ضرب در ۶۰ (HOURS.V) و در ۶۰ (MINUTES.V) تبدیل می شود.

```

EVENT DEPARTURE GIVEN DEPARTING-CUSTOMER
LET RESPONSE.TIME = TIME.Y - ARRIVAL.TIME(DEPARTING-CUSTOMER)
DESTROY A CUSTOMER CALLED DEPARTING-CUSTOMER

IF RESPONSE.TIME < 0 THEN
  LET IS.SERVER = 1
ELSE
  LET IS.SERVER = 0
REGARDLESS

ADD 1 TO NUMBER OF DEPARTURES
IF NUMBER OF DEPARTURES IS GE NCUST,
  CALL REPORT-GENERATOR
ALWAYS

IF QUEUE IS EMPTY,
  LET SERVER = IDLE
OTHERWISE
  REMOVE FIRST CUSTOMER FROM QUEUE
  SCHEDULE A DEPARTURE GIVEN CUSTOMER IN
  NORMAL.F(MSYT, SIGMA, 1) MINUTES
REGARDLESS

RETURN
END

```

شکل ۲۶-۱۰ روتین پیش آمد خروج SIMSCRIPT برای مدل صف و یک سرور بس وند

شکل ۲۷-۱۰ فهرست روتین REPORT GENERATOR را فراهم می آورد (روتین یک پیش آمد نیست) و از طریق روتین پیش آمد DEPARTURE صدا زده می شود. شکل عمومی دستور PRINT عبارتست از

PRINT n LINES WITH variable-name THUS

n همین تعداد خطوطی است که برنامه نویس می خواهد در خروجی ببیند. توجه اینکه فرمت تصویر برای متغیرها مقدار M4AT جایی که * * * وجود دارد، قرار می گیرد و مانند آن دستور STOP انتهای

خاصی که باید خارج شود اشاره می کند. هر موجودیت موقت ایجاد CREATE یک اشاره گر برای تشخیص آن از دیگر مشتری دارد. مقدار این اشاره گر باید ذخیره شود مگر آنها کم شوند.

اشاره گر به هر موجودیت CUSTOMER فقط در شروع سروریش در پیش آمد DEPARTURE ذخیره می گردد. اشاره گر به هر موجودیت CUSTOMER با نام آن مشتری منتظر سروریش، به وسیله دستور "FILE ARRIVAL IN QUEUE" در سری نام "QUEUE" ذخیره می شود جایی که پیش آمد ARRIVAL با SERVER در وضعیت BUSY رخ می دهد. وقتی یک ورود رخ می دهد، ورودی بعدی به وسیله تولید یک زمان ورودی زمان بندی SCHEDULE می شود. توجه کنید که مولد تصادفی نرمال با نام NORMAL.F سه آرگومان نیاز دارد: میانگین (MUST)، انحراف معیار (SIGMA) و جزو اعداد تصادفی مورد نیاز (یک عدد صحیح از ۱ تا ۱۰).

```

EVENT ARRIVAL
CREATE A CUSTOMER
LET ARRIVAL.TIME = TIME.Y

IF SERVER IS EQUAL TO IDLE,
  LET SERVER = BUSY
  SCHEDULE A DEPARTURE GIVEN CUSTOMER IN
  NORMAL.F(MSYT, SIGMA, 1) MINUTES
ELSE
  FILE CUSTOMER IN QUEUE
REGARDLESS
SCHEDULE AN ARRIVAL IN EXPONENTIAL.F(MSYT, 1) MINUTES
RETURN
END

```

شکل ۲۵-۱۰ روتین پیش آمد ورود SIMSCRIPT برای مدل صف یک سرور بس وند

روتین پیش آمد DEPARTURE در شکل ۲۶-۱۰ نشان داده شده است. وقتی این پیش آمد رخ می دهد صفت DEPARTING-CUSTOMER شامل اشاره گر ذخیره شده قبلی در پیش آمد می باشد. این اشاره گر تعیین کننده موجودیت CUSTOMER با این پیش آمد DEPARTURE است و جهت برداشت ورودی ARRIVAL استفاده می شود. TIME زمان DEPARTING-CUSTOMER که از طریق مشتری REPONSE.TIME می تواند محاسبه گردد. موجودیت CUSTOMER بعدی به نام DEPARTING-CUSTOMER رها می شود DESTROYED یعنی حافظه تخصیصی به این موجودیت به SIMSCRIPT واگذار می شود (تسامی موجودیهای موقت پس از ترک سیستم باید جهت تداوم شییه سازی مهیا شوند تا حافظه اشغال نشود).

سیس متغیر IS.RT.4 براساس آنکه RESPONSE.TIME بیشتر از ۴ دقیقه باشد یا نه به ۱ یا ۰ مقدار می گیرد. پیش فرض واحد زمان در SIMSCRIPT بر حسب روز است و RESPONSE.TIME بر حسب روز

تشیبہ سازی

زبان چرانات مختلف، از اعداد تصادفی، را استفاده می‌کنند (مرتبط با خطای تصادفی است).
شکل ۲۸-۱۰ مدل FORTRAN در شکل ۲۲-۱۰ توجه کنید. این اختلاف بدین خاطرات که هر دو
شکل ۲۸-۱۰ خروجی شبیه‌سازی را نشان می‌دهد. به اختلاف خروجی‌های مدل SIMSCRIPT در
روتین باعث اتمام شبیه‌سازی است.

```

ROUTINE REPORT GENERATOR
PRINT 1 LINE THUS
  SINGLE SERVER QUEUE SIMULATION - GROCERY STORE CHECKOUT COUNTER
SKIP 2 OUTPUT LINES
PRINT 4 LINES WITH
  MEAN, MSVT, SIGMA, NCST THUS
  MEAN INTERARRIVAL TIME
  MEAN SERVICE TIME
  STANDARD DEVIATION OF SERVICE TIMES
  NUMBER OF CUSTOMERS TO BE SERVED
SKIP 3 OUTPUT LINES
PRINT 1 LINES WITH
  RHO, MAX.Q.LENGTH, AVG.RT*HOURS, V*MINUTES, V,
  PROB ABT GE 4, TIME*HOURS, V*MINUTES, V, AND
  NUMBER OF DEPARTURES THUS
  SERVER UTILIZATION
  MAXIMUM LINE LENGTH
  AVERAGE RESPONSE TIME
  PROPORTION WHO SPEND FOUR
  MINUTES OR MORE IN SYSTEM
  SIMULATION RUNLENGTH
  NUMBER OF DEPARTURES

```

شکل ۱۰-۲۷ مولد گزارش SIMSCRIPT صف و یک سرویس دهنده

| | |
|-------------------------------------|------|
| MEAN INTERARRIVAL TIME | 4.50 |
| MEAN SERVICE TIME | 3.20 |
| STANDARD DEVIATION OF SERVICE TIMES | .60 |
| NUMBER OF CUSTOMERS TO BE SERVED | 1000 |

| | |
|--|--------------|
| SERVER UTILIZATION | .57 |
| MAXIMUM LINE LENGTH | 8 |
| AVERAGE RESPONSE TIME | 6.51 MINUTES |
| PROPORTION WHO SPEND FOUR
MINUTES OR MORE IN SYSTEM | .53 |
| SIMULATION RUNLENGTH | 4794.73 |
| NUMBER OF DEPARTURES | 1000 |

SCHEDULE A STOP.SIMULATION IN 80 HOURS

بررسی یک پیش آمد DEPARTURE باید خارج کند. NUMBER.OF.DEPARTURES

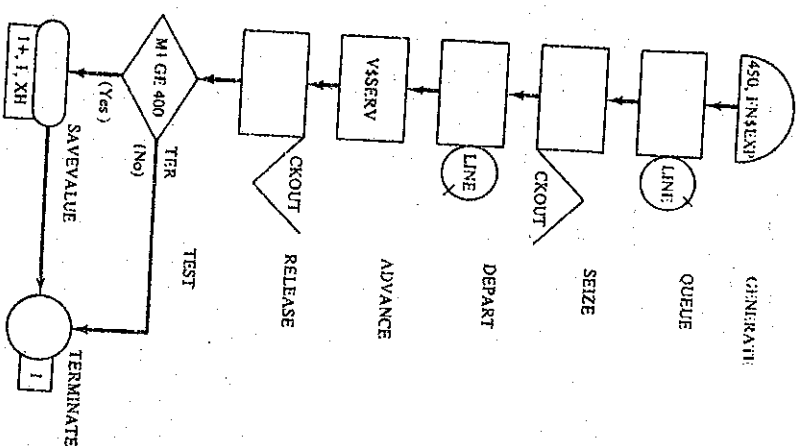
| SEIZE | CKOUT | نوع سروس |
|---------|-------------------------------|-------------|
| : | (بلاک‌های مدل کردن زمان سروس) | فعالیت سروس |
| RELEASE | RELEASE CKOUT | پایان سروس |

در GPSS تراکنش‌های بسیاری هم زمان می‌تواند در بلاک دیاگرام موجود باشد. هر تراکنش فعال همیشه در یک بلاک معین قرار دارد. عمل یک بلاک وقتی که یک تراکنش وارد یک بلاک گردد صورت می‌گیرد و بعضی بلاک‌ها (مانند QUEUE, DEPART, RELEASE, ADVANCE) تراکنش را قبول می‌کنند و اجازه می‌دهند که در صورت امکان به بلاک بعدی رد شود و بلاک‌های دیگر اغلب ورود یک تراکنش را قبول نمی‌کنند. بلاک SEIZE جایی که تسهیل به وسیله تراکنش دیگری اشغال شده یا مشغول باشد، ورود به یک تراکنش را قبول نمی‌کند. در چنین حالتی، تراکنش‌ها قصد SEIZE را دارند. تسهیل در بلاک بلافاصله قبل از بلاک SEIZE باقی می‌ماند. تراکنش‌ها به صورت FIFO وارد می‌شوند. فعالیت‌ها به وسیله بلاک‌های ADVANCE ارائه می‌گردند. در شکل ۱۰-۲۹ بلاک ADVANCE دارای زمان‌های سروس با توزیع نرمال می‌باشد. زمان یک سروس به وسیله فرمول دستور FVARIABLE محاسبه می‌شود که ارجاعی به FUNCTION NORM است. تابع FUNCTION مولد اعداد تصادفی نرمال استاندارد با میانگین صفر و انحراف معیار یک می‌باشد. دستور FVARIABLE عدد تصادفی نرمال استاندارد را به تصادفی با میانگین ۰ و انحراف معیار ۱ واحد زمانی تبدیل می‌کند.

آمارها در خط انتظار به وسیله بلاک‌های DEPART ... QUEUE جمع‌آوری می‌شود که با هم یک موجودیت صف را در GPSS تعریف می‌کنند. توجه کنید که موجودیت صف در GPSS نمی‌تواند یک خط انتظار تشکیل دهد. بلاک SEIZE تشکیل خط انتظار می‌دهد. بلاک‌های DEPART ... QUEUE آمارهای گوناگون روی این خط انتظار را راه‌انداز می‌گیرند. با ورود به بلاک QUEUE در شکل ۱۰-۲۹ یک تراکنش گفته می‌شود عضوی از صف است که "LINE" نام دارد. میانگین زمان صرف شده در صف به میانگین زمان تراکنش‌ها در صف اضافه می‌شود و بطور خودکار آمارها جمع‌آوری و در انتهای شبیه‌سازی چاپ می‌گردد.

در شکل ۱۰-۳۰ دستور SIMULATE یک کارت کنترلی است که GPSS را برای اجرای شبیه‌سازی هدایت می‌کند. دو تابع تعریف شده بعدی GPSS برای تولید اعداد تصادفی نرمال استاندارد است. دستور FVARIABLE زمان سروس مورد نیاز با توزیع نرمال از یک تابع تصادفی نرمال استاندارد را

یک سروس دهنده تنها به وسیله یک تسهیل با نام CKOUT ارائه می‌گردد یک تسهیل به وسیله یک جفت بلاک SEIZE و RELEASE به صورت زیر مدل می‌شود:



شکل ۱۰-۲۹ بلاک دیاگرام GPSS برای شبیه‌سازی صف و یک سروس دهنده

توجه آنکه تمامی آمارهای مورد نیاز (مانند جمع آوری شده در مدل FORTRAN) در خروجی استاندارد وجود دارد. GPSS بطور خودکار صفات سیستمها و تراکشنها را نگهداری می کند مانند M1 که معرف زمان حرکت تراکشن از تولید در مدل است. بلاکهای زیر برای مدلسازی تعداد تراکشنهایی که زمان پاسخ (M1) ۴ دقیقه یا بیشتر دارند را می شمارد.

بررسی بزرگتر یا مساوی بودن M1 با ۴۰۰

TESTGE M1,400,TER

اگر باشد XH1 یکی اضافه شود

SAVEVALVE 1+1,XH

شمارش نهایی در "متغیر ذخیره" XH1 ذخیره می شود.

خروجی استاندارد برای مدل GPSS در شکل ۱۰-۳۱ نشان داده شده است.

۱۰.۷.۵ شبیه سازی با استفاده SLAM

یک شبکه SLAM شامل گره ها و انشعابها می باشد. یک انشعاب ارائه کننده گذر زمان است که معرف یک فعالیت (Activity) می باشد و حتی معرف یک تعدادی سروس دهنده است. یک انشعاب به صورت دستور ACTIVITY گذاشته می شود. گره ها عبارتند از:

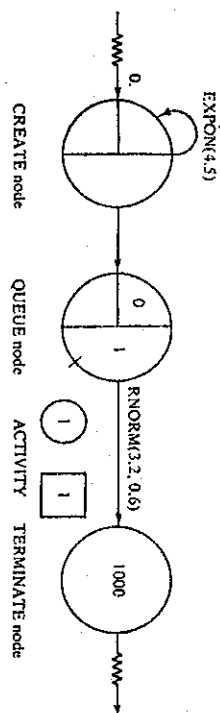
- پیش آمد ورود، گره CREATE
- تأخیر یا وضعیت انتظار، گره QUEUE
- پیش آمد خروج، گره TERMINATE
- و غیره.

مثال ۱۰-۹ (شبیه سازی صف و یک سروس دهنده با SLAM)

در شکل ۱۰-۳۲ مدل یک شبکه SLAM برای فروشگاه آمده است. توجه این که هر گره یک سمبل تعریف شده به علاوه اپراند هایش دارد. تبدیل آن شبکه به دستورات SLAM در شکل ۱۰-۳۳ نشان داده شده است.

فراهم می کند. سپس دستورات بلاکی مربوط به بلاک دیاگرام متعاقب آن می آیند. دستورالعمل آخر START 1000، شروع و پایان شبیه سازی است. یک شمارنده با یک مقدار اولیه 1000 به وسیله "A" هر زمان که یک تراکشن به بلاک TERMINATE وارد می شود کاهش می یابد، "A" یک اپراند بلاکی TERMINATE (در شکل ۱۰-۳۰) است. وقتی که شمارنده صفر شود شبیه سازی متوقف و گزارش خروجی (شکل ۱۰-۳۱) تولید می گردد.

| BLOCK NUMBER | *LOC | OPERATION | A,B,C,D,E,F,G,H,I,J | COMMENTS | CARD NUMBER |
|--------------|------|-------------------------------------|---------------------|----------|-------------|
| 1 | * | SIMULATE | | | 1 |
| 2 | * | SIMULATION OF A SINGLE SERVER QUEUE | | | 2 |
| 3 | * | EXP FUNCTION RNI,C24 | | | 3 |
| 4 | * | EXP FUNCTION RNI,C24 | | | 4 |
| 5 | * | EXP FUNCTION RNI,C24 | | | 5 |
| 6 | * | EXP FUNCTION RNI,C24 | | | 6 |
| 7 | * | EXP FUNCTION RNI,C24 | | | 7 |
| 8 | * | EXP FUNCTION RNI,C24 | | | 8 |
| 9 | * | EXP FUNCTION RNI,C24 | | | 9 |
| 10 | * | EXP FUNCTION RNI,C24 | | | 10 |
| 11 | * | EXP FUNCTION RNI,C24 | | | 11 |
| 12 | * | EXP FUNCTION RNI,C24 | | | 12 |
| 13 | * | EXP FUNCTION RNI,C24 | | | 13 |
| 14 | * | EXP FUNCTION RNI,C24 | | | 14 |
| 15 | * | EXP FUNCTION RNI,C24 | | | 15 |
| 16 | * | EXP FUNCTION RNI,C24 | | | 16 |
| 17 | * | EXP FUNCTION RNI,C24 | | | 17 |
| 18 | * | EXP FUNCTION RNI,C24 | | | 18 |
| 19 | * | EXP FUNCTION RNI,C24 | | | 19 |
| 20 | * | EXP FUNCTION RNI,C24 | | | 20 |
| 21 | * | EXP FUNCTION RNI,C24 | | | 21 |
| 22 | * | EXP FUNCTION RNI,C24 | | | 22 |
| 23 | * | EXP FUNCTION RNI,C24 | | | 23 |
| 24 | * | EXP FUNCTION RNI,C24 | | | 24 |
| 25 | * | EXP FUNCTION RNI,C24 | | | 25 |
| 26 | * | EXP FUNCTION RNI,C24 | | | 26 |
| 27 | * | EXP FUNCTION RNI,C24 | | | 27 |
| 28 | * | EXP FUNCTION RNI,C24 | | | 28 |
| 29 | * | EXP FUNCTION RNI,C24 | | | 29 |
| 30 | * | EXP FUNCTION RNI,C24 | | | 30 |
| 31 | * | EXP FUNCTION RNI,C24 | | | 31 |

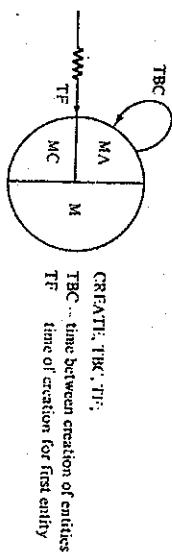


شکل ۱۰-۲۲ شبکه SLAM برای شبیهسازی صف و یک سرور دهنده

GEN RANKS AND CLASS, SINGLE SERVER QUEUE EXAMPLE, 5/31/1982, 1;
LIMITS, 1, 0, 30; MODEL CAN USE 1 FILE, MAX NO. OF SIMULTANEOUS ENTRIES 30
NETWORK;
CREATE, EXPON(4,5);
QUEUE(1);
ACTIVITY(1);
TERMINATE, 1000;
ENDNETWORK;
FIN;
END OF SIMULATION

شکل ۱۰-۲۳ مدل SLAM برای صف و یک سرور دهنده

شکل ۱۰-۲۲ نشان می‌دهد که یک صف و یک سرور دهنده با سه گره و یک انتخاب غیر صفر مدل می‌شود. گره CREATE پیش آمد ورود را ارائه می‌کند. یک موجودیت ایجاد می‌شود CREATE و در طول شبکه جریان می‌یابد. سبیل گره CREATE، دستور معادل و اپراند هایش به صورت زیر است:



| RELATIVE CLOCK | 462628 | ABSOLUTE CLOCK | 462628 | TERMINATIONS TO GO | 0 |
|----------------|---------|----------------|--------|--------------------|-------|
| BLOCK COUNTS | | | | | |
| BLOCK | CURRENT | TOTAL | BLOCK | CURRENT | TOTAL |
| 1 | 0 | 1001 | | | |
| 2 | 1 | 1001 | | | |
| 3 | 0 | 1000 | | | |
| 4 | 0 | 1000 | | | |
| 5 | 0 | 1000 | | | |
| 6 | 0 | 1000 | | | |
| 7 | 0 | 1000 | | | |
| 8 | 0 | 648 | | | |
| TER | 0 | 1000 | | | |

| FACILITY | NUMBER ENTRIES | AVERAGE TIME/TRAN | TOTAL TIME | AVAIL. TIME | UNAVAIL. TIME | CURRENT STATUS | PERCENT AVAILABILITY | TRANSACTION NUMBER SEIZING | PREEMPTING |
|----------|----------------|-------------------|------------|-------------|---------------|----------------|----------------------|----------------------------|------------|
| CKOUT | 1000 | 318.196 | 0.688 | | | A | 100.000 | | |

| QUEUE | MAXIMUM CONTENTS | AVERAGE CONTENTS | TOTAL ENTRIES | ZERO ENTRIES | PERCENT ZEROS | AVERAGE TIME/TRANS | SAVERAGE TIME/TRANS | TABLE NUMBER | CURRENT CONTENTS |
|---------|------------------|------------------|---------------|--------------|---------------|--------------------|---------------------|--------------|------------------|
| LINE 11 | 11 | 0.915 | 1001 | 326 | 32.6 | 422.671 | 626.806 | | 1 |

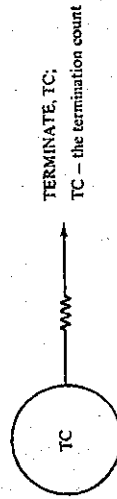
| NUMBER .. CONTENTS | NUMBER .. CONTENTS | NUMBER .. CONTENTS | NUMBER .. CONTENTS | NUMBER .. CONTENTS | NUMBER .. CONTENTS |
|--------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| 1 | 648 | | | | |

شکل ۱۰-۳۱ خروجی گزارش GPSS برای صف و یک سرور دهنده

ایراند N تعداد سرویس دهنده‌های موازی را بیان می‌کند. ایراند A شماره فعالیت انتخابی توسط کاربر و مبین تعریف گروه واحد سرویس دهنده‌ها است. آمارها بطور خودکار برای هر گروه از سرویس دهنده‌ها جمع آوری می‌شود. ایراند DUR مبین دوره (طول) فعالیت است که می‌تواند ثابت، تابعی از یک سیستم توزیع آماری، تصادفی و غیره باشد. در شکل ۱۰-۳۲، $A=1$ یعنی فعالیت شماره ۱ با $N=1$ یک سرویس دهنده و دوره $DUR=RNORM(3,2,0,6)$ که توزیع نرمال با میانگین $3/2$ و انحراف معیار $6/6$ واحد زمان (دقیقه) است.

هر انشعاب در یک مدل SLAM باید یک گره شروع و یک گره انتهایی داشته باشد. به عنوان نمونه، یک گره QUEUE شروع برای یک انشعاب خاص است. گره انتهایی می‌تواند هر گره دیگری باشد. NLBL در گره ACTIVITY می‌تواند برچسب گره نهایی را نشان دهد.

در شکل ۱۰-۳۲ گره نهایی انشعاب یک گره TERMINATE است که مبین یک پیش‌آمد خروج می‌باشد یعنی موجودیت سیستم را ترک می‌کند و محو می‌شود. سمبل آن، ایراند هایش و دستور مربوطه عبارتست از:



TC یک عدد صحیح مثبت است که وقتی موجودیهای عبوری از این گره به TC برسد شبیه‌سازی تمام می‌شود (در مثال 1000 است). اگر TC ذکر نشود فقط جمع آوری موجودیها را انجام می‌دهد و شبیه‌سازی را قطع نمی‌کند.

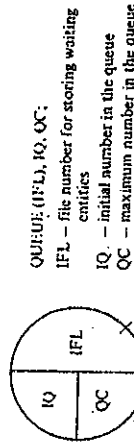
هر برنامه SLAM با دستور NETWORK آغاز و با ENDDNETWORK می‌یابد. شرح بیشتر در فصول آتی ارائه می‌شود. آمار خروجی در شکل ۱۰-۳۴ نشان داده شده است و بخشی از آن عبارتست از:

حداکثر طول صف $A =$

$0/7694$ کارایی اشتغال سرویس دهنده =

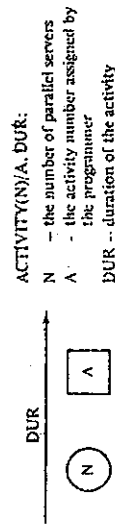
شرح SLAM در این فصل محدود است و برای مثال فوق ارائه و در فصول بعدی بطور کامل بیان می‌گردد. در شکل ۱۰-۳۲ $TBC=EXPON(4,5)$ یعنی زمان بین ورودیها دارای توزیع نمایی یا میانگین $4/5$ واحد زمانی است. ایراند Tf حذف گردیده که پیش‌فرض آن CREATE است لذا اولین ورودی در زمان صف شبیه‌سازی رخ می‌دهد. گره CREATE در شبکه به وسیله یک انشعاب که زمان بر نیست دنبال می‌شود. چنین انشعاباتی اتصال گره‌ها نام دارند و برای اتصال ۲ گره شبکه ترسیم می‌شوند. دستورالعملی برای یک اتصال نیاز نیست.

گره بعدی QUEUE مؤید یک تأخیر یا انتظار شرطی است و مؤید انتظار موجودیها تا شروع سرویس می‌باشد. سمبل گره QUEUE، دستور معادل آن و ایراند هایش عبارتند از:



ایراند IFL شماره فایلی است که موجودیها را تا شروع سرویس نگهداری می‌کند. در صورتی موجودیها نگهداری می‌شوند که تمامی سرویس دهنده‌ها مشغول باشند، در غیراین صورت بلافاصله سرویس یک موجودیت آغاز می‌گردد (پیش‌فرض سیاست صف FIFO). ایراند IQ پیش‌فرض 0 و پیش‌فرض QC نامحدود است. IQ یعنی در آغاز شبیه‌سازی صف خالی و QC حداکثر گنجایش طرف (فرایند) را دارد. در شکل ۱۰-۳۲، مقدار 1 دارد، IQ حذف شده‌اند، بنابراین خط انتظار فایلی 1، خالی و گنجایش نامحدود دارد. آمارها براساس میانگین فزات فایلی و میانگین زمان صرف شده آنها جمع آوری می‌شود.

در شکل ۱۰-۳۲ به دنبال گره QUEUE انشعاب ACTIVITY آمده است که سمبل، دستور مربوطه و ایراند هایش آن به صورت زیر است:



یک انشعاب مؤید یک فعالیت می‌باشد که یک تعریف ساده زمانی مانند زمان یک سرویس است.

۱۰.۸ - شبیه ساز سیستم صف با یک سرویس دهنده (پیش آمده های همزمان ورود خروج)

در این بخش به شبیه سازی یک سیستم صف می پردازیم که دارای یک سرویس دهنده می باشد و نکته قابل توجه همزمانی رخداد پیش آمده ها است. در این مثال به همزمانی رخداد پیش آمده های ورود و خروج توجه نمودیم.

عملیات ورود و سرویس به N مشتری را به وسیله یک سرویس دهنده انجام می دهیم. مشتریان به صورت شماره های $1, 2, \dots, N$ علامتگذاری شده اند. AT_k را زمان بین ورود مشتری $(k-1)$ و مشتری k در سیستم فرض می کنیم.

این زمانها با استفاده از توزیعهای تصادفی مشخص به عنوان نمونه تولید می شوند (با استفاده از یک مولد تصادفی) و بطور مشابه ST_k زمان سرویس مشتری k می باشد که $K = 1, 2, \dots, n$ زمان سرویس نیز تصادفی است. CAT_k زمان ورود مشتری k ام فرض می کنیم فرض می شود که صفی وجود ندارد و سرویس دهنده نیز آزاد است. اولین مشتری در زمان صفر می رسد و به سرویس دهنده می رود. پس از سرویس دیدن در زمان ST_1 از سرویس دهنده خارج می شود. دومین مشتری وارد می شود در زمان:

$$CAT_2 = AT_2$$

اگر $CAT_2 > ST_1$ ، مشتری دوم منتظر می گردد (یک صف با طول یک تشکیل می دهد) با برپرد انتظار:

$$WT_2 = ST_1 = CAT_2$$

WT_k زمان انتظار k امین مشتری در صف است. اگر $CAT_k > ST_1$ باشد خروج اولین مشتری قبل از ورود دومی انجام می شود. بنابراین سرویس دهنده یکبار می ماند که پرورد یکبارگی از عبارتست از:

$$IDT_2 = CAT_2 - ST_1$$

IDT_k یعنی زمانی که سرویس دهنده منتظر می شود تا مشتری k ام وارد شود. می توان فرض کرد که $(i-1)$ مشتری به سیستم وارد شده اند و $(i-1)$ مشتری سیستم را ترک کرده اند. پس مشتری بعدی که وارد می شود i امین و مشتری بعدی که خارج می شود i امین است. لذا:

$$1 \leq i \leq N$$

و اگر ادا باشد، طول صف عبارتست از:

زمان ورود بعدی $NAT = CAT_i$ می شود. زمان خروج بعدی جمعی زمان خروجها CDT_i ، i امین مشتری است.

$$NDT = CDT_i = \text{جمع زمان ورود ز} + \text{زمان انتظار ز} + \text{زمان سرویس ز} = CAT_i + WT_i + ST_i$$

SLAM SUMMARY REPORT

SIMULATION PROJECT SINGLE SERVER QUEUE

BY BANKS AND CARSON

DATE 5/31/1982

RUN NUMBER 1 OF 1

CURRENT TIME .4164E+04

STATISTICAL ARRAYS CLEARED AT TIME 0.

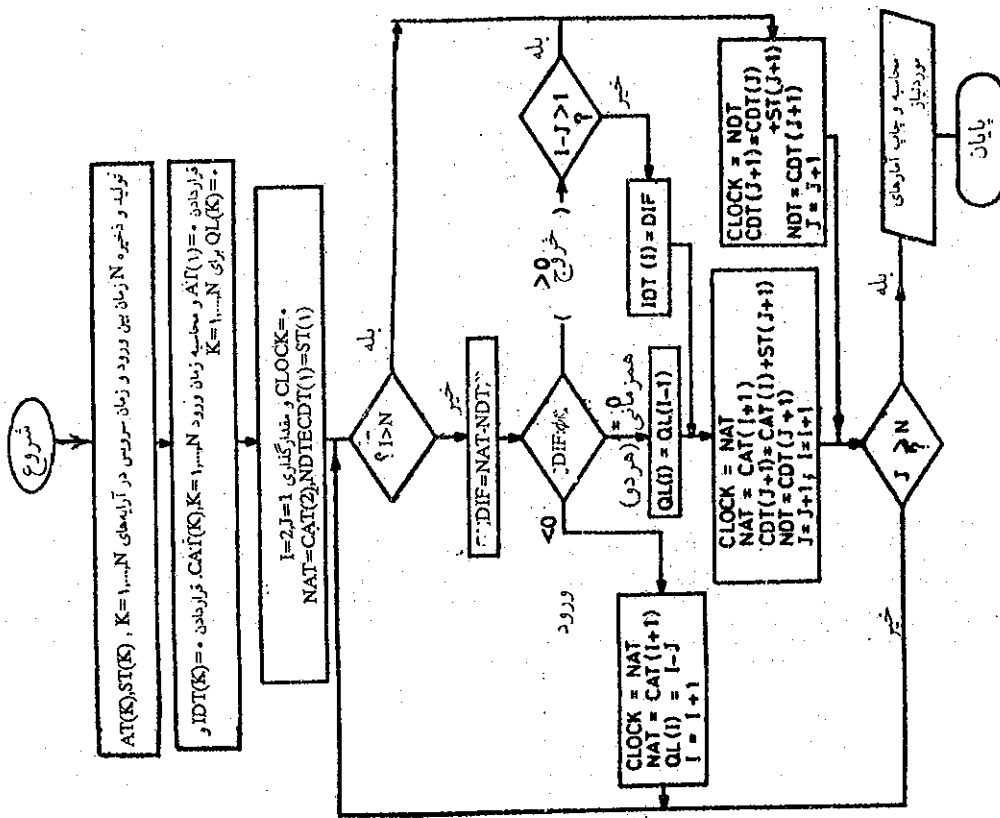
FILE STATISTICS

| FILE NUMBER | ASSOCIATED NODE TYPE | AVERAGE LENGTH | STANDARD DEVIATION | MAXIMUM LENGTH | CURRENT LENGTH | AVERAGE WAITING TIME |
|-------------|----------------------|----------------|--------------------|----------------|----------------|----------------------|
| 1 | QUEUE | 1.1221 | 1.4672 | 8 | 0 | 4.6677 |
| 2 | | 1.7694 | .4212 | 3 | 2 | 2.6523 |

SERVICE ACTIVITY STATISTICS

| ACTIVITY INDEX | START NODE LABEL/TYPE | SERVER CAPACITY | AVERAGE UTILIZATION | STANDARD DEVIATION | CURRENT UTILIZATION | AVERAGE BLOCKAGE | MAXIMUM IDLE TIME/SERVERS | MAXIMUM BUSY TIME/SERVERS | ENTITY COUNT |
|----------------|-----------------------|-----------------|---------------------|--------------------|---------------------|------------------|---------------------------|---------------------------|--------------|
| 1 | QUEUE | 1 | .7694 | .4212 | 1 | 0.0000 | 21.4036 | 232.2203 | 1000 |

شکل ۳۴-۱۰ گزارش خروجی SLAM صف و یک سرویس دهنده



شکل ۱۰-۳۵: فلوجارت شبیه‌سازی یک صف و یک سرویس‌دهنده با پیش‌آمدهای همزمان

NAT با NDT انجام می‌گیرد. با اختلاف

$$DIF = NAT - NDT$$

می‌توان گفت:

اگر $DIF < 0$ باشد، ورود صورت می‌گیرد و طول صف افزایش می‌یابد.

اگر $DIF > 0$ و $DIF \neq 1$ باشد، خروج صورت می‌گیرد و طول صف کاهش می‌یابد.

اگر $DIF = 0$ باشد، همزمانی ورود و خروج داریم.

روال شبیه‌سازی پیش‌آمد بعدی این صف ساده در فلوجارت شکل ۱۰-۳۵ نشان داده شده است.

در فلوجارت زمانهای بین ورود AT_i و زمانهای سرویس ST_i می‌تواند به وسیله صدا زدن یک زیربرنامه مناسب تولید شود. از زمانهای بین ورود، زمان ورود به راحتی محاسبه می‌شود:

$$CAT_K = CAT_{K-1} + AT_K$$

$$CAT_1 = AT_1 = 0$$

و زمانهای پیش‌آمد به وسیله متغیر CLOCK تعیین می‌شود. به عنوان خروجی آمارهای موارد زیر می‌تواند جمع شوند که بعضی از آنها به صورت زیر می‌باشند:

۱- ماکزیم طول صف.

۲- میانگین طول صف.

۳- میانگین طول زمان غیر خالی بودن صف.

۴- میانگین تعداد مشتریان در صف.

۵- احتمال آنکه m مشتری در سیستم موجود باشند.

۶- میانگین خط انتظار (تامی مشتریان).

۷- میانگین زمان انتظار مشتری منتظر.

۸- حداکثر زمان انتظار.

۹- میانگین زمان صرف شده توسط یک مشتری در سیستم.

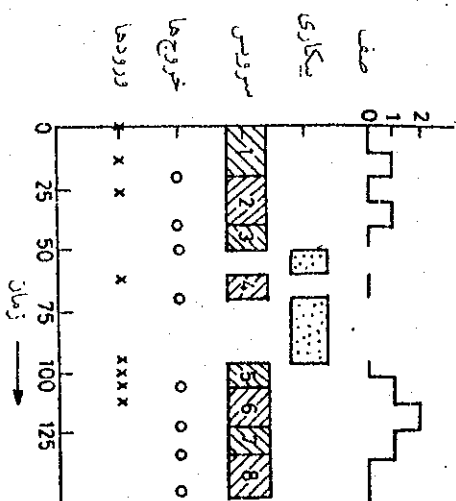
۱۰- کل زمان یکبارگی سرویس‌دهنده.

۱۱- درصد زمانی که سرویس‌دهنده بیکار است.

در فلوجارت بعد از تأمین ورود $N, N-1, 2, 1$ برادر $QL(I)$ طول صف را بدست می‌آورد. از این

طریق میانگین طول صف و ماکزیم آن بدست می‌آید. بطور مشابه آرایه $IDT(I)$ را محاسبه می‌کنیم که

زمان بیکاری سرویس‌دهنده وقتی برای $N, N-1, 2, 1$ برادر I را می‌دهد. از این



شکل ۱۰-۳-۱۰ صف و یک سرویس‌دهنده

۱۰.۹- شبیه‌سازی یک صف با دو سرویس‌دهنده

در دنیای واقعی بسیاری از سیستم‌های صف پیش از یک سرویس‌دهنده دارند لذا صف با چند سرویس‌دهنده عملی‌تر است. برای بیان بهتر مثال زیر ارائه می‌شود.

مثال ۱۰-۱۱- اکثریتها مطابق زمانهای ورود تصادفی به انگ‌گاه وارد می‌شوند چک می‌کند که کدام بارانداز خالی باشد کشتی به طرف باراندازی که خالی است می‌رود. اگر هر دو بارانداز خالی باشد کشتی به طرفی می‌رود که آن بارانداز برای مدت بیشتری بیکار می‌باشد. اگر هر دو بارانداز مشغول است، کشتی به صف عمومی وارد و سبب افزایش زمان انتظار خودش می‌شود و طول صف را یکی افزایش می‌دهد. کشتیها داخل صف به روش اولین ورودی اولین سرویس (FIFO) (تخلیه، بارگیری و هدایت) سرویس داده می‌شوند.

زمانهای سرویس هر بارانداز به عنوان نمونه‌هایی مستقل با توزیع‌های مشخص می‌تواند باشد. نه تنها توزیع زمان بین ورود بلکه توزیع زمانهای سرویس نیز نیاز به توزیع‌هایی دارد. وقتی که یک سرویس‌دهنده قبل از ورود کشتی بعدی بیکاری می‌شود تا وقتی که یک کشتی برسد و به بارانداز بیکار وارد شود زمان بیکاری رخ می‌دهد. این سیستم صف گرچه ساده بنظر می‌رسد برای مطالعه و تحلیل بسیار مشکل است. ما از طریق انجام شبیه‌سازی اطلاعات معمولی (مانند طول صف،

آرایه ماکزیمم زمان بیکاری یا میانگین زمان بیکاری سرویس‌دهنده بدست می‌آید. زمان انتظار (امین مشتری می‌شود:

$$WT_i = CDT_i + ST_i - CAT_i$$

سده مشتری به مدت است آن به راحی موجودند. و لذا WT_i باشد بدست می‌آید. از WT_i می‌توان کل زمان انتظار، ماکزیمم زمان انتظار یا میانگین انتظار را بدست آورد. در بعضی آزمایشات زمان انتظار مورد نیاز نیست بلکه کل زمان گذرانده شده مشتری در سیستم مد نظر می‌باشد که $WT_i + ST_i$. اگر آمارهای مختلف نیاز است یک زیربرنامه برای این امر طراحی می‌شود و در مواقع لازم صدامی زیم (جریانات آن به عهده خواننده است).

مثال ۱۰-۱۰- برای فهم بهتر روان شبیه‌سازی این صف مثال زیر را شرح می‌دهیم. فرض $N=8$ مشتری است و زمانهای بین ورود و زمانهای سرویس تولید عبارتند از:

$$AT_i = (0, 10, 15, 25, 30, 10, 5, 5)$$

$$ST_i = (20, 15, 10, 15, 15, 10, 10, 10)$$

$$i = 1, 2, \dots, 8$$

$$CAT_i = (0, 10, 25, 60, 90, 100, 105, 110)$$

تشخیص این پروسه در شکل ۱۰-۳۶ نشان داده شده است. هر خروجی یا دایره (O) و هر ورودی یا ضربدر (X) ارائه شده است. آرایه‌های تولید شده به وسیله روان می‌شود:

$$CDT_i = (20, 25, 25, 65, 105, 120, 130, 140)$$

$$i = 1, 2, \dots, 8$$

زمان خروج‌ها

طول صف بلافاصله پس از هر ورود:

$$Q_{it} = (0, 1, 1, 0, 0, 1, 1, 2)$$

$$IDT_i = (0, 0, 0, 1, 0, 25, 0, 0)$$

زمان بیکاری سرویس‌دهنده:

برنامه در زمان ۱۴۰ خاتمه می‌یابد یعنی وقتی که آخرین خروج انجام می‌شود. توجه کنید: در حالتی که دو پیش آمدید. همزمان رخ می‌دهند خروج مشتری شماره ۵ و ورود مشتری شماره ۷ در زمان ۱۰۵ است.

شکل ۱۰-۳۶- نیز یک نوع خروجی شبیه‌سازی می‌باشد که به صورت گرافیکی ارائه شده است. آمارهای مورد نیاز و بعضی از آنها به صورت گرافیکی و با شکل ارائه و البته در قالب بردارهای متعددی بدست آمده‌اند.

مشابه مقدار گذاری اولیه و تولید داده، آمارها نیاز دارند تا از طریق زیر برنامه زیر بدست می آیند.

STAT(CAT, CDT, IDT, QL)

که ۴ آرایه را استفاده می کند و به وسیله برنامه اصلی فراهم می شوند:

CAT(I), CDT(I,J), IDT(I,J), QL(I), $I=1,N, J=1,2$

فصل یازدهم

شبه سازی سیستم های انبار

پیش بینی و کنترل انبارداری

سیستم های انبار همانند سیستم های صف مطالعه، مدل، آنالیز و شبه سازی می گردند. مطالعه انبارها مانند خطر ط انتظار دارای محاسبات و ریاضیات جالبی است. تئوری صف و تئوری انبار از عملگرهای تحقیق در عملیات و علوم مدیریتی می باشند.

برادلی می گوید مشکل مدیریت انبار همان مشکل ذخیره موجودی کافی برای تعدادی از اقلام کالاها جهت تقاضای الگوی مورد نظر درخواست شده می باشد که ایجاد یک توازن مستدل ما بین هزینه نگهداری اقلام در انبار و جریمه از رده خارج شدن (کاهش فروش یا اضافه کاری) از آن جمله اند.

یک قلم کالا ممکن است یک محصول تولیدی فروخته شده به وسیله مغازه و یا قطعات یدکی ماشین یک کارخانه و یا واگن های راه آهن و یا میزان نقد کردن موجودی در بانک بر اساس تقاضای متقاضیان باشد و یا حتی میزان فارغ التحصیلان مهندسی در یک کشور بر اساس تقاضای کار باشد. براساسی یافتن یک متغیر عمومی و وسیع برای حل مشکلات گوناگون که بتواند مشکلات کنترل انبار را به صورت فرمولهای ریاضی تبدیل کند، بسیار مورد نیاز است. البته مدل های متفاوت در سیستم های انبار وجود دارد. همانکاری که در سیستم های صف انجام داده ایم، ابتدا در یک سیستم انبارداری خیلی ساده در نظر می گیریم و یک معادله تحلیلی برای کسب نمونه ای از تئوری پایه ای انبار بدست می آوریم.

۱۱.۱ - عناصر تئوری انبارداری

سه نوع هزینه در یک سیستم انبار وجود دارد و اهمیت آنها بستگی به سیستم تعریف شده دارد که این سه هزینه عبارتند از:

$$C = \frac{Q}{2} \cdot k + \frac{D}{Q} \cdot r$$

که در آن k هزینه روزانه انبارداری در قلم کالا، D میزان فروش روزانه و r هزینه سفارش مجدد یک نوبت تحویل کالا است. $\left(\frac{D}{Q}\right)$ تعداد روزهای مانده دو تحویل پس بدین است. حال دیفرانسیل C نسبت به Q را بدست آورده و $\frac{dC}{dQ}$ را مساوی صفر قرار می دهیم:

$$(11.1)$$

$$Q = \sqrt{\frac{2Dr}{k}}$$

مقدار Q در رابطه (11.1) میزان سفارش مجدد است. که کل هزینه انبارداری C به حداقل خواهد رسید. این میزان بهینه سفارش Q به عنوان (EOQ) حد اقتصادی سفارش موسوم است. پاسخ این که چه زمانی یک سفارش انجام شود؟ به وسیله معادله 11.1 داده می شود یک سفارش $Q = \sqrt{\frac{2Dr}{k}}$ کلاپی، در زمانهای ثابت $Q = \sqrt{\frac{2Dr}{k}}$ روز داده می شود و میانگین انبار $Q = \sqrt{\frac{2Dr}{k}}$ است. مشاهده می کنیم که این سطح میانگین انبار فقط بر حسب جذر تفاضای روزانه D افزایش می یابد. معادله (11.1) فرمول ویلسون یا فرمول اندازه کلا است که برای زمانهای بیشتر از ۵۰ سال مورد استفاده قرار می گیرد و بهترین فرمول در تئوری انبار است.

تاخیر ثابت تحویل کالا:

در ساده ترین سیستم انبار که از آن معادله 11.1 بدست آمده و ما هیچ زمانی برای تاخیر تحویل سفارش کالا در نظر نگرفته ایم. بنابراین مدلی با واقع بینی بیشتر با اضافه کردن مقدار ثابت غیر صفر L روز تاخیر تحویل می سازیم. بنابراین میزان تاخیر L را می دانیم همچنین زمانی که باید هر سفارش برسد را نیز داریم لذا نیازمندیم که سفارشات را L تاخیر انجام دهیم. این عمل در مدل پایه هیچ پیچیدگی ایجاد نمی کند چرا که زمان تاخیر L به صورت ثابتی باقی خواهد ماند. اما اگر L یک متغیر تصادفی باشد با پیچیدگی مواجه خواهیم بود.

EOQ با هزینه کمبود کالا: حد اقتصادی سفارش به وسیله معادله (11.1) مشخص است که برای یک سیستم انبار در هر مقدار مورد نظر ثابت می باشد و ما هیچ وقت تخلیه موجودی انبار نداریم. یک عبارت مشابه می تواند برای یک سیستم با تخلیه موجودی انبار جهت بخشی از زمان نیز بدست آید. معمولاً وقتی که فرشته تخته انبار دارد در نوع وضعیت اتفاق می افتد:

(۱) مشتری به اماکن مختلفی رفته و بنابراین هر فروشی که برای ما نتیجه داشته باشد از دست رفته تلقی می شود و هیچ فروشی نداریم.

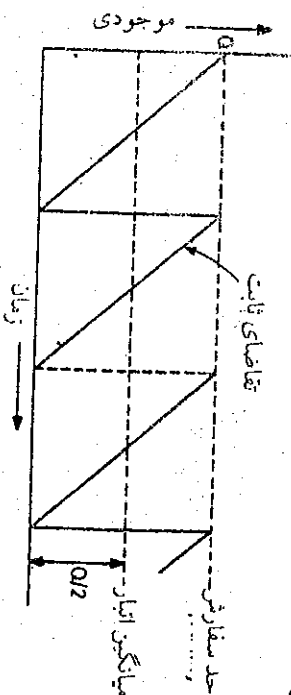
(۱) هزینه انجام یک سفارش که آن را هزینه سفارش مجدد (reorder cost) یا هزینه راه اندازی اولیه (Setup Cost) می نامند.

(۲) هزینه نگهداری در یک انبار که آن را هزینه انبارداری (inventory holding) یا هزینه حمل (Carrying Cost) نامند که شامل انبارداری، بهره حق بیمه و غیره می باشد.

(۳) هزینه کاهش سود عبارتست از کاهش سود درآمد، سرقتی و غیره زمانی که تخلیه انبار داریم پیش می آید.

دریافت سه مقدار والگوی تقاضا معمولاً در تعیین الف) زمان تجدید سفارش، ب) میزان سفارش مجدد بکار می رود تا آن که هزینه کل انبارداری حداقل گردد. البته بازنشرهای دیگری مانند تاخیر زمانی (lead time) یا تاخیر تحویل (delivery lag) برای سیستم نیز وجود دارد که زمان فاصله بین سفارش تا رسیدن کالا است.

ساده ترین مدل انبارداری با یک قلم کالا در شکل ۱-۱ آمده است. در مدل مذکور تفاضای روزانه کالا ساده ترین فرض می شود. زمانی که موجودی انبار به صفر نزول می کند، کالاها تجدید سفارش شده و عدد ثابتی فرض می شود. زمانی که سفارش انبار در $Q/2$ آغاز و با نرخ ثابت به صفر میل می کند و آنگاه بدون هیچ تاخیری تحویل می گردد. سطح انبار در یک سال Q فرض می شود.



شکل ۱-۱ مدل انبار با تقاضای ثابت

در این مدل خاص، ما هیچ وقت نیاز به تخلیه موجودی کالا نداریم و هیچ هزینه کسری کالا موجود نیست. هزینه انبار در هر روز (یا هر واحدی از زمان) برابر است با:

که در آن b هزینه کمبود موجودی انبار برای سفارش روزانه هر کالا است. با توجه به اینکه نیاز روزانه برابر D است،

یک سفارش مجدد هر $\frac{Q}{D}$ روز انجام می‌گیرد. بنابراین میانگین هزینه سفارش مجدد برای هر روز rD/Q است که در آن r هزینه سفارش مجدد می‌باشد. با بدست آوردن این سه هزینه خواهیم داشت، میانگین کل هزینه انبارداری در هر روز برابر:

$$C = \frac{kQ\alpha}{2} + \frac{b(1-\alpha)^2Q}{2} + \frac{Dr}{Q} = \frac{Q}{2} [k\alpha^2 + b(1-\alpha)^2] + \frac{Dr}{Q} \quad (11.2)$$

برای تعیین مقدار بهینه کسر α دیفرانسیل معادله (11.2) نسبت به α را مساوی صفر قرار می‌دهیم:

$$\frac{dC}{d\alpha} = \frac{Q}{2} [2k\alpha - 2b(1-\alpha)] = 0 \quad (11.3)$$

$$\alpha = \frac{b}{k+b}$$

با جایگذاری مقدار α در رابطه (11.2) کل هزینه خواهد شد:

$$C = \frac{Q}{2} \frac{bk}{k+b} + \frac{Dr}{Q} \quad (11.4)$$

به منظور بدست آوردن مقدار بهینه حد سفارش مجدد Q ، دیفرانسیل رابطه (11.4) نسبت به Q را مساوی صفر قرار می‌دهیم:

$$\frac{dC}{dQ} = \frac{bk}{2(k+b)} - \frac{Dr}{Q^2} = 0$$

که حد اقتصادی سفارش مجدد EOQ ، برای این سیستم مساوی است با:

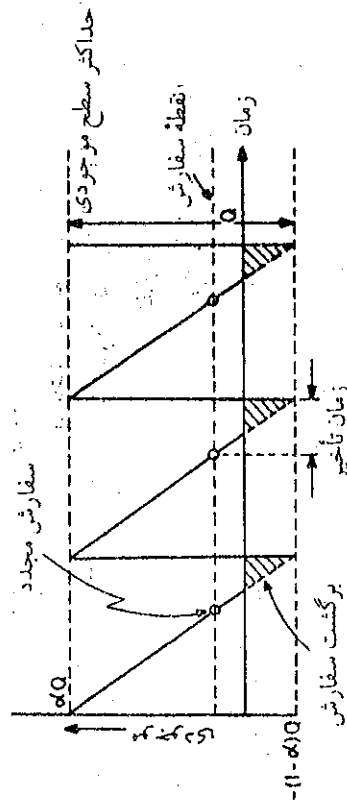
$$Q = \sqrt{\frac{2Dr}{k}} \cdot \sqrt{\frac{k+b}{b}} \quad (11.5)$$

اولین عامل در این معادله EOQ است که به وسیله معادله (11.1) ارائه شده و هیچ کمبودی وجود ندارد. دومین عامل بزرگتر از یک و ناشی از به وجود آمدن مقدار معین هزینه موجودی انبار که برابر b است می‌باشد. اگر این هزینه b از k بزرگتر گردد، معادله (11.5) به معادله (11.1) کاهش خواهد یافت. قابل توجه است که به علت هزینه کمبود b ، انداز بهینه سفارش Q افزایش اما حداکثر انبار αQ کاهش می‌یابد. میانگین هزینه روزانه در شرایط بهینه به وسیله جایگزینی معادله (11.5) در معادله

(۲) زمانی که مشتری سفارش خود را رها می‌کند و این موجودی باقی سفارش یا برگشت از سفارش به محض بازگشت موجودی پُر می‌شود.

با وجود این برای هر قلم کالا در هر واحد زمانی جریمه وجود دارد که باید برای هر موجودی سفارش واریز گردد. (هزینه نگهداری سفارش مجدد، هزینه خنل کالا به مشتری و غیره). اجازه دهید که نخست نوع برگشت سفارش را تعیین کنیم.

یک سیستم با ثابت تقاضای روزانه D ، حد سفارش مجدد Q و زمان ثابت غیر صفر تأخیر t که در شکل (۱۱.۲) نشان داده شده است را در نظر می‌گیریم. برای یک بخش α از زمان ما یک انبار غیر خالی و برای باقیمانده زمانه $(1-\alpha)$ برگشت از سفارش داریم (تأخیر هاشور زده زیر محور x ها). موجودی به ازای سفارش کالا به زودی در سفارش بعدی (از Q قلم کالا) پُر می‌شود.



شکل ۱۱-۲ نیاز ثابت با کمبود

برای چنین سیستم انبارداری داریم:
میانگین موجودی انبار روزها، زمانی که یک انبار مثبت داریم: $\frac{Q\alpha}{2}$

میانگین روزانه موجودی انبار، میانگین تمامی روزها $\frac{Q\alpha^2}{2}$

میانگین روزانه هزینه انبارداری $k \frac{Q\alpha^2}{2}$

بنابراین
و k هزینه روزانه نگهداری هر کالا است. بطور مشابه،

میانگین کمبود در انتای روزهایی که کمبودی وجود داشته باشد $\frac{(1-\alpha)Q}{2}$

میانگین کمبود برای تمامی روزها $\frac{(1-\alpha)^2Q}{2}$

بنابراین

$$\frac{b(1-\alpha)^2Q}{2} = \text{میانگین هزینه کمبود روزانه}$$

$$\alpha = \frac{b}{k+b} = \frac{.02}{.005 + .02} = 0.8$$

این مقدار برای این مثال عددی به خصوص می‌باشد. بنابراین $\alpha = 0.8$ تغییرات در درجه سروس قابل قبول باشد.

اما در بخش زیر میزان مفروض ساخته شده دارای بالاترین سطح اقتصادی می‌باشد. با صد درصد سطح سروس (بدون کمبود) هزینه برابر ۱۰ ریال در هر روز می‌باشد که در یک سطح $\alpha = 0.8$ میزان $1/10$ ریال برای هر روز ذخیره خواهد شد.

۱۱.۲- مدل‌های پیچیده‌تر انبار

تجلاً عبارات تحلیلی ساده برای دو مدل مختلف انبارداری ساده به دست آورده‌ایم و اینها فقط شامل نمونه‌ای از عبارات تحلیلی موجود در تئوری انبار هستند. فرمول‌های پیچیده‌تری در تئوری ریاضیات انبار با انواع مدل‌های مختلف انبار موجود هستند که فاکتورهای پیچیده‌ای دارند و در زیر آمده است. پرکردن تدریجی:

در محاسبه معادله (۱۱.۱) فرض کردیم که به محض رسیدن سفارش انبار پر می‌شود. در بیشتر مواقع به صورت تدریجی اتفاق می‌افتد. تقاضای پیوسته‌ای به چرخش محصولات تولید داده‌شود. اگر در طول پربرد تولیدات، q واحد محصول در هر روز تولید شود و D واحد فروش رده‌بازار با نرخ (p/q) واحد در هر روز با این پربرد افزایش می‌یابد. بر اساس معادله (۱۱.۳) یک محیط تولیدی به صورت معادله (۱۱.۷) نشان داده شده است:

$$Q = \sqrt{\frac{2Dr}{k(1-D/p)}} \quad (11.7)$$

که حد اقتصادی سفارش محصولات را ارائه می‌دهد.

تولیدات چندکاناله‌ای:

فرمول (۱۱.۷) فقط برای تولیدات تک محصولی به کار می‌رود و اگر چند محصول متفاوت موجود باشد آنها را در کل به صورت مستقل از یکدیگر نگاه‌داری می‌کنند. اغلب چندین قلم مختلف کالا وجود دارد که برنامه یکسان دارند. اول باید این اقلام در دسته‌های بزرگ تولید شوند که دسته‌های مختلف محصولات، گوناگون ایجاد می‌کنند. در چنین وضعیتی "سیکل تولید محصول" یک دسته در یک سیکل ایجاد می‌شود. می‌توان نشان داد که در این وضعیت تعداد بهینه سیکل‌های محصول در هر واحد زمانی به صورت زیر بدست می‌آید:

$$\sqrt{\frac{\sum D_i k_i (1 - D_i/p_i)}{22\eta}} \quad (11.8)$$

(۱۱.۳) بدست می‌آید:

$$C = \sqrt{\frac{2D\eta k}{k+b}} \quad (11.9)$$

که با یک فاکتور $\sqrt{\frac{b}{k+b}}$ از هزینه مربوط کمتر است اگر کمبود مجاز باشد.

برای شرح این فرمول‌ها اجازه دهید از مثالهای عددی استفاده کنیم:

$D =$ فروش روزانه، ۵۰ واحد در روز

$r =$ هزینه تحویل یک سفارش، ۲۰۰ ریال در هر سفارش

$k =$ هزینه حمل هر کار در روز، ۰.۰۵/ هر کار در هر روز

$b =$ هزینه کمبود کالا از هر برگشت از سفارش، ۰.۰۲/ در روز هر واحد برگشت از سفارش

این مقادیر را در رابطه (۱۱.۱) قرار می‌دهیم سپس حد اقتصادی سفارش با در نظر گرفتن هیچ‌گونه کمبودی خواهد بود:

$$\text{واحد} \quad \sqrt{\frac{2Dr}{k}} = 2,000$$

اگر کمبود وجود داشته باشد، با استفاده از رابطه (۱۱.۵) حد سفارش بهینه برابر است با:

$$\text{واحد} \quad 2,000 \times \sqrt{\frac{k+b}{b}} = 2,236$$

بعلاوه حداقل هزینه انبارداری کل که کمبودی وجود ندارد برابر خواهد بود با:

$$\text{ریال در هر روز} \quad \sqrt{2Drk} = 10$$

اگر کمبود وجود داشته باشد حداقل هزینه کل انبارداری برابر است با:

$$\text{ریال در هر روز} \quad 10 \times \sqrt{\frac{b}{k+b}} = 8.94$$

که کمبودی برابر $1/10$ ریال در هر روز را برای زمانی که کمبود وجود داشته باشد نشان می‌دهد. میزان α به عنوان بخشی از زمان برای اینکه ما دارای موجودی غیر صفر باشیم، یک عامل سطح سروس برای یک سیستم انبار است.

بیشترین میزان اقتصادی α به وسیله معادله (۱۱.۳) بدست می‌آید.

که در آن

$Dj =$ تقاضا برای زامین محصولات در واحد زمانی،

$Kj =$ هزینه حمل برای زامین محصول در واحد زمانی،

$rj =$ هزینه نگهداری برای زامین تولید و

$Pj =$ نرخ تولید برای زامین محصول است.

محدودیت سومیه:

در فرمول بدست آمده معادله (۱۸) این چنین فرض شده است که کمبود سرمایه وجود ندارد و

بوسیله فرمول EOQ افزایش انبار می دهیم همیشه این چنین نیست و محدودیت وجود دارد که مقدار

کل انبار نباید از x ریال تجاوز کند. معادله (۱۸) می تواند بنحوی مناسب جهت این امر تغییر یابد.

تخفیف مبلغ:

در بیشتر وضعیت های خرید با سفارشات دارای مبلغ زیاد رویه رو هستیم. این قیمت ممکن است

در نقاط مجزا یا در حالت های متوالی قطع شود. در اولین وضعیت وقتی به اندازه سفارش برابر یا

بیشتر از عدد معینی مانند ۱۰۰ و ۵۰۰۰ و غیره شود.

علت تخفیف قیمت یا شکست قیمت به سه عامل بستگی دارد:

(۱) وجود پس انداز مستقیم ناشی از کاهش قیمت.

(۲) کاهش هزینه کالا که باعث کاهش کالاست و هزینه حمل در هر واحد کاهش می یابد.

(۳) با وجود مزایای تخفیف که در سفارشات بزرگتر به وجود می آید ولی میانگین موجودی انبار

افزایش می یابد و از این رو هزینه حمل افزایش خواهد یافت.

به کار بردن این موارد در محاسبات برای هر محدوده کالا معادله ای است که هزینه کل انبار در هر

واحد زمانی را معین می کند. به وسیله دیفرانسیل این معادلات، برای هر محدوده قیمت EOQ تعیین

می شود. با این امر بهینه ترین هزینه انبار و در هر محدوده تخفیف بدست می آید. در دومین وضعیت

تغییرات کاهش قیمت بطور پیوسته با اندازه آن تغییر می کند. آنگاه هزینه سفارش مجدد r با مقدار زیر

$$r = Q \cdot q$$

چابگیرین می شود:

که در آن q نشانگر تخفیف موجود است و بصورت خطی با حد سفارش مجدد Q افزایش می یابد. به

نظر می آید که یک تخفیف نتواند چابگیرین عبارت EOQ در معادله (۱۱) شود. بنابراین

موقعیت هایی وجود دارد که ساختار قیمت آن ها بسیار پیچیده است و بصورت تحلیلی قابل حل

نیست.

تقاضای متغیر:

الگوی تقاضا در تمامی سیستم های انبار یک متغیر تصادفی است. در یک مثال ساده، در سیستم

انبار عادی ما فرض کردیم که نیازهای روزانه توزیع یکساخت از صفر تا ۹۹ داشته است. یک

چند جمله ای پواسن برای تقاضای روزانه بسیار واقع بینانه می باشد، زیرا از استقلال تصادفی ورود

مشتریان نتیجه گرفته شده است. از دیگر جهات اگر مشتریان بطور مستقل وارد شوند زمان بین

ورودشان توزیع نمائی خواهد یافت و اگر سفارش های هر مشتری فقط یک واحد باشد در کل نیاز به

توزیع چند جمله ای پواسن در هر پرپود دارد. با وجود این اگر تعداد کالاهای مورد نیاز از مشتری به

مشتری بطور تصادفی تغییر می کند ما توزیع های متفاوت برای تقاضاها بدست خواهیم آورد. که

وابسته به چگونگی تغییر تقاضای کالاهای مشتریان خواهند بود. برای مثال اگر کالاهای سفارشی هر

مشتری توزیع هندسی داشته باشد و زمان بین ورود مشتریان دارای توزیع فوق هندسی است. توزیع

نرمال تقریباً بیشتر چند جمله ای های پواسن، گاما و فوق هندسی را آن هم زمانی که میانگین افزایش

یابد خوب تقریب می زند. بنابراین اغلب تقاضاها بر اساس معادله نرمال برای بیشتر دوره های زمانی

در تمرینها یافت می شود. برای مثال اگر مقدار میانگین تعداد تقاضاها در یک دوره از شش یا بیشتر

تجاوز کند تقاضا تقریباً به وسیله معادله نرمال تقریب خواهد شد. تغییر تقاضا می تواند بوسیله دو

سیاست زیر دنبال شود: ۱- دوره های بازبینی دوره ای مثلاً یکبار در ماه است و سپس سفارش کالا

برای انبار با یک حد متغیر سفارش در صورتی که موجودی انبار به سطح مشخص برسد، باشد چنین

سیسی (شکل ۱۱-۳) به سیستم دوره ای بازبینی معروف است.

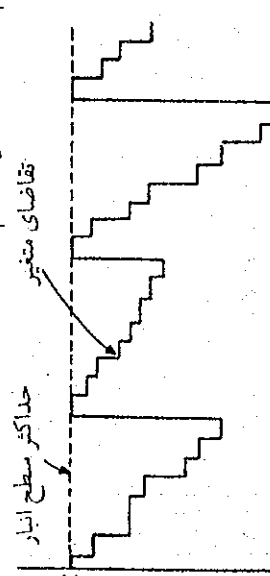
۲- به جای ثبات بازبینی و تغییر حد سفارش، مخالف عمل کنیم ثبات سطح سفارش Q با تغییر زمان

سفارش مجدد مطابق نیاز جایی که وضعیت انبار به یک سطح خاص P یا زیر آن برسد، یک سفارش

مشخص Q می دهیم چنین سیستمی که در شکل (۱۱-۳) نشان داده شده است به سیستم بازبینی

پیوسته یا سیستم گزارش تراکشی موسوم است.

تقاضای متغیر



شکل ۱۱-۳ سیستم بازبینی دوره ای

خروجی: خروجی نمونه سطح سروس و میانگین موجودی برای هر سیاست (P-Q) را می‌دهد. این خروجی، یک نقشه از سطح سروس و میانگین موجودی می‌باشد. نقاط پائین‌تر در این نقشه به وسیله خطوط راست متصل می‌شوند که در شکل ۶-۱۱ آمده است. این منحنی نمایش داده شده پائین‌ترین میانگین موجودی برای یک سطح سروس است. با توجه به شکل (۶-۱۱) ما فقط ۲۰ نقطه را نمایش داده‌ایم و نقاط باقیمانده برای جلوگیری از درهم شدن شکل حذف شده‌اند. سیاست‌های به کار رفته در منحنی نمایش داده شده جزو بهترین سیاست‌ها است.

آتهایی که بالا هستند نشان دهنده کم ظرفیتی می‌باشد چرا که آنها در نتیجه بالاتر نگه داشتن میانگین مقدار روزانه موجودی برای هر سطح سروس به وجود آمده‌اند. برای مثال گزینه (۹) نمونه یک سطح سروس با $۶۱/۲$ درصد و با نیاز به میانگین روزانه موجودی با $۲/۱$ واحد می‌باشد.

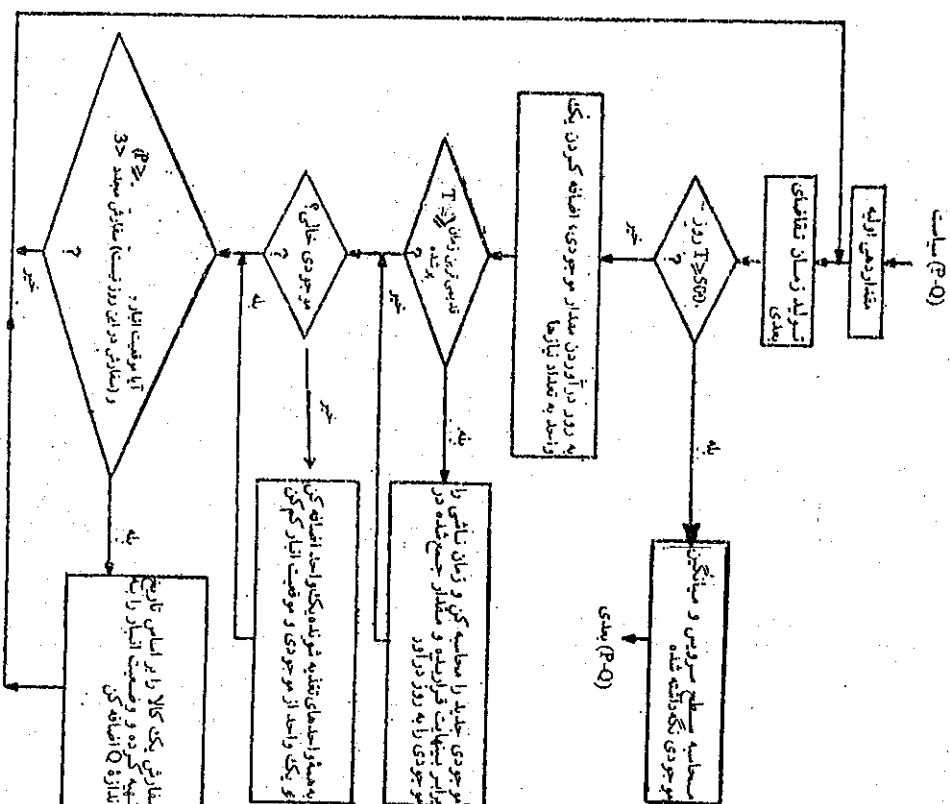
با اینکه گزینه (۱۱) با گرفتن سطح سروس حدود $۶۱/۲$ درصد و همچنین میانگین موجودی به کار رفته روزانه برابر $۳/۸$ واحد است.

آشکار است که دومی از سیاست انبارداری ضعیف‌تری برخوردار است. جدول (۱-۱) مقدار ۲۰ گزینه (حاصل از ۹۰ شیبه سازی) مرتب شده به ترتیب افزایش سطح سروس را داراست (گزینه‌ها مساری همان‌هایی است که در شکل ۶-۱۱ آمده است). اهمیت ویژه قابل مشاهده از جدول ۱-۱۱ (بشکل ۶-۱۱) آن است که بهترین سیاست‌ها کاملاً روی بهترین نقطه سفارش مجدد ۴ یا بهترین مقدار سفارش مجدد ۵ تنها پایه گذاری نشده است، بلکه روی ترکیبی از آن‌ها قرار دارد. این شیبه سازی هم چنین ثابت می‌کند که یک بازنگری به تعداد ۵ بدون در نظر گرفتن اثرات آن روی سطح سروس، حتی اگر این ۵ یک EOQ باشد کاری غیرعقلانی است. بنابراین یک خط منحنی همانند یک نمایش داده در شکل ۶-۱۱ برای مدیریت سیاست انبار بسیار خوبی به نظر می‌آید.

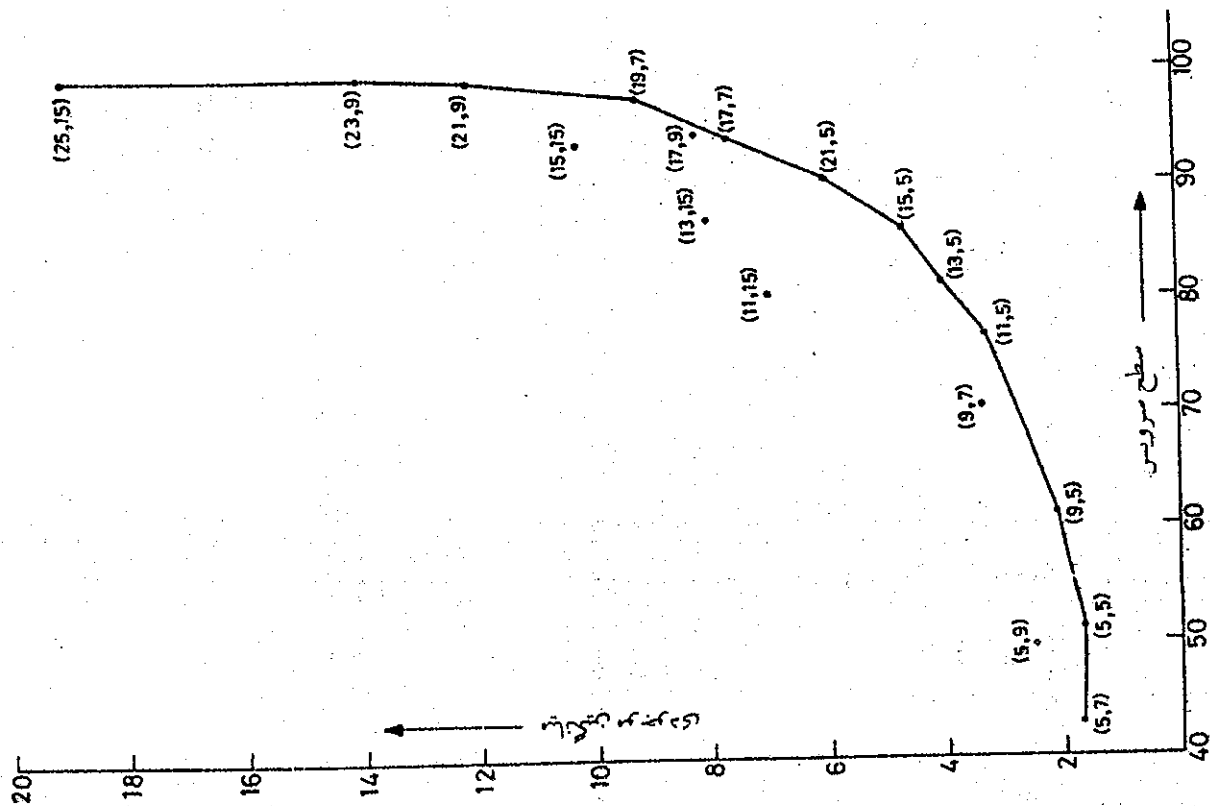
۱۱-۴ نمونه شیبه سازی شماره دو

در این نمونه دوم ما یک سیستم انبار را با ارزش‌های متفاوت از آنچه در بخش ۱۱-۱ شرح داده شده شیبه سازی می‌کنیم. هدف از شیبه سازی شعاع گسترده‌ای از سیاست‌های انبارداری به همراه یک نمایش از نیازهای حداقل می‌باشد. سیستم به وسیله پارامترهای زیر مشخص می‌گردد:

(۱) تعداد واحدهای مورد نیاز در چند جملهای پواسن به همراه یک میانگین مورد نیاز پنج واحد در هر روز می‌باشد.



شکل ۵-۱۱ شیبه سازی یک سیاست (P-Q) انبار



شکل ۱۱-۶ سطح سرویس و میانگین موجودی

| سطح سرویس | میانگین انبار | P تولید | Q فروش |
|-----------|---------------|---------|--------|
| 42.6 | 1.7 | 5 | 7 |
| 49.7 | 2.6 | 5 | 9 |
| 51.2 | 1.7 | 5 | 5 |
| 57.2 | 2.0 | 7 | 5 |
| 61.2 | 2.1 | 9 | 5 |
| 61.2 | 3.8 | 7 | 11 |
| 70.5 | 3.5 | 9 | 7 |
| 76.8 | 3.3 | 11 | 5 |
| 80.3 | 7.0 | 11 | 15 |
| 82.5 | 4.1 | 13 | 5 |
| 86.3 | 4.6 | 15 | 5 |
| 86.6 | 8.0 | 13 | 15 |
| 90.5 | 6.0 | 21 | 5 |
| 94.0 | 7.6 | 17 | 7 |
| 93.7 | 10.2 | 15 | 15 |
| 94.7 | 8.2 | 17 | 9 |
| 97.3 | 9.2 | 19 | 7 |
| 99.2 | 12.1 | 21 | 9 |
| 99.8 | 14.0 | 23 | 9 |
| 99.8 | 19.0 | 25 | 15 |

جدول ۱۱-۱ میانگین موجودی برای گرفتن یک سطح سرویس

زمان هدایت بدست آمده از طریق چند جمله‌ای آرنگ با پارامتر $m=3$ و میانگین زمان هدایت ۲۱

روز و $\beta=7$ است.

نیاز دیگری که نمی‌توانیم کامل کنیم بازنگری مجدد و قانع شدن است زمانی که یک محموله از

کارخانه برسد.

در هر زمان در نظر گرفته شده سه کالای برجسته وجود دارد.

مقادیر هزینه جمع شده برابرند با: مقدار بازنگری‌ها به اندازه ۲۰۰ عدد برای هر کالا، ارزش

حمل و نقل ۰/۲ در هر روز برای هر واحد و تعداد کالاهای برگشتی ۲ عدد در هر روز برای هر واحد

برای کلی کالاهای حمل شده.

زیر داریم: تعداد سفارشات انجام نشده و تاریخ ناشی از هر کدام. ما از یک آرایه دک بعدی به نام DB (برای تاریخ) با اندازه ۳ به همراه بخش های اطلاعاتی آن است. زمانی که یک سفارش انجام شده مرتبط داریم DD(1) زمان واقعی رسیدن سفارش به مغازه را نگه داری می کند. زمانی که یک سفارش انجام نمی شود DD(0) پیشترین مقدار خود را داراست (NNNTY) که اگر زمان شیپسازی فقط به ۵۰۰ رسیده باشد برابر ۹۹۹ است.

سپس در آغاز هر اجرا که هیچ سفارش انجام نشده نداریم می بایست برای ۳ تا ۱۲۳=۱ و DD(0)=NNNTY را در نظر بگیریم. قدم بعدی در این راه تعریف میانگین کل موجودی است و این امر وابستگی به روش بازگویی و هزینه حمل دارد. برای مثال ممکن است هزینه شناساژ فقط بر اساس موجودی واقعی هر شب یکبار اتفاق افتد.

در وضعیت دیگر بر اساس میزان سطح ماکزیم موجودی در هر روز پرداخت شود. در این شیپسازی ما از تعاریف ریاضی برای جمع آوری میانگین موجودی استفاده کرده ایم. مقدار واقعی موجودی را مابین دو پیش آمد پی در پی به وسیله فاصله زمان بین آن دو بدست می آوریم. مجموع همه تولیدات و حاصل تقسیم آن بر ۵۰۰ در طی مدت شیپسازی، میانگین واقعی جمع شده هر روز را در چرخه شیپسازی به ما می دهد. بنابراین میانگین مجموع موجودی روزانه برای هر زمانی مانند برابر رابطه (۱۱.۹) است:

$$(11.9) \quad \frac{1}{T} \sum_{t=1}^T (t_{i+1} - t_i) \cdot s_i$$

که در آن t_i زمان i امین اتفاق (در هر روز) است، t_{i+1} همان $(i+1)$ امین اتفاق است. s_i مقدار موجودی بلافاصله بعد از t_i است.

تقسیم کلی زمان شیپسازی NDAYS فقط، در پایان هر اجرا انجام می شود. در طی هر اجرا تعداد اتفاقات برابر رابطه (۱۱.۱۰) است.

$$(11.10) \quad (t_{i+1} - t_i) \cdot s_i$$

که به مجموع اجراها اضافه خواهد شد (که به آن مجموع نگه داری شده موجودی CHS می گویند). در روز برای راحتی این مقادیر را به دو قسمت جدا کرده ایم: قسمت اول فقط پاسخ پیش آمدهای تقاضا است. قسمت دوم تحویل سفارشات است که مابین تقاضاها اتفاق می افتد.

فلوچارت شکل ۵-۱۱ منطق شیپسازی را ارائه می کند. فلوچارت فقط برای یک ترکیب (P,Q) می باشد که دو حلقه خارجی برای برای P و Q متغیر جهت جلوگیری از ابهامات در شکل استفاده کرده ایم.

(۱) تقاضاها در واحدهای متغیر می رسند و زمانهای تقاضا از یک توزیع نمایی منفی با مقدار میانگین ۲۵٪ روز (برای میانگین ۴ واحد تقاضا در روز) تبعیت می کند.

(۲) کمپوها بعداً ما را دچار اشکال می کنند کم شدن فروش کمپوها را همیشه به همراه دارد.

(۳) زمان تأخیر بین سه و چهار روز تغییر خواهد کرد و بستگی دارد به زمانی که سفارش انجام گرفته است. یک سفارش در هر زمانی در طی ۴ امین روز در شروع (K+4) روز انجام می گیرد.

(۴) در هر زمان تقریباً سه سفارش انجام نشده داریم و در هر روز یک سفارش انجام می شود.

(۵) هزینه کل سفارش در مقایسه با هزینه حمل و نقل بسیار ناچیز است بنابراین تعداد سفارشات در یک دوره برای ما جالب نخواهد بود. ما فقط به مقادیر میانگین موجودی و تعداد کاهش فروش اهمیت می دهیم.

(۶) یک موجودی ده واحدی داریم و هیچ سفارش انجام نشده نداریم می خواهیم لیست سیاستهای (P,Q) ای که کمترین مقدار میانگین موجودی برای سطوح مختلف سروس ها را می دهند، پیدا کنیم. یک روز شیپسازی برای همه مقایسه های نقطه سفارش P از ۲۵ تا ۱۰۰ گام های دوتایی و حد سفارش Q از ۱۵ تا ۱۰۰ گام های دوتایی می نویسیم. برای هر کدام از این ها $66=100-34$ ، ترکیب مختلف در مقایسه با سطوح میانگین موجودی و سطوح سروس محاسبه شده و چاپ خواهد شد. زمان شیپسازی روز NDAYS=500 می باشد. موجودی را با ۱۰ واحد مقداردهی اولیه می کنیم و هیچ سفارش انجام نشده را در آغاز دوره شیپسازی نداریم.

درواقع شیپسازی تکرار مقادیر مختلف (P,Q) در دو حلقه اصلی است که با نقطه سفارش (حلقه P) داخل حد سفارش (حلقه Q) شروع می شود. برای هر اجزاء باید از مجموعه ای از اعداد تصادفی استفاده کنیم. به همراه هر اجرایی درواقع برای یک چرخه ۵۰۰ روزه مقداردهی داده ها عبارتند از: (CTD) کل تقاضاها، (TUS) کل واحدهای ذخیره شده و (CSH) مجموع موجودی مقدار صفر می گیرند. مقدار واقعی موجودی (Stock) با ۱۰ مقداردهی می شود و هیچ سفارش برگردانی وجود ندارد.

وضعیت ایبار (ANVPOS) نیز هم چنین با ۱۰ و زمان جاری (T) با صفر مقداردهی می شود. در این جا روش پیش آمد بعدی را تحقیقاً برای هر ۱۰۰ روز استفاده کرده ایم و دوباره پیش آمد در طی شیپسازی اتفاق خواهد افتاد:

الف) پیش آمد تقاضای بعدی.

ب) پیش آمد تحویل بعدی سفارش.

در زمانهایی که می توانیم بیشتر از ۳ سفارش داشته باشیم، نیاز به نگه داری دو مقدار اطلاع به صورت

سفارش اولی اتفاق افتد.

به بیان دیگر سفارشات همیشه در توالی یکسان برسند آن وقت زمانهای تاخیر نمی‌تواند از متغیرهای تصادفی مستقل تبعیت کنند. زیرا این فعل و انفعالات پیچیده مابین سفارشات برگزیده، مدل‌هایی با چندین سفارش، زمان‌های تاخیر اتفاقی، وابستگی به تقاضاهای تصادفی و مانند آن ممکن است فقط از طریق شبیه‌سازی مورد مطالعه قرار گیرد. بنابراین آنچه که به راحتی می‌توان دید این است که شبیه موقیعت صف، وضعیت عمر واقعی انبار بدون شبیه‌سازی پیچیده خواهد بود. و همیشه غیرممکن است که تاثیر نقش همه عملگرها در نظر گرفته شود.

در تجزیه و تحلیل ریاضی میانگین تقاضاها، میانگین زمان تاخیر و غیره در فرمول‌ها همانطور که در بخش ۱۱.۲ شرح داده شد فقط به صورت خطوط تقریبی به کار رفته‌اند. در واقع کنترل انبار یکی از مهمترین مسائل اساسی در تجارت و صنعت است. مدیریت می‌خواهد با کارکردن یک مجموعه هدفدار از قبل از مصرف آنها رضایت داشته باشد. سیاست‌های یک آزمایش شبیه‌سازی ممکن است اثرات پارامترهای مختلف را قبل از آنکه مشکل ساز شوند، مطالعه کند. بنابراین عملیات سیستم برای یک ماه یا یک سال در یک دقیقه شبیه‌سازی خواهد شد. نتایج دوره‌ها، هزینه کل انبارداری، کمبودها، تعداد رکوردها، میانگین سطح انبار و غیره برای سیاست‌های مختلف انبار می‌تواند مقایسه شود و از آن میان یکی را برگزید.

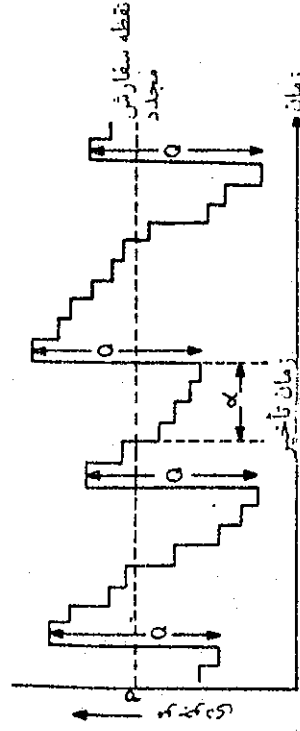
۱۱.۳ - نمونه شبیه‌سازی شماره یک

سطح سرویس برای یک سیستم انبار ممکن است براساس نرخ کالاهای ارائه شده برای کل واحدهای تقاضا شده تعریف شود. سطح سرویس که معمولاً بر حسب درصد بیان می‌شود، در سیاستهای تصمیم‌گیری انبار مهمترین عامل می‌باشد. در مدیریت اغلب اگر سطح سرویس پایین باشد آن سیاست انبارداری رد می‌شود. حتی اگر در مهمترین سطح اقتصادی باشد.

برای یک سطح پایین سرویس با ناتوانی‌هایی برای سرویس دادن به مشتریان ممکن است در طول یک اجرای طولانی، آن تجارت کلاً فلج شود.

در یک سطح سرویس طراحی شده، سطح میانگین موجودی بالا خواهد رفت که باید به صورت طبیعی نگهداری و کنترل شود. با وجود این برای گرفتن یک سطح سرویس ممکن است میانگین موجودی مختلفی وجود داشته باشد که به ترکیب (P, Q) ، بستگی دارد. می‌خواهیم با یک شمع بزرگی از سیاست‌های انبارداری به منظور بازدهی ترکیبات (P, Q) در بالاترین سطح سرویس برای یک مقدار میانگین موجودی تحلیل کنیم. حال مشخصات سیستم انبار را با کمک پارامترهایی دنبال می‌کنیم:

نیازمند آن است که برای هر وضعیت انبار نگهداری تمامی تراکشن‌ها انجام گیرد. آشکاراتی است که اگر آمارهای تقاضا موجود باشد، دلیل قانع‌کننده‌ای جهت خارج شدن از حد موجودی را ندارد و هیچوقت از سطح موجودی مورد نظر انبار دور نخواهیم شد. بنابراین همیشه وقتی که تقاضا تغییرات تصادفی دارد باید هزینه کمبود را در نظر داشت. برای انواع سیستم‌های ساده تصادفی، ممکن است عبارت تحلیلی برای میانگین هزینه مشخص سیاست انبارداری به خصوص را یافت. بنابراین یک عبارت تحلیلی برای عموم سیستم‌های انبار با تقاضای تصادفی وجود ندارد.



شکل ۱۱-۴ زمان بازبینی متغیر با حد سفارش ثابت

تغییر زمان تاخیر سفارش:

گرچه اغلب برای سادگی تاخیر زمان سفارش به عنوان چند روز ثابت فرض می‌شود در واقع این امر بندرت اتفاق می‌افتد. این زمان معمولاً تصادفی است و روشهای پیش‌بینی این متغیر نیز وجود دارد. این زمان از توزیع ارنلگ تبعیت کند. توزیع گاما با انتگرال روی پارامتر k ، در صورتی که $k=1$ باشد به صورت نمایی و در صورتی که k بسیار بزرگ باشد به صورت نرمال خواهد بود.

سفارشات ممتاز چندتایی:

در سیستم انبار ساده به روشنی بیان می‌شود که در هر زمان می‌توان یک سفارش مجدد برچسته داشت. در واقع ما با سیستم‌هایی روبه‌رو هستیم که دو یا چندین سفارش مجدد در آنها می‌توان برگزید. یک مدل که اجازه دهد، بیش از یک یا سفارش مجدد در یک زمان را برگزینیم و دارای زمان تاخیر تصادفی باشیم باعث به وجود آمدن شکل دشواری برای روش تحلیلی خواهد شد. برای اینکه اگر زمان سفارشات مجدد متوالی مستقل باشند در آن صورت ممکن است که سفارش دومی قبل از

خروجی: خروجی نمونه سطح سروس و میانگین موجودی برای هر سیاست (P-Q) راسی دهد. این خروجی، یک نقشه از سطح سروس و میانگین موجودی می باشد. نقاط پائین تر در این نقشه به وسیله خطوط راست متصل می شوند که در شکل ۶-۱۱ آمده است. این منحنی نمایش داده شده پایین ترین میانگین موجودی برای یک سطح سروس است. با توجه به شکل (۶-۱۱) ما فقط ۲ نقطه را نمایش داده ایم و نقاط باقیمانده برای جلوگیری از درهم شدن شکل حذف شده اند. سیاست های به کار رفته در منحنی نمایش داده شده جزو بهترین سیاست ها است.

آنهايي که بالا هستند نشان دهنده کم ظرفیتی می باشد چرا که آنها در نتیجه بالاتر نگه داشتن میانگین مقدار روزانه موجودی برای هر سطح سروس به وجود آمده اند. برای مثال گزینه (۹-۵) شماره یک سطح سروس با $61/2$ درصد و با نیاز به میانگین روزانه موجودی با $2/1$ واحد می باشد.

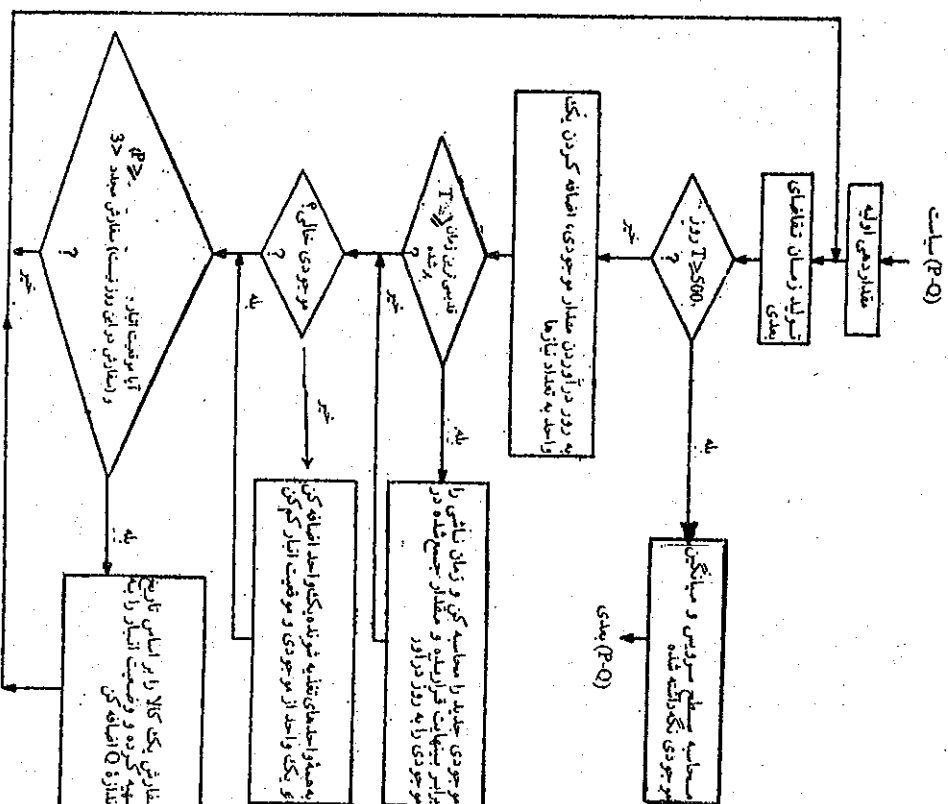
با اینکه گزینه (۱۱-۷) با گرفتن سطح سروس حدود $91/2$ درصد و همچنین میانگین موجودی به کار رفته روزانه برابر $3/8$ واحد است.

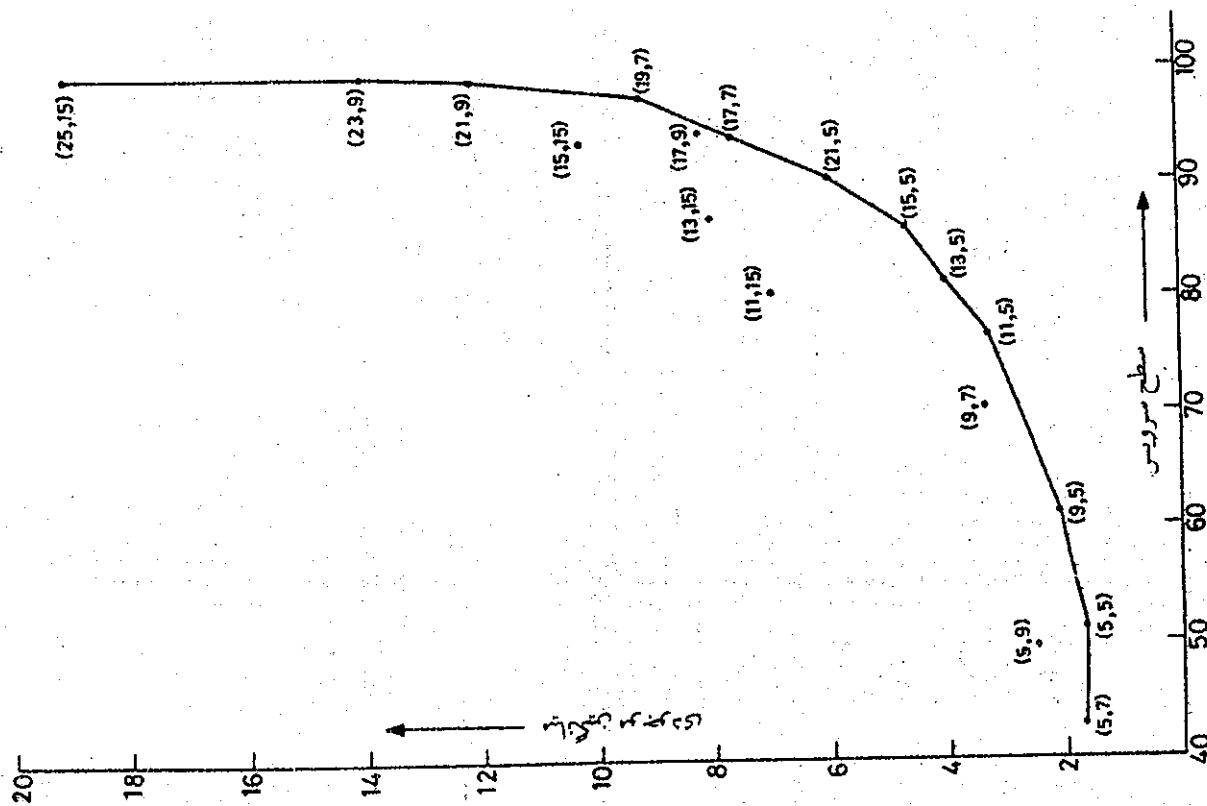
آشکار است که دومی از سیاست ابزار داری ضعیفتری برخوردار است. جدول (۱۱-۱) مقدار ۲۰ گزینه (حاصل از ۶۶ شییه سازی) مرتب شده به ترتیب افزایش سطح سروس را داراست (گزینه ها مساوی همان هائی است که در شکل ۶-۱۱ آمده است). اهمیت ویژه قابل مشاهده از جدول ۱۱-۱ (راشکل ۶-۱۱) آن است که بهترین سیاست ها کاملاً روی بهترین نقطه سفارش مجدد P یا بهترین مقدار سفارش مجدد Q تنها پایه گذاری نشده است، بلکه روی ترکیبی از آن دو قرار دارد. این شییه سازی هم چنین ثابت می کند که یک بازنگری به تعداد Q بدون در نظر گرفتن اثرات آن روی سطح سروس، حتی اگر این Q یک EOQ باشد کاری غیر عقلانی است. بنابراین یک خط منحنی همگام یک نمایش داده در شکل ۶-۱۱ برای مدیریت سیاست ابزار بسیار خوبی به نظر می آید.

۱۱-۴ نمونه شییه سازی شماره دو

در این نمونه دوم ما یک سیستم ابزار را با ارزش های متفاوت از آنچه در بخش ۱۱-۱ شرح داده شده شییه سازی می کنیم. هدف از شییه سازی شعاع گسترده ای از سیاست های ابزار داری به همراه یک نمایشی از نیازهای حداقل می باشد. سیستم به وسیله پارامترهای زیر مشخص می گردد:

(۱) تعداد واحدهای مورد نیاز در چند جمله ای پواسن به همراه یک میانگین مورد نیاز پنج واحد در هر روز می باشد.





شکل ۱۱-۶ سطح سرویس و میانگین موجودی

| سطح سرویس | میانگین انتظار | P تولید | Q فروش |
|-----------|----------------|---------|--------|
| 42.6 | 1.7 | 5 | 7 |
| 49.7 | 2.6 | 5 | 9 |
| 51.2 | 1.7 | 5 | 5 |
| 57.2 | 2.0 | 7 | 5 |
| 61.2 | 2.1 | 9 | 5 |
| 61.2 | 3.8 | 7 | 11 |
| 70.5 | 3.5 | 9 | 7 |
| 76.8 | 3.3 | 11 | 5 |
| 80.3 | 7.0 | 11 | 15 |
| 82.5 | 4.1 | 13 | 5 |
| 86.3 | 4.6 | 15 | 5 |
| 86.6 | 8.0 | 13 | 15 |
| 90.5 | 6.0 | 21 | 5 |
| 94.0 | 7.6 | 17 | 7 |
| 93.7 | 10.2 | 15 | 15 |
| 94.7 | 8.2 | 17 | 9 |
| 97.3 | 9.2 | 19 | 7 |
| 99.2 | 12.1 | 21 | 9 |
| 99.8 | 14.0 | 23 | 9 |
| 99.8 | 19.0 | 25 | 15 |

جدول ۱۱-۱ میانگین موجودی برای گرفتن یک سطح سرویس

(۲) زمان هدایت بدست آمده از طریق چندجمله‌ای از رنگ با پارامتر $m=3$ و میانگین زمان هدایت ۲۱ روز و $\beta=7$ است.

(۳) نیاز دیگری که نمی‌توانیم کامل کنیم بازنگری مجدد و قانع شدن است زمانی که یک محموله از کارخانه برسد.

(۴) در هر زمان در نظر گرفته شده سه کالای برجسته وجود دارد.

(۵) مقادیر هزینه جمع شده برابرند با: مقدار بازنگری‌ها به اندازه ۲۰۰ عدد برای هر کالا، ارزش

حمل و نقل ۰/۲ در هر روز برای هر واحد و تعداد کالاهای برگشتی ۲ عدد در هر روز برای هر واحد

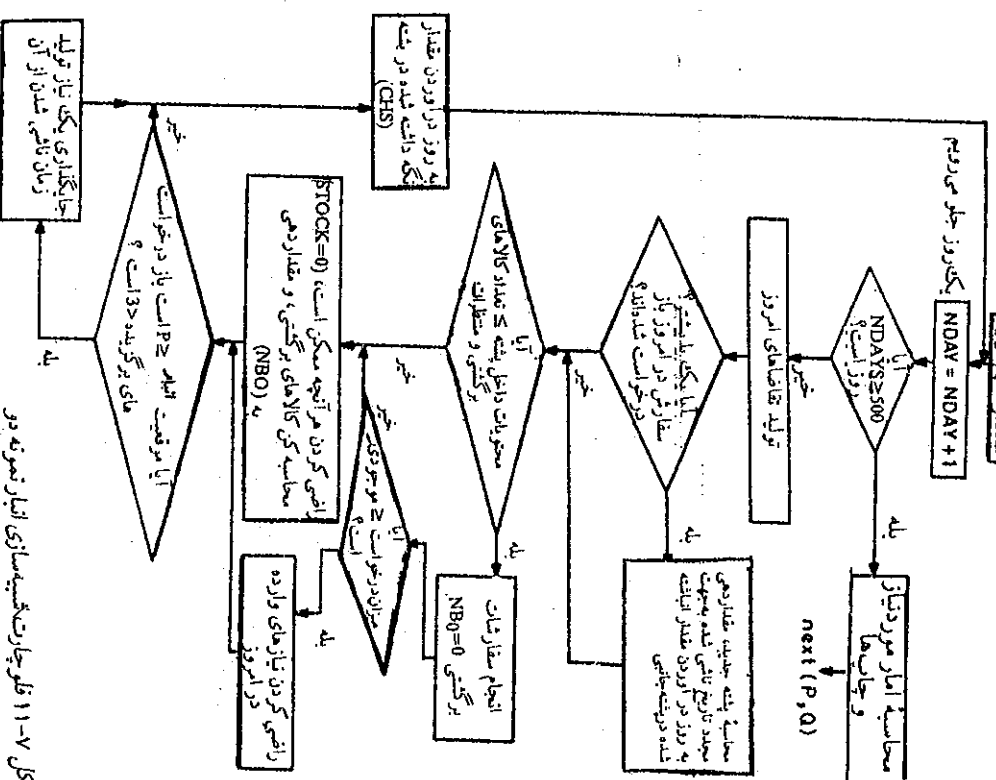
برای کل کالاهای حمل شده.

می‌داریم و میانگین همه کالاهای رسیده را بدست می‌آوریم بنابراین میانگین موجودی جانبی برابر

$$\text{است با: } ABSTOK = \frac{1}{NORD} \sum B_i,$$

که در آن NORD تعداد کل بازنگری‌های رسیده در طی زمان شبه‌سازی می‌باشد. در این جای مقدار قابل اهمیت در میانگین زمان هدایت \bar{T} است و آن چیزی نیست جز تفاوتی مابین نقطه بازنگری و میانگین بافر موجودی که $\bar{T} = P - ABSTOK$ است. فلو چارت شکل ۷-۱۱ مطلق این شبه‌سازی را شرح می‌دهد.

سیاست
می‌دهد.
Policy (P, Q)



شکل ۷-۱۱ فلو چارت شبه‌سازی انبار نمونه دو

ما برای همه ترکیباتی از P نقطه بازنگری از ۱۰۰ تا ۱۵۰ و تعداد Q بازنگری از ۵۰ تا ۱۴۰ و هر دو در کام‌های ده‌تایی را شبه‌سازی می‌کنیم.

برای هر ترکیبی ما می‌خواهیم که:

- (۱) میانگین فروش روزانه، ADASALE
- (۲) میانگین کالاهای برگشتی روزانه ADBO
- (۳) میانگین روزانه انبار AVSTOK
- (۴) میانگین موجودی جانبی انبار ABSTOK
- (۵) مقدار میانگین روزانه برای هر یک از سه عنوان یا مقدار حمل شده CCOST یا مقدار بازنگری RCOOST و مقدار کمبود S-COST و غیره محاسبه و چاپ می‌نمایم. عامل بدست آوردن زمان هدایت در این جا یک متغیر تصادفی از رنگ است که نیاز به هیچ پیچیدگی ندارد. ما فقط رویای روی یک تابع برای تولید متغیرهای از رنگ استفاده می‌کنیم. در این جا می‌توانیم حدود سه بازنگری داشته باشیم. بازنگری چندتایی می‌تواند به کمک یک آرایه یک بعدی (DD) با سایز ۳ نگهداری شود که برای نمونه شبه‌سازی در قسمت ۱۱-۳ انجام داده بودیم. در این جا کاهش فروش وجود ندارد. هر بازخاتی نیاز به بازنگری و معمولاً یک ساخت خوب دارد بنابراین مقدار کمبودها برابر تعداد کالای بازگشتی است. راجع به خورده‌چی، در این روش آمار زیادی را جمع می‌کنیم و آنگاه زودتر از مثال قسمت ۱۱-۳ آنها را انجام می‌دهیم. مگر برای موجودی جانبی که آمارهای دیگری تعریف شده و بعداً شرح داده خواهد شد.

موجودی جانبی:

موجودی در زمان رسیدن یک بازنگری موجودی جانبی یا موجودی مطمئن را ضداً می‌زند. در پیشنهاد نامبرده هدف موجودی جانبی قانع کردن نیازهای روزانه زمان هدایت است، بدون اینکه نیاز به زمان هدایت بیشتر از مقدار موجود باشد. در یک سیستم نیازی به نگهداری هیچ بافر جانبی نداریم و می‌توانیم P نقطه بازنگری روی STACK بر روی یک سطح صفر تا زمان رسیدن بازنگری جدید را نگه داریم. در این جا هم موجودی را خالی و هم موجودی جانبی را صفر می‌کنیم. بنابراین در شرایط یک سیستم تصادفی به منظور اجتناب از خالی شدن موجودی، نقطه بازنگری P معمولاً در یک سطح بالایی نسبت به میانگین زمان هدایت قرار می‌گیرد. می‌توانیم موجودی جانبی را سرعاً بعد از رسیدن اولین بازنگری به صورت زیر بدست آوریم: $B_i = P - t_i$

که در آن t_i به کل اختلافات مورد نیاز ما بین چرخه قرار گرفتن اولین کالا تا رسیدن آن کالا است. به منظور بدست آوردن میانگین بافر جانبی، مقدار STACK را برای هر زمان رسیدن مجدد یک کالا نگه

فصل دوازدهم

شبیه‌سازی شبکه ارزیابی و تجدیدنظر در برنامه‌ها (PERT)

۱۲- روش ارزیابی و تجدیدنظر در برنامه‌ها (PERT)

تا سال ۱۹۵۰ روش مؤثری برای برنامه‌ریزی، کنترل و مدیریت پروژه‌های بزرگ وجود نداشت. در سالهای ۱۹۵۹ توسعه روش مسیر بحرانی (CPM) و روش ارزیابی و تجدیدنظر در برنامه‌ها (PERT) کمک بسیار بزرگی محسوب می‌شد. هر دو روش CPM و PERT ارتباط بین "فعالتهای" مختلف یک پروژه را مشخص می‌کنند و فعالتهای بحرانی که نیاز به توجه بیشتر دارند را تعیین می‌نمایند. CPM برای وقتی به کار می‌رود که زمان هر فعالیت قطعی و معین است و به طریق مشابه PERT برای وقتی که زمان قطعی هر فعالیت را نداریم و سه نوع زمان برای آن طرح می‌کنیم، استفاده می‌شود. از این رو CPM که براساس یک تقریب زمانی قرار دارد ساده‌تر از PERT می‌باشد.

علاوه بر این آنالیز یک پروژه از طریق PERT براساس سه تقریب زمانی برای هر فعالیت قرار دارد که عبارتند از:

- زمان خوش‌بینی کامل (کمترین)
- زمان بدبینی کامل (بیشترین)
- زمان عمومی کامل شدن (محتمل‌ترین)

به جای یک توزیع احتمال، سه تقریب زمانی جهت رسیدن به نتیجه بهینه دارد. اگر محتمل‌ترین زمان را T_m ، زمان خوش‌بینی T_s و زمان بدبینی T_L باشد، زمان مورد نیاز یعنی T_e عبارتست از:

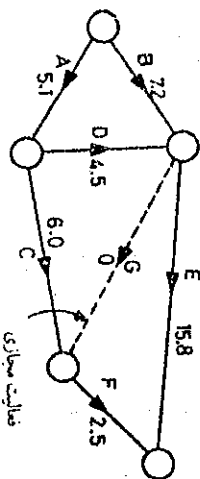
$$T_e = (T_m + T_s + T_L) / 6$$

شبیه‌سازی برای هر فعالیت از یک توزیع آماری متقنی استفاده می‌کند و یک نتیجه قابل انعطاف ارائه می‌دهد. در این فصل به بررسی و شبیه‌سازی PERT می‌پردازیم.

۱۲.۱ مدل شبکه‌ای یک پروژه

معمولاً یک پروژه می‌تواند از طریق تعدادی فعالتهای مجزا انجام شود. برای مثال پروژه ساختمان سازی دارای فعالتهایی از قبیل: انتخاب مکان، تصمیم‌گیری روی معماری، قرار دادن اسکلت و غیره

یک "کار" نیست و نام "فعالیت مجازی" دارد. فعالیت‌های مجازی وقتی نیاز است که در فعالیت‌های موجود دقت ارتباطات قبلی کافی نیست و نیاز به زمان و سایر منابع ندارد.



شکل ۱۲-۲ فعالیت مجازی در یک شبکه

تعداد ورودی در یک گره "درجه ورودی" ^(۱۱) آن گره و تعداد خروجی‌های یک گره "درجه خروجی" ^(۱۲) می‌نامند. یک شبکه، حداقل یک گره با درجه ورودی صفر و حداقل یک گره با درجه خروجی صفر دارد. اگر بیش از یک گره با درجه ورودی صفر باشد، می‌توانیم یکی از آن‌ها را انتخاب کنیم و خطوط مجازی ترسیم شده این گره به دیگر گره‌ها برود. شبکه حاصل یک گره با درجه ورودی صفر دارد. این گره، گره منبع ^(۱۳) و گره‌ای که با درجه خروجی صفر است، "گره نفوذ" ^(۱۴) نام دارد. گره منبع مولد شروع و گره نفوذ اتمام پروژه را نشان می‌دهد. پس هر شبکه دارای یک گره منبع و گره نفوذ است.

۱۲-۲-۱ آنالیز یک شبکه فعالیت

شبکه‌های فعالیت به وسیله مدیران برای مطالعه و شبیه‌سازی اجرای پروژه، آنالیز می‌شوند. دوتا از عمومی‌ترین این اتصالات PERT و CPM است. توالی خطوط پشت سرهم از گره منبع به گره نفوذ در شبکه "مسیر" ^(۱۵) نام دارد. برای مثال در شکل ۱۲-۲، ADE یک مسیر و BGF مسیر دیگر است. طول یک مسیر، جمع زمان تمامی فعالیت‌های در طول آن مسیر می‌باشد. برای مثال در شکل ۱۲-۲ داریم:

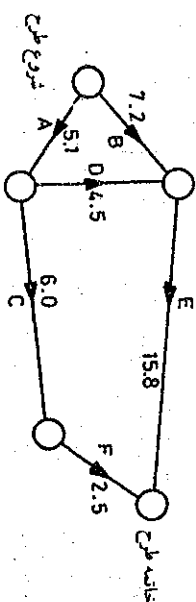
| | | | |
|------------|-------------------------|-----|----------|
| طول ADF | $= 5.1 + 4.5 + 15.8$ | روز | $= 25.4$ |
| طول $ADGF$ | $= 5.1 + 4.5 + 0 + 2.5$ | " | $= 12.1$ |
| طول ACF | $= 5.1 + 6.0 + 2.5$ | " | $= 13.6$ |
| طول BGF | $= 7.2 + 0 + 2.5$ | " | $= 9.7$ |
| طول BE | $= 7.2 + 15.8$ | " | $= 23$ |

- 1- Job
- 2- in-degree
- 3- out-degree
- 4- source
- 5- sink
- 6- path

است. بنابر دلایل تکنیکی بعضی فعالیت‌ها نمی‌توانند قبل از بعضی دیگر شروع و کامل شوند.

لذا این فعالیت‌ها تقدم ارتباطات و محدود دارد. علاوه بر فعالیت زمان معین نیاز دارد که دوره فعالیت نامیده می‌شود. فعالیت‌ها با ارتباطات یا روابط فیما بین و دوره‌های زمانی، به وسیله گراف فعالیت یا "شبکه فعالیت" ارائه می‌گردند. فعالیت‌ها به صورت خطوط مستقیم بیان می‌شوند (فلش، کمان یا انتخاب) و گره‌ها شروع و پایان (تکین) فعالیت را نشان می‌دهند که پیش‌آمدها ^(۱۶) یا بار سنگین ^(۱۷) پروژه هستند. یک فعالیت که به وسیله پیکان (یا) ارائه می‌شود نمی‌تواند قبل از تمامی فعالیت‌هایی که به گره ا ختم می‌شوند کامل گردد.

مثال ۱۲-۱ فرض یک پروژه شامل شش کار معین بدون روهم انتخابی، A، B، C، D، E، F باشد و A باید قبل از C و D؛ D باید قبل از E و همچنین C قبل از F انجام شود. دوره فعالیت‌ها ۱/۵، ۲/۵، ۱۷/۲، ۱۶/۱، ۱۵/۸، ۴/۵ و ۲/۵ روز است شبکه فعالیت این پروژه در شکل ۱۲-۱ نشان داده شده است.



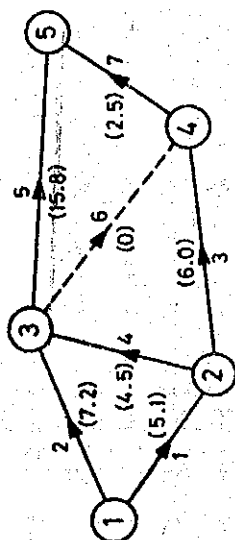
شکل ۱۲-۱ شبکه فعالیت مثال ۱۲-۱

این پروژه "سیکل" ^(۱۸) ندارد، در غیر اینصورت یک موقعیت غیرممکن داریم که هیچ فعالیت در سیکل قابل انجام نیست.

فعالیت مجازی ^(۱۹)

در مثال ۱۲-۱ حالتی داریم که فعالیت F قبل از مقداردهی B و D کامل نمی‌شود. می‌توان فعالیت G را با کشیدن یک پیکان اضافه در شکل ۱۲-۲ با خط چین نشان داد. این خط چین نقطه یک ارتباط است

- 1- events
- 2- milestone
- 3- cycle
- 4- Dummy



شماره فعالیت K

| | $S(k)$ | $F(k)$ | $T(k)$ |
|---|--------|--------|--------|
| 1 | 1 | 2 | 5.1 |
| 2 | 1 | 3 | 7.2 |
| 3 | 2 | 4 | 6.0 |
| 4 | 2 | 3 | 4.5 |
| 5 | 3 | 5 | 15.8 |
| 6 | 3 | 4 | 0 |
| 7 | 4 | 5 | 2.5 |

شکل ۱۲-۳ فعالیت‌های راسته هم‌بندی و ارائه آن

بنابراین، قدیمی‌ترین فعالیت K می‌تواند در زمان آخرین فعالیت تکمیل شده در $S(K)$ کامل گردد. فرض $EST(K)$ زمان قدیمی‌ترین شروع و $EFT(K)$ زمان قدیمی‌ترین پایان K امین فعالیت باشد. بنابراین:

$$EFT(k) = EST(k) + T(k) \quad (12.1)$$

صدا کردن گروه‌هایی است که پیش آمده‌ها را ارائه می‌کند. یک پیش آمد واقعی که تمامی فعالیت‌های قطع شده در گره k کامل می‌شود، انجام یافته تلقی می‌گردد. بنابراین:

$$ENT(0) = \text{MAX} (EFT(i) \text{ در آغاز شده و داریم}) \quad (12.2)$$

لذا قدیمی‌ترین فعالیت K وقتی که گره $S(K)$ انجام شود، می‌تواند آغاز گردد و داریم:

$$EST(k) = ENT(S(k)) \quad (12.3)$$

پیشگرایی

حال پسایش شبکه را از گره منبع به گره نفوذ از طریق هر خط در مسیر پیشگرا انجام می‌دهیم. هدف این پیش‌گرایی محاسبه قدیمی‌ترین زمان شروع، قدیمی‌ترین زمان خاتمه تمامی فعالیت‌ها یا

طراحی‌ترین مسیر، "مسیر بحرانی" نام دارد. فعالیت‌های A و D در شکل ۱۲-۲ بحرانی هستند. طول یک مسیر بحرانی زمان تکمیل پروژه است. بهترین بخش آنالیز یک شبکه، تعیین فعالیت‌های بحرانی و تعیین زمان کامل شدن پروژه است. یک تأخیر در اجرای فعالیت‌های بحرانی تأخیر در کل پروژه است. در یک شبکه ممکن است بیش از یک مسیر بحرانی باشد. فهرست فعالیت‌های یک پروژه را "راسته هم‌بندی" (۱۲) گویند. اگر هیچ فعالیتی در فهرست ظاهر نشود قبل از اینکه تمامی پیشینه بلافاصله آنها آماده شده باشد.

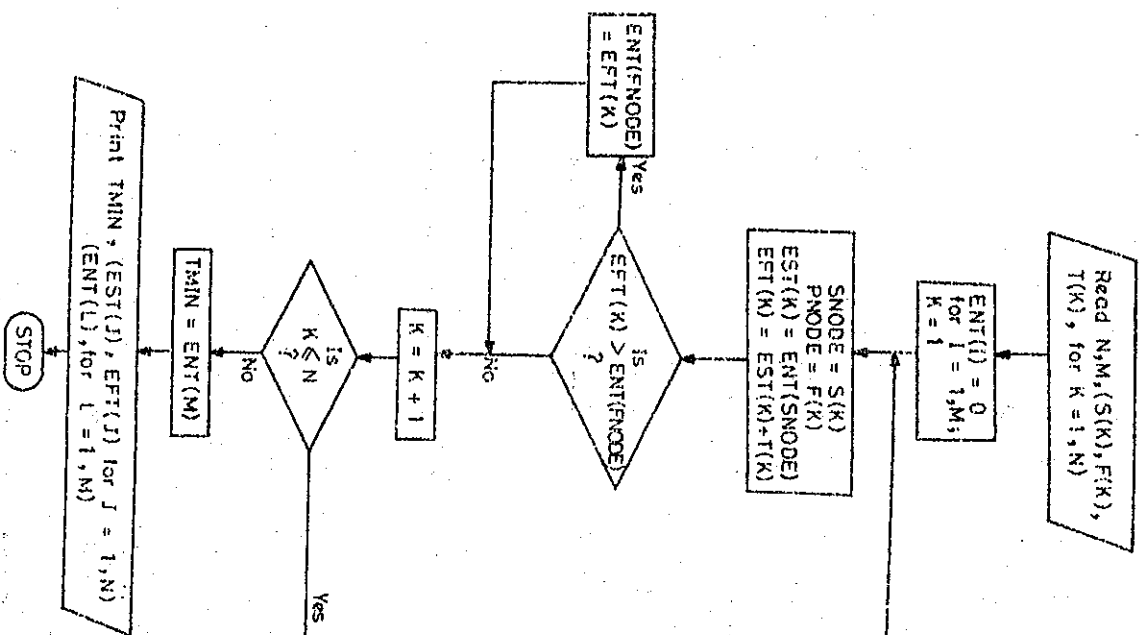
برای مثال در شکل ۱۲-۲ فهرست (A, B, C, D, E, G, F) یک فهرست راسته هم‌بندی است اما نیست زیرا فعالیت F قبل از G ظاهر شده و G پیش نیاز F است.

معمولاً بیش از یک راسته هم‌بندی ممکن وجود دارد. همین‌طور (B, A, D, C, E, G, F) نیز یک راسته هم‌بندی است. یک راسته هم‌بندی اگر فقط اگر سیکل در شبکه نباشد می‌تواند نشان داده شود. ما با یک فعالیت آغاز می‌کنیم که پیش نیاز ندارد. این را اول در فهرست می‌گذاریم و این فعالیت را از شبکه خارج کنیم و فعالیت دیگری را که پیش نیاز ندارد از بقیه شبکه برمی‌گیریم. این فعالیت را دومین فعالیت فهرست قرار می‌دهیم و از شبکه خارج می‌کنیم و تا آخر ادامه می‌دهیم. فهرست حاصل یک فهرست راسته هم‌بندی است که زیرنامه‌های استاندارد برای این امر وجود دارد.

فرض می‌کنیم که شبکه دارای N فعالیت و M گره است، فعالیت‌ها در لیست به صورت راسته هم‌بندی با برچسب $1, 2, \dots, N$ قرار دارند، (شامل فعالیت‌های مجازی) و گره‌ها دارای برچسب $1, 2, \dots, M$ می‌باشند. گره k منبع و گره M گره نفوذ است. هر فعالیت با گره شروع و گره پایانی و زمان انجامش مشخص می‌شود. $S(K)$ یعنی گره شروع K امین فعالیت و $F(K)$ گره پایان K امین فعالیت و $T(K)$ زمان K امین فعالیت می‌باشد. به عنوان مثال شکل ۱۲-۳ این امر را نشان می‌دهد.

۱۲-۳ محاسبه مسیر بحرانی

یک دفعه که ارائه شبکه یک پروژه کامل شد، فعالیت‌ها بطور مقتضی برچسب می‌خورند و به فرم مقتضی (باب دارهای (K) ، $S(K)$ ، $F(K)$ ، $T(K)$) شرح داده می‌شوند، آماده‌ایم که کل زمان انجام پروژه و مسیر بحرانی را تعیین کنیم. برای این هدف، اول زمان‌های قدیمی‌ترین امکان شروع و تکمیل هر فعالیت مشخص می‌شود. به وضوح K امین فعالیت نمی‌تواند شروع شود تا تمامی فعالیت‌های قطع شده در گره $S(K)$ (شروع گره برای K) کامل شوند.



شکل ۴-۱۲ فلوجارت پیشگرای

قدیمی‌ترین زمان انجام هر کدام از گره‌ها می‌باشد. ما از گره ۱ (شروع) شروع می‌کنیم. و زمان انجامش را به صفر مرتب می‌کنیم.

$$ENT(1) = 0$$

لذا فعالیت ۱ در گره ۱ شروع می‌شود.

$$\begin{aligned} EST(1) &= 0 \\ EFT(1) &= 0 + T(1) \end{aligned}$$

بطور مشابه، تمامی فعالیت‌های گره ۱ به صورت FET محاسبه می‌شود. حال معادله (۴.۲) ما را قادر می‌سازد که ENT گره بعدی را محاسبه کنیم و سپس محاسبه EST و EFT فعالیت‌ها و مانند آن، تا تمامی فعالیت‌های پروژه بیان شود. نگارش ساده است زیرا فعالیت‌ها به صورت راسته هم‌بندی شرح داده شده‌اند. هیچ فعالیتی انجام نمی‌گیرد مگر آنکه تمامی پیش‌آمدهای بلافاصله آن انجام شده باشند.

شکل ۴-۱۲ فلوجارت پیشگرای را نشان می‌دهد.

با شرح فلوجارت اجازه دهید با فرض شبکه شکل ۴-۱۲ پیشگرای را بسازیم. $M=5$ و $N=7$ و $S(K)$ و $T(K)$ و $F(K)$ از شکل خوانده می‌شوند. ENT تمامی گره‌ها صفر و $K=1$ می‌شود.

در ابتدای اولین اجرای حلقه:

$$\begin{aligned} SNode &= S(1) = 1 \\ PNode &= F(1) = 2 \\ EST(1) &= ENT(1) = 0 \\ EFT(1) &= EST(1) + T(1) = 0 + 5.1 = 5.1 \\ EFT(1) &= 5.1 > ENT(2) = 0 \\ ENT(2) &= 5.1 \\ K &= 2 \end{aligned}$$

از اینرو:

چون $2 \leq 7$ حلقه تکرار می‌شود:

$$\begin{aligned} SNode &= S(2) = 1 \\ PNode &= F(2) = 3 \\ EST(2) &= ENT(1) = 0 \\ EFT(2) &= 0 + T(2) = 7.2 \\ EFT(2) &= 7.2 > ENT(3) = 0 \\ ENT(3) &= 7.2 \\ K &= 3 \end{aligned}$$

از اینرو:

باز چون $3 \leq 7$ حلقه تکرار می‌شود:

$$\begin{aligned} SNode &= S(3) = 2 \\ PNode &= F(3) = 4 \\ EST(3) &= ENT(2) = 5.1 \\ EFT(3) &= EST(3) + T(3) = 5.1 + 6.0 = 11.1 \end{aligned}$$

و مانند آن.

(۱۲.۴) LST را محاسبه می کنیم. حال حداقل یک گره داریم (دیگری از M) که LST تمامی فعالیتها از شروع و محاسبه شده است. با معادله (۱۲.۵)، LNT(M) را محاسبه می کنیم. از این زمان LFT تمامی فعالیتها که در x پایان می پذیرد را می یابیم. بنابراین پسگرائی شبکه را از طریق معادلات (۱۲.۶)، (۱۲.۴) و ادامه می دهیم تا به گره ۱ برسیم، LNT را برای تمامی گره ها و LFT و LST را برای تمام فعالیتهای شبکه می یابیم. به عنوان نمونه این روش برای شکل ۱۲-۳ عبارتست از:

$$\begin{aligned} LNT(5) &= 25.4 \text{ days} \\ LFT(7) &= LFT(5) = 25.4 \text{ days} \\ LST(7) &= 25.4 - 2.5 = 22.9 \text{ days} \\ LST(5) &= 25.4 - 15.8 = 9.6 \text{ days} \\ LNT(4) &= 22.9 \text{ days} \\ LFT(6) &= LFT(3) = 22.9 \text{ days} \\ LST(6) &= 22.9 - 0.0 = 22.9 \text{ days} \\ LST(3) &= 22.9 - 6.0 = 16.9 \text{ days} \\ LNT(3) &= \text{Min} \{22.9, 9.6\} = 9.6 \text{ days} \\ LFT(4) &= LFT(2) = 9.6 \text{ days} \\ LST(4) &= 9.6 - 4.5 = 5.1 \text{ days} \\ LST(2) &= 9.6 - 7.2 = 2.4 \text{ days} \\ LNT(2) &= \text{Min} \{16.9, 5.1\} = 5.1 \text{ days} \\ LFT(1) &= 5.1 \text{ days} \\ LST(1) &= 5.1 - 5.1 = 0.0 \text{ days} \\ LNT(1) &= \text{min} \{0.0, 2.4\} = 0.0 \text{ days} \end{aligned}$$

بنابراین پسگرائی جهت هر گره از LNT(۱) و LST(۱) برای هر فعالیت زدر شبکه بدست می آید. پسگرائی به طریق دیگر از LNT(۱)، EST(۱) و EFT(۱) حاصل می شود.

بطبی: اختلاف بین آخرین زمانی که یک گره انجام می شود (بدون تأخیر) و دیرترین زمانی که امکان انجام این گره وجود دارد، ارائه آزادی عمل در گره است. (LNT(۱)-ENT(۱)) مقدار زمانی است که گره در پروژه می تواند انجام شود. این اختلاف "بطبی (کندی)" در گره نام دارد. به وضوح برای یک گره که در مسیر بحرانی می افتد مقدار این کندی صفر است. بطور مشابه برای فعالیت ز مقدار در مسیر بحرانی LST(۱)-EST(۱) آزادی عمل را فراهم می کند. بنابراین با یک پسگرائی در شبکه به وسیله بررسی تساوی EST و LST مسیر بحرانی تعیین می شود.

پیاده سازی پسگرائی
مانند وضعیت پیشگرائی، پسگرائی نیز می تواند انجام گیرد. فلوجارت شکل ۱۲-۵ یک پسگرائی و تعیین فعالیتهای بحرانی را نشان می دهد.

وقتی $K=8$ است مقادیر بردارها این چنین است:

$$\begin{aligned} EST &= (0, 0, 5.1, 5.1, 9.6, 9.6, 11.1) \\ EFT &= (5.1, 7.2, 11.1, 9.6, 25.4, 9.6, 13.6) \\ ENT &= (0, 5.1, 9.6, 11.1, 25.4) \\ TMIN &= ENT(5) = 25.4 \end{aligned}$$

زمان کامل شدن پروژه:

ENT(M) قدیمی ترین زمان انجام گره نفوذ است که نیاز دارد تا کامل شود. این زمان مساوی طول مسیر بحرانی شبکه است. از نظر تئوری، تمامی مسیر پیمایش شده و مسیر بحرانی تعیین می شود. گر چه واقعاً این امر عملی نیست. تکنیک زیر برای یافتن یک مسیر بحرانی با پیمایش پسگرائی شبکه کارآتر می باشد. اگر یک فعالیت ز شبکه در گره S(۱) آغاز و در گره F(۱) قطع شود، حال باید قدیمی ترین زمان امکان ز بتوانند محاسبه و مقداردهی شود، که EST(۱) است و قدیمی ترین زمان EFT(۱) به وسیله فعالیت Z کامل می گردد. LFT(۱) قدیمی ترین زمان ز باید بدون تأخیر زمان کامل پروژه، پایان یابد. پس قدیمی ترین زمان LST از (بدون تأخیر در تکمیل پروژه) عبارتست از:

$$(12.4)$$

$$LST(1) = LFT(1) - T(1)$$

جایی که T(۱) زمان فعالیت ز است. قدیمی ترین زمانی که گره Z انجام می گیرد:

$$(12.5)$$

$$LNT(1) = \text{Min}(LST)$$

بطور مشابه پایان هر فعالیت که در گره Z قطع شود، می تواند تا زمان LNT(۱) تأخیر افتد به بیان دیگر:

$$(12.6)$$

$$LFT(1) = LNT(1)$$

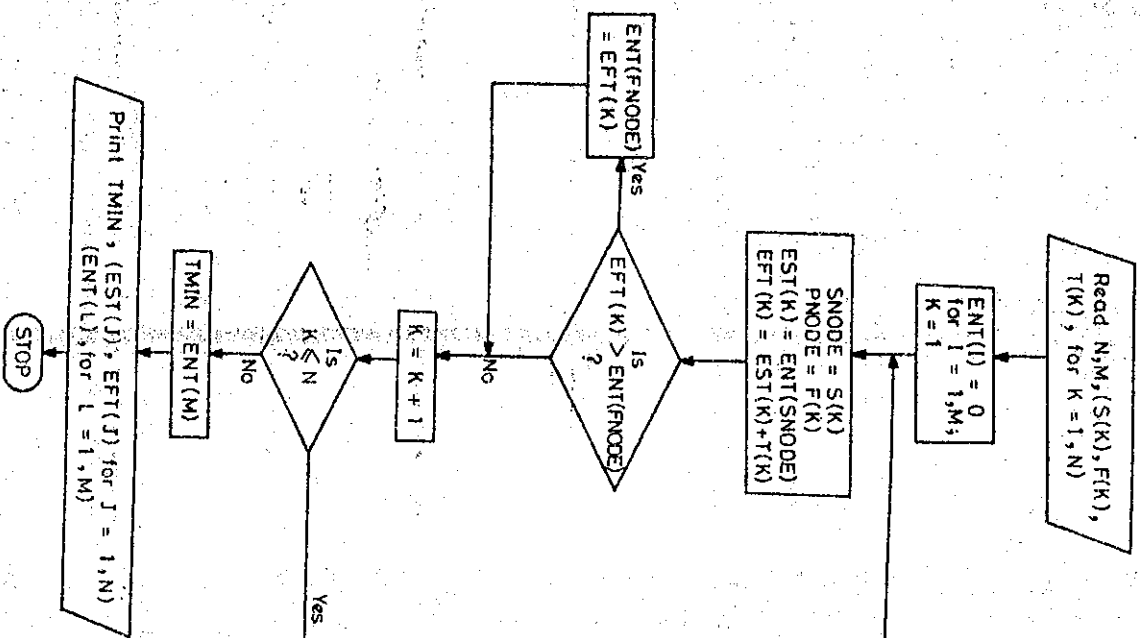
پسگرائی

حال از گره M (نفوذ) شروع می کنیم و زمان قدیمی ترین گره برابر زمان کامل شدن پروژه TMIN

(در انتهای پسگرائی محاسبه شده) قرار می دهیم، که عبارتست از:

$$LNT(M) = TMIN = ENT(M)$$

با معادله (۱۲.۶) این زمان را به LFT برای هر فعالیت انتهایی در M تنظیم می کنیم و با این LFT با معادله



شکل ۱۲-۲ فلورچارت پیشگرای

قدیمی‌ترین زمان انجام هر کدام از گره‌ها می‌باشد. ما از گره ۱ (منبع) شروع می‌کنیم. و زمان انجامش را به صفر مرتب می‌کنیم:

$$ENT(1) = 0$$

لذا فعالیت ۱ در گره ۱ شروع می‌شود:

$$\begin{aligned} EST(1) &= 0 \\ EFT(1) &= 0 + T(1) \end{aligned}$$

بطور مشابه، تمامی فعالیت‌های گره ۱ به صورت FET محاسبه می‌شود. حال معادله (۱۲.۲) ما را قادر می‌سازد که ENT گره بعدی را محاسبه کنیم و سپس محاسبه EFT و فعالیت‌ها مانند آن، تا تمامی فعالیت‌های پروژه بیان شود. نگرش ساده است زیرا فعالیت‌ها به صورت راسته هم‌بندی شرح داده شده‌اند. هیچ فعالیتی انجام نمی‌گیرد مگر آنکه تمامی پیش آمده‌های بلافاصله آن انجام شده باشند.

شکل ۱۲-۲ فلورچارت پیشگرای را نشان می‌دهد.

با شرح فلورچارت اجازه دهید با فرض شبکه شکل ۱۲-۳ پیشگرای را بسازیم. $M=5$ و $N=7$ و $S(K)$ و $T(K)$ و $F(K)$ از شکل خوانده می‌شوند. ENT تمامی گره‌ها صفر و $K=1$ می‌شود.

در ابتدای اولین اجرای حلقه:

$$\begin{aligned} S(NODE) &= S(1) = 1 \\ F(NODE) &= F(1) = 2 \\ EST(1) &= ENT(1) = 0 \\ EFT(1) &= EST(1) + T(1) = 0 + 5.1 = 5.1 \\ ENT(1) &= 5.1 > ENT(2) = 0 \\ K &= 2 \end{aligned}$$

از اینرو:

چون $2 \leq 7$ حلقه تکرار می‌شود:

$$\begin{aligned} S(NODE) &= S(2) = 1 \\ F(NODE) &= F(2) = 3 \\ EST(2) &= ENT(1) = 0 \\ EFT(2) &= 0 + T(2) = 7.2 \\ EFT(2) &= 7.2 > ENT(3) = 0 \\ ENT(3) &= 7.2 \end{aligned}$$

از اینرو:

باز چون $3 \leq 7$ حلقه تکرار می‌شود:

$$\begin{aligned} S(NODE) &= S(3) = 2 \\ F(NODE) &= F(3) = 4 \\ EST(3) &= ENT(2) = 5.1 \\ EFT(3) &= EST(3) + T(3) = 5.1 + 6.0 = 11.1 \end{aligned}$$

و مانند آن.

(۱۲.۴) LST را محاسبه می‌کنیم. حال حداقل یک گره داریم (دیگری از M) که α LST تمامی فعالیتها از شروع و محاسبه شده است. با معادله (۱۲.۵)، LNT(M) را محاسبه می‌کنیم. از این زمان LFT تمامی فعالیتها که در x پایان می‌پذیرد را می‌یابیم. بنابراین پسگرائی شبکه را از طریق معادلات (۱۲.۶)، (۱۲.۴) و ادامه می‌دهیم تا به گره ۱ برسیم، LNT را برای تمامی گره‌ها و LFT را برای تمام فعالیتهای شبکه می‌یابیم. به عنوان نمونه این روش برای شکل ۱۲-۳ عبارتست از:

$$\begin{aligned} LNT(5) &= 25.4 \text{ days} \\ LFT(5) &= LFT(5) = 25.4 \text{ days} \\ LFT(7) &= 25.4 - 2.5 = 22.9 \text{ days} \\ LST(7) &= 25.4 - 15.8 = 9.6 \text{ days} \\ LST(5) &= 25.4 - 15.8 = 9.6 \text{ days} \\ LNT(4) &= 22.9 \text{ days} \\ LFT(6) &= LFT(3) = 22.9 \text{ days} \\ LST(6) &= 22.9 - 0.0 = 22.9 \text{ days} \\ LST(3) &= 22.9 - 6.0 = 16.9 \text{ days} \\ LNT(3) &= \text{Min} \{22.9, 9.6\} = 9.6 \text{ days} \\ LFT(4) &= LFT(2) = 9.6 \text{ days} \\ LST(4) &= 9.6 - 4.5 = 5.1 \text{ days} \\ LST(2) &= 9.6 - 7.2 = 2.4 \text{ days} \\ LNT(2) &= \text{Min} \{16.9, 5.1\} = 5.1 \text{ days} \\ LFT(1) &= 5.1 \text{ days} \\ LST(1) &= 5.1 - 5.1 = 0.0 \text{ days} \\ LNT(1) &= \text{min} \{0.0, 2.4\} = 0.0 \text{ days} \end{aligned}$$

بنابراین پسگرائی جهت هر گره از LNT(0) و LST(0) برای هر فعالیت از شبکه بدست می‌آید. پسگرائی به طریق دیگر از LNT(0)، EST(0) و EFT(0) حاصل می‌شود.

بطبی: اختلاف بین آخرین زمانی که یک گره انجام می‌شود (بدون تأخیر) و دیرترین زمانی که امکان انجام این گره وجود دارد، ارائه آزادی عمل در گره است. LNT(0)-ENT(0) مقدار زمانی است که گره در پروژه می‌تواند انجام شود. این اختلاف "بطبی (کندی)" در گره نام دارد. به وضوح برای یک گره که در مسیر بحرانی می‌افتد مقدار این کندی صفر است. بطور مشابه برای فعالیت از مقدار LFT(0)-EFT(0) آزادی عمل را فراهم می‌کند. بنابراین با یک پسگرائی در شبکه به وسیله بررسی تساوی EST و LST مسیر بحرانی تعیین می‌شود.

پیاده‌سازی پسگرائی
مانند وضعیت پیشگرائی، پسگرائی نیز می‌تواند انجام گیرد. فلوچارت شکل ۱۲-۵ یک پسگرائی و تعیین فعالیتهای بحرانی را نشان می‌دهد.

وقتی $K=8$ است مقادیر بردارها این چنین است:

$$\begin{aligned} EST &= (0, 0, 5.1, 5.1, 9.6, 9.6, 11.1) \\ EFT &= (5.1, 7.2, 11.1, 9.6, 25.4, 9.6, 13.6) \\ ENT &= (0, 5.1, 9.6, 11.1, 25.4) \\ TMIN &= ENT(5) = 25.4 \end{aligned}$$

زمان کامل شدن پروژه:

ENT(M) قدیمی‌ترین زمان انجام گره نفوذ است که نیاز دارد تا کامل شود. این زمان مساوی طول مسیر بحرانی شبکه است. از نظر تئوری، تمامی مسیر پیمایش شده و مسیر بحرانی تعیین می‌شود. گرچه واقعاً این امر عملی نیست. تکنیک زیر برای یافتن یک مسیر بحرانی با پیمایش پسگرائی شبکه کارآتر می‌باشد. اگر یک فعالیت از شبکه در گره S(0) آغاز و در گره F(0) قطع شود، حال باید قدیمی‌ترین زمان امکان زیوانند محاسبه و مقداردهی شود، که EST(0) است و قدیمی‌ترین زمان EFT(0) به وسیله فعالیت از کامل می‌گردد. LFT(0) قدیمی‌ترین زمان زیانید بدون تأخیر زمان کامل پروژه، پایان یابد. پس قدیمی‌ترین زمان LST از (بدون تأخیر در تکمیل پروژه) عبارتست از:

$$(12.4)$$

$$LST(0) = LFT(0) - T(0)$$

جایی که T(0) زمان فعالیت است. قدیمی‌ترین زمانی که گره انجام می‌گیرد:

$$(12.5)$$

$$LNT(i) = \text{Min}(LST)$$

{(تمامی فعالیتهایی که در شروع می‌شود) (LST به تأخیر افتد به بیان دیگر:

$$(12.6)$$

$$LNT(i) = LFT(i) \text{ (هر فعالیتی که در قطع می‌شود)}$$

پسگرائی

حال از گره M (نفوذ) شروع می‌کنیم و زمان قدیمی‌ترین گره برابر زمان کامل شدن پروژه TMIN (در افتای پسگرائی محاسبه شده) قرار می‌دهیم، که عبارتست از:

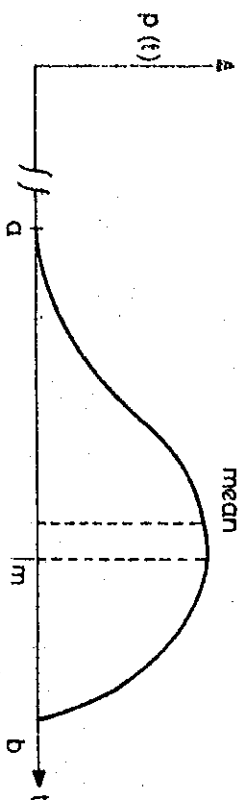
$$LNT(M) = TMIN = ENT(M)$$

با معادله (۱۲.۶) این زمان را به LFT برای هر فعالیت انتهایی در M تنظیم می‌کنیم و با این LFT با معادله

۱۲.۴- تولید در زمان فعالیت

گرچه فرض کرده‌ایم که هر فعالیت مقدار زمانی ثابتی برای کامل شدن نیاز دارد ولی در بسیاری از پروژه‌ها ممکن است تردیدی در دوره فعالیت‌های مختلف وجود داشته باشد. مستویاً برای پروژه‌هایی که شامل فعالیت‌های تحقیقاتی و توسعه‌ای است درست می‌باشد. در این صورت نیاز به نمونه‌های تصادفی برای این فعالیت‌ها است. لذا در این موارد با تکرار نمونه‌ها، مسیرهای بحرانی دارای زمانهای مختلف و غیر یکسان می‌باشد.

در این وضعیت به دنبال یافتن بهترین تقریب توزیع فرمان برای زمان فعالیت‌ها هستیم. در روال واقعی PERT (در ارتش و آمریکا برای تحقیقات و توسعه در سال ۱۹۵۷) فرض شد که این دوره‌ها زمان با توزیع تا داده شود. برای هر فعالیت به یک بیش‌ترین شخصی قوی با تعریف خوش‌بینانه (a) تقریب بدبینانه (b) و بهترین تقریب مانند (m) است. شکل ۱۲-۶ این امر را نشان می‌دهد.



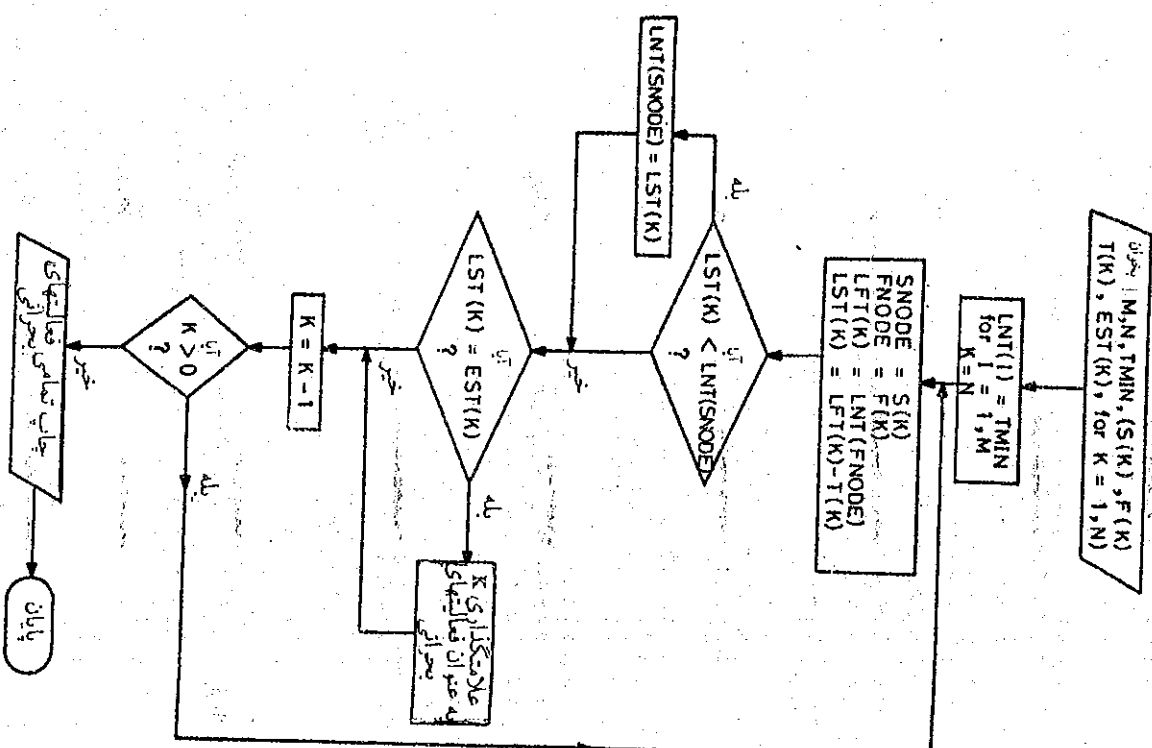
شکل ۶-۱۲ توزیع بتا

با این سه پارامتر میانگین و واریانس عبارتست از:

$$T_M(U) = \frac{a(U) + 4m(U) + b(U)}{6}$$

$$VAR(U) = \left(\frac{b(U) - a(U)}{6} \right)^2$$

$$J = 1, 2, \dots, N.$$



شکل ۵-۱۲ فلورچارت پیگردانی و مسیر بحرانی

۱۲.۵- شبکه سازی یک شبکه فعالیت

- (۱) یک فهرست راسته هم‌بندی فعالیتها $1, 2, \dots, h$ را ارائه می‌دهیم.
 (۲) برای هر فعالیت i به گره شروع $S(i)$ و گره پایان $F(i)$ N می‌باشد.
 (۳) توزیع احتمال را پادامترهای مقضی برای زمان هر فعالیت یک خروجی دارد.

برای شبیه‌سازی این وضعیت تصادفی، یک نمونه برای هر زمان هر فعالیت از توزیع احتمالی مربوطه به صورت تصادفی تولید می‌کنیم. این نمونه‌ها به وسیله $(TS(i), N, i=2, \dots, N)$ نگهداری می‌شود. براساس این زمانها، زمان کامل شدن پروژه TMIN تعیین و فعالیت‌های بحرانی مشخص می‌گردند. با نمونه‌های دیگر این عمل را بارها تکرار می‌کنیم و سوانیق را ثبت می‌نماییم. به وسیله تحلیل‌های موردنظر سعی می‌کنیم توزیع نرمالی بیابیم و بهترین توزیع‌ها را داشته باشیم.

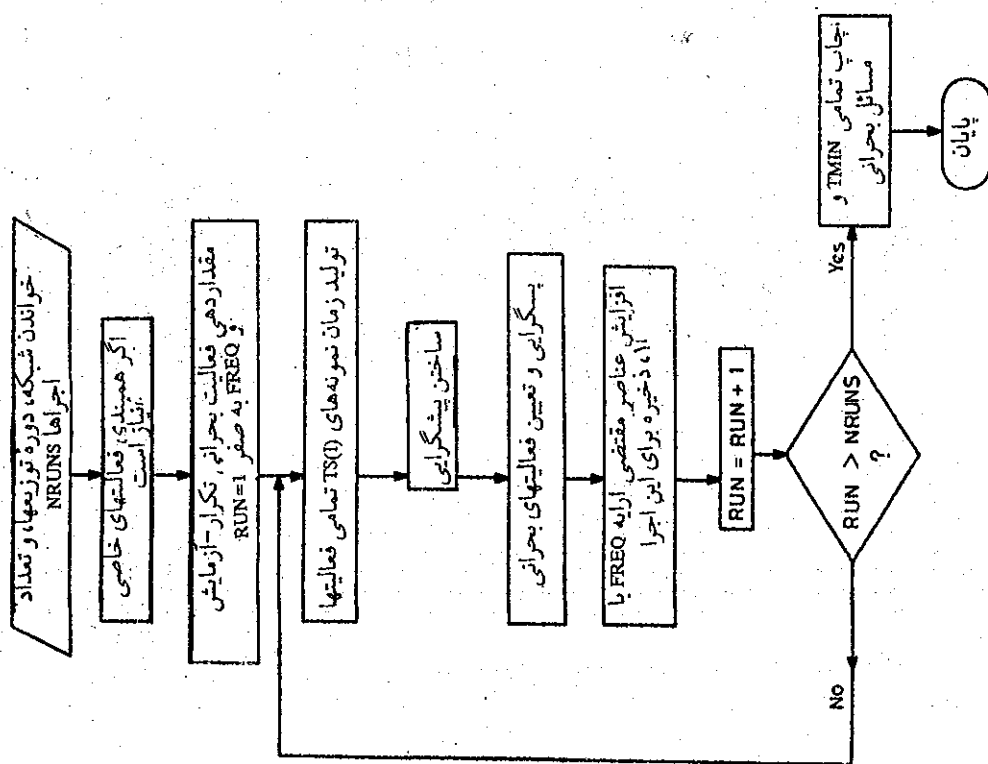
உதயசுந்தரி

یافتهن فعالیتهای بحرانی با داده‌های شبیه‌سازی شده اطلاعات، مفید است. با ثبت تعداد دفعاتی که یک فعالیت بحرانی می‌شود، می‌توان برای یک فعالیت بحرانی درصد تکرار در شبیه‌سازی را یافت. این "شاخص بحرانی" برای آن فعالیت خاص است. برای مثال، اگر یک فعالیت ۱۵۰۰ بار از آن ۱۰۰۰۰ بار تکرار شبیه‌سازی در یک مسیر بحرانی قرار دارد، درصد شاخص بحرانی ۱۵٪ است. اگر میانگین زمان استفاده شود، فعالیت ۱۵٪ بحرانی می‌شود و باعث دقت توجه به فعالیتهای شبیه‌سازی به ما می‌گردد که فعالیت ۱۵٪ احتمال ۱۵٪ بحرانی می‌شود و باعث توجه به فعالیتهای است. این امر تعیین درجه بحرانی بودن فعالیتهاست که به ما کمک زیادی می‌کنند.

روال کاملاً شبیه‌سازی یک شبکه فعالیت در فلوجارت شکل ۷-۱۲ نشان داده شده است.

مثال ۲-۲

شکل ۱۲-۸ یک شبکه از ۲۵ فعالیت و ۱۳ گروه را نشان می دهد. جدول ۱-۱۲ فعالیتها و زمان هر فعالیت را نشان می دهد. میانگین زمان فعالیت MV و انحراف معیار آن STGMA در جدول آمده است. فعالیتها نیز به صورت راسته هم بندی حاضرند.



در جدول ۱۲-۲ و ۱۲-۳ شاخص های تکرار محدوده مورد نظر خروجی نشان داده شده است.

شماره فعالیت

شاخص بحرانی

| | |
|----|-------|
| 1 | 0.000 |
| 2 | 0.176 |
| 3 | 0.000 |
| 4 | 0.176 |
| 5 | 0.824 |
| 6 | 0.688 |
| 7 | 0.118 |
| 8 | 0.118 |
| 9 | 0.000 |
| 10 | 0.018 |
| 11 | 0.000 |
| 12 | 0.864 |
| 13 | 0.000 |
| 14 | 0.000 |
| 15 | 0.000 |
| 16 | 0.000 |
| 17 | 0.000 |
| 18 | 0.074 |
| 19 | 0.927 |
| 20 | 0.000 |
| 21 | 0.000 |
| 22 | 0.000 |
| 23 | 0.927 |
| 24 | 1.000 |
| 25 | 0.000 |

جدول ۱۲-۲ شاخص بحرانی فعالیت مختلف

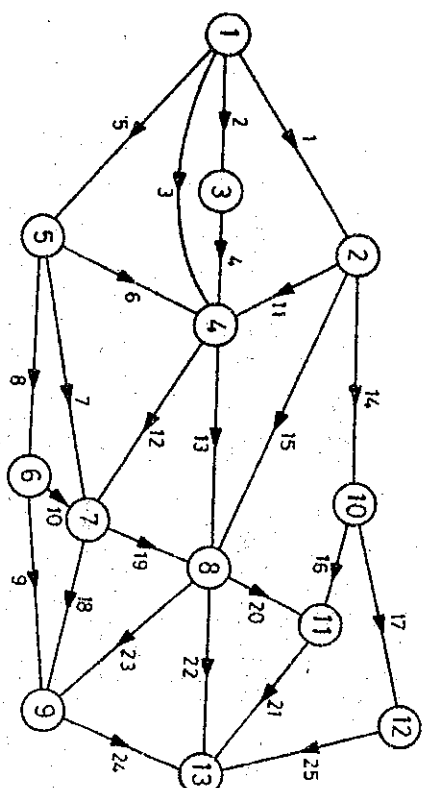
I

TMIN

Count (I)

| | | |
|----|---------|-----|
| 1 | 38 - 40 | 0 |
| 2 | 40 - 42 | 0 |
| 3 | 42 - 44 | 1 |
| 4 | 44 - 46 | 2 |
| 5 | 46 - 48 | 8 |
| 6 | 48 - 50 | 9 |
| 7 | 50 - 52 | 48 |
| 8 | 52 - 54 | 64 |
| 9 | 54 - 56 | 120 |
| 10 | 56 - 58 | 135 |
| 11 | 58 - 60 | 168 |
| 12 | 60 - 62 | 147 |
| 13 | 62 - 64 | 130 |
| 14 | 64 - 66 | 80 |
| 15 | 66 - 68 | 58 |
| 16 | 68 - 70 | 13 |
| 17 | 70 - 72 | 12 |
| 18 | 72 - 74 | 4 |
| 19 | 74 - 76 | 1 |
| 20 | 76 - 78 | 0 |

جدول ۱۲-۳ داده خروجی برای هسته گرام

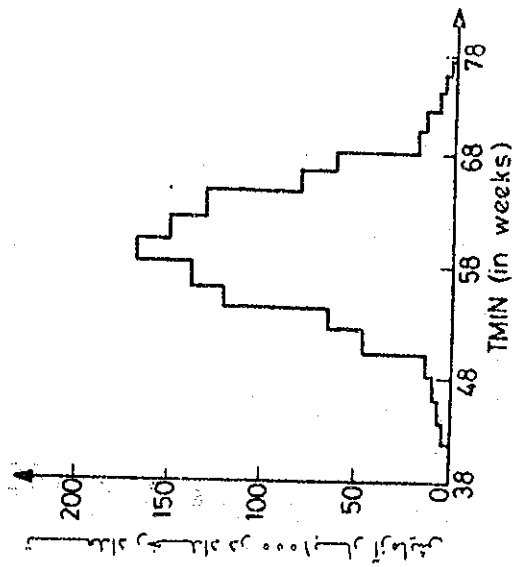


شکل ۱۲-۸ شبکه فعالیت

| شماره فعالیت | کد | تکرار | AMU | STGK4 |
|--------------|-------|-------|-------|-------|
| K | ST(K) | TR(K) | | |
| 1 | 1 | 2 | 5.00 | 1.17 |
| 2 | 1 | 3 | 10.00 | 2.53 |
| 3 | 1 | 4 | 6.00 | 1.24 |
| 4 | 3 | 4 | 9.00 | 1.99 |
| 5 | 1 | 5 | 8.00 | 1.52 |
| 6 | 5 | 4 | 15.00 | 3.36 |
| 7 | 5 | 7 | 19.00 | 3.68 |
| 8 | 5 | 6 | 6.00 | 1.05 |
| 9 | 6 | 9 | 8.00 | 0.98 |
| 10 | 6 | 7 | 11.00 | 2.39 |
| 11 | 2 | 4 | 7.00 | 0.05 |
| 12 | 4 | 7 | 9.00 | 0.32 |
| 13 | 4 | 8 | 3.00 | 0.67 |
| 14 | 2 | 10 | 2.00 | 0.43 |
| 15 | 2 | 8 | 8.00 | 1.64 |
| 16 | 10 | 11 | 6.00 | 0.09 |
| 17 | 10 | 12 | 1.00 | 0.20 |
| 18 | 7 | 9 | 10.00 | 1.09 |
| 19 | 7 | 8 | 3.00 | 0.70 |
| 20 | 8 | 11 | 1.00 | 0.25 |
| 21 | 11 | 13 | 4.00 | 0.09 |
| 22 | 8 | 13 | 5.00 | 0.11 |
| 23 | 8 | 9 | 9.00 | 0.53 |
| 24 | 9 | 13 | 14.00 | 3.67 |
| 25 | 12 | 13 | 10.00 | 2.08 |

جدول ۱۲-۱ فعالیتها

شکل ۹-۱۲ هیستوگرام ترسیمی به ازای ۱۰۰۰ تکرار را نشان می‌دهد. می‌بینیم که شکل ۹-۱۲ به اندازه کافی افزایش آرام ندارد و مسطح نیست. لذا، باید تغییراتی در زمانها داد که مثلاً باید تکرار را از ۱۰۰۰ به ۵۰۰۰ رساند.



شکل ۹-۱۲ توزیع فرکانسی زمان کامل شدن پروژه TMIN در ۱۰۰۰ بار آزمایش

فصل سیزدهم

طراحی و ارزیابی آزمایشات شبه‌سازی

در چهار فصل گذشته نشان دادیم که چگونه می‌توان روشی کامیوتری را برای شبه‌سازی یک سیستم گسسته تصادفی از قبیل صف و انبار ارائه داد و روشهای موردنظر را بر روی سیستم‌های بررسی شده اعمال کرد که موفقیت آمیز هم خواهد بود. به عنوان مثال در تعیین شرایط اولیه از قبیل طول اولیه صف یا موجودی اولیه انبار همچنین در تعیین مدت زمان اجرای روش آزادی در عمل داریم، مثلاً در ۶ ماه (یا ۱۸۰ روز) اجرا کنیم.

موضوع اصلی مورد بحث تعیین پارامترهای مذکور می‌باشد: طول زمان اجراء، شرایط اولیه، تعداد تکرارها و غیره. بنابراین می‌توان اطلاعات مفیدی را در مورد عملکرد سیستم با کمترین هزینه بدست آورد و راه بدست آوردن هر یک از آنها مشکل عملدهای در شبه‌سازی محسوب می‌شود. در حال حاضر روئین واحدی برای طراحی بهترین روش شبه‌سازی در دسترس نیست و موارد معدودی در اختیار است که در این بخش مورد بحث قرار می‌گیرد.

۱۳-۱- مدت زمان اجرای شبه‌سازی

اجرای شبه‌سازی، ثبت بی‌وقفه عکس‌العمل و رفتار سیستم تحت اجرای تعیین شده از متغیرهای قابل کنترل می‌باشد. یکی از سئوالات مهم در کلیه روشهای شبه‌سازی تصادفی، مدت زمانی لازم برای اجرای شبه‌سازی (پریود شبه‌سازی) است. در نتیجه درجه اطمینان بیشتری نسبت به نتایج عددی آزمایش به ما داده می‌شود و قبل از پاسخگویی به آن سئوال بهتر است سئوال مشابه با آن را در آمار کلاسیک بررسی کنیم. فرض کنید متغیر تصادفی x را با میانگین μ و انحراف معیار σ داریم. ما در مورد عملکرد چگالی آن به غیر از اینکه ثابت است اطلاع دیگری نداریم. (مثلاً چگالی آن با زمان تغییر نمی‌کند) فرض کنید n نمونه مستقل x_1, x_2, \dots, x_n را داریم حال میانگین نمونه‌هایی مستقل محاسبه می‌شود:

$$1.96 \times \frac{\sigma}{\sqrt{n}} = \frac{\sigma}{25}$$

$$n = (1.96 \times 25)^2 = 2,401$$

بنابراین اگر ما ۲۴۰۱ نمونه مستقل از متغیر تصادفی x را در نظر بگیریم و نمونه میانگین \bar{x} را محاسبه کنیم می‌توانیم اعلام کنیم که ۹۵٪ احتمال دارد که اختلاف بین \bar{x} و میانگین واقعی μ کمتر از $\frac{1}{25}$ انحراف معیار باشد.

رابطه کلی بین پارامترهای به صورت زیر می‌باشد:

$$Pr\{\mu - t \leq \bar{x} \leq \mu + t\} = 1 - \alpha \quad (13.1)$$

تلفرانس در هر یک از طرفین μ (میانگین) در واقع محلی است که انتظار داریم در تفریب \bar{x} قرار گیرد (با احتمال $1 - \alpha$). از آنجایی که تقریباً توزیع نرمال دارد، ارتباط آن به متغیر z توزیع نرمال استاندارد به صورت زیر بیان می‌گردد:

$$\begin{aligned} \bar{x} &= z \cdot \left(\text{انحراف معیار} \right) + \mu \\ &= z \cdot (\sigma/\sqrt{n}) + \mu \end{aligned}$$

$$Pr\{\mu - t \leq \bar{x} \leq \mu + t\} = Pr\{\mu - t \leq z \cdot (\sigma/\sqrt{n}) + \mu \leq \mu + t\}$$

سپس:

$$\begin{aligned} &= Pr\{-t \leq z(\sigma/\sqrt{n}) \leq t\} \\ &= Pr\left\{-\frac{t}{\sigma/\sqrt{n}} \leq z \leq \frac{t}{\sigma/\sqrt{n}}\right\} \quad (13.2) \end{aligned}$$

انتگرال تابع چگالی نرمال استاندارد z از $-\infty$ تا ∞ می‌شود:

$$\Phi(t) = \int_{-\infty}^t \frac{1}{\sqrt{2\pi}} e^{-z^2/2} dz = Pr\{z \leq t\}$$

جدول مقادیر انتگرالی فوق برای مقادیر مختلف z خیلی وسیع می‌باشد. از آنجائیکه توزیع نرمال تقریباً در حوالی میانگین است و کل ناحیه $\Phi(z)$ تحت منحنی یک است، بنابراین به راحتی می‌توان

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (13.3)$$

چگونه میانگین نمونه \bar{x} را به میانگین μ نزدیک کنیم؟ واضح است که هرچه تعداد نمونه‌های n بزرگتر باشد تفاوت بین \bar{x} و μ کمتر خواهد بود. به این معنا که هرچه n به سمت بی‌نهایت میل کند اختلاف بین \bar{x} و μ (به سمت صفر میل خواهد کرد. علاوه بر آن بر طبق قضیه حد مرکزی در آمار، میانگین نمونه \bar{x} یک متغیر تصادفی تقریباً با توزیع نرمال با میانگین μ و انحراف معیار σ/\sqrt{n} می‌شود (اگر n به اندازه کافی بزرگ باشد یعنی $n > 10$ برای بیشتر توزیع‌های x کافی است).

از آنجایی که انحراف معیار x معیاری برای اندازه‌گیری پراکندگی است (با همان واحدی که x دارد)، همانطور که مشاهده می‌کنید دقت در x دارای رابطه عکس با ریشه دوم n (تعداد نمونه‌ها) می‌باشد. با فرض n نمونه می‌توان محاسبه کرد که با چه احتمالی \bar{x} به μ نزدیک شود و برعکس برای یک حد تلفرانس تعیین شده (تفاوت بین \bar{x} و μ بر حسب مقدار انحراف معیار x) می‌توان تعداد نمونه‌های مورد نیاز را محاسبه کرد.

مثال ۱-۱۳

فرض کنید در یک نمونه‌برداری با احتمال ۹۵٪ میانگین نمونه‌ها \bar{x} داخل انحراف معیار $\sigma/25$ از میانگین واقعی μ باشد، می‌خواهیم n را بدست آوریم.

$$Pr\{\mu - (\sigma/25) \leq \bar{x} \leq \mu + (\sigma/25)\} = 0.95$$

تقریباً x دارای توزیع نرمال با میانگین μ و انحراف معیار σ/\sqrt{n} است. همچنین می‌دانیم که بخشی از ناحیه تحت تابع چگالی نرمال با انحراف معیار σ/\sqrt{n} نسبت به میانگین، نشان دهنده احتمال واقع شدن نمونه تصادفی در این محدوده ۹۵٪ میانگین می‌باشد. با مشاهده جدول ناحیه تحت منحنی تابع چگالی نرمال توجه می‌شویم که احتمال قرار گرفتن در این ناحیه ۹۵٪ است.

با احتمال $\Phi(2/5)$ تحت هر ناحیه انتهایی، انحراف معیار σ/\sqrt{n} از $\sigma/25$ روی هر گوشه میانگین μ را

باید داشته باشیم.

بنابراین:

$$\delta_{est} = \frac{1}{M-1} \sum_{i=1}^m (x_i - \bar{x})^2 \quad (13.7)$$

واریانس) متغیر تصادفی نرمال z می شود.

$$Z = \frac{\bar{x} - \mu}{\delta_{est}} \sqrt{m}$$

مطابق توزیع نرمال استاندارد، توزیع نمی شود در عوض توزیع z دانش آموز دارد.

اگر مقدار n به اندازه کافی زیاد باشد ($n > 50$) اختلاف بین ۲ توزیع قابل اغماض می باشد. خوشبختانه در اکثر آزمایشات شبیه سازی، طول زمان اجرا به منظور ایجاد چنین شرایطی مناسب می باشد.

طول اجرای شبیه سازی تصادفی ثابت:

ما می توانیم از معادله (۱۳.۶) برای تعیین طول اجرا (RUN) در یک شبیه سازی استفاده کنیم مشروط براینکه دو شرط زیر در معادله وجود داشته باشد: (۱) توزیع بدون تغییر است یعنی شبیه سازی به یک وضعیت ثابت رسیده قبل از اینکه ما \dots را مشاهده کرده باشیم و (۲) نمونه های x_1, x_2, \dots مستقل هستند. بعضی از شبیه سازی های سیستم های تصادفی اساساً ثابت می باشند جایی که زمان قانونمند نیست. به عنوان مثال شبیه سازی یک سیستم PERT به این رده تعلق دارد و حالت گذرا نداریم. جدا از آنر سلسله اعداد تصادفی، شبیه سازی اساساً در وضعیت ثابت ضروری است.

در یک وضعیت پایدار از شروع شبیه سازی، خروجی های شامل پیش آمدهای مستقل هر یک از قانون احتمالی مشابه پیروی می کنند. از معادله (۱۳.۶) برای تعیین طول یک اجرا (RUN) می توان استفاده نمود. مثالی از شبیه سازی تصادفی ثابت ارائه می شود تا ببینید چگونه طول اجزاء با استفاده از معادله (۱۳.۶) تعیین می شود.

مثال ۱۳-۳: بازی تاس تخته نرد را در نظر بگیرید که از شما خواسته شده شبیه سازی کنید. ۱۰۰ بازی را اجرا و چاپ کنید. برندگان گروه A را فرض کنید دارای خروجی به صورت زیر می باشد (از ۱۰۰ بازی)

تعداد دفعاتی که گروه A برنده شده = ۵۶

تعداد دفعاتی که گروه A برنده شده = ۴۴

آیا مدت زمان اجرا به اندازه کافی برای تعیین نتیجه با ضریب اطمینان بالا کافی است؟ اگر نه سپس چه مدت زمان لازم است تا میانگین برندگان هر بازی با دقت ± 500 و با احتمال 95% بدست آید؟ به منظور یافتن انحراف معیار باید از فرمول (۱۳.۶) برای بدست آوردن عدد n استفاده کرد زیرا μ معین هستند. در صورت فقدان اطلاعات فوق باید از روال زیر حداطمینان بالا و پایین بدست آورد (U و L):

رابطه زیر را بیان کرد:

$$Pr\{-y < z \leq y\} = 2Pr\{z \leq y\} - 1 \quad (13.4)$$

بنابراین با مساوی قرار دادن رابطه (۱۳.۳)، (۱۳.۴) و جایگزینی (۱۳.۲) در آن رابطه ۱۳.۵ بدست می آید:

$$2Pr\left\{z \leq \frac{1}{\sigma} \sqrt{n}\right\} - 1 = 1 - \alpha$$

(۱۳.۵)

$$Pr\left\{z < \frac{1}{\sigma} \sqrt{n}\right\} = 1 - \frac{\alpha}{2}$$

بنابراین برای مقدار مشخص α در جدول $\Phi(y)$ نگاه می کنیم و مقدار متناظر y را به این ترتیب که $\Phi(y) = 1 - \alpha/2$ پیدا می کنیم و عبارت فوق را با استفاده از $y = y_{1-\alpha/2}$ مشخص می کنیم. در نتیجه این مقدار با $\frac{1}{\sigma} \sqrt{n}$ برابر خواهد بود. که تعداد نمونه ها از این رابطه عبارتست از:

$$n = \frac{(y_{1-\alpha/2})^2 \sigma^2}{\epsilon^2} \quad (13.6)$$

۲. حد تراتس می باشد که انتظار آن را داریم یا مورد قبول ما است. ۲. واریانس و -- دو انتهای نرمال استاندارد با احتمال $(1-\alpha)$ است. سطح اطمینان $(1-\alpha)$ باید ۹۰ درصد برای $y_{1-\alpha/2} = 1/6.5$ باشد یا ۹۹٪ درصد برای $y_{1-\alpha/2} = 2/5.8$ است.

مثال ۱۳-۲: فرض کنید یک کارخانه لبنیاتی داریم که محصول شیر روزانه آن به طور تصادفی است. حال می خواهیم میانگین محصولات روزانه را با ± 40 لیتر از میانگین واقعی محصول با سطح اطمینان ۹۵٪ محاسبه کنیم. میانگین انحراف معیار محصولات روزانه تقریباً ۲۰۰ لیتر شده است. برای چند روز باید محصولات روزانه را از یک کارخانه لبنیاتی اندازه گیری کنیم؟ حال مقادیر زیر را در معادله (۱۳.۶) جایگزین کنید:

$$t = 40$$

$$\sigma = 200$$

$$y_{1-\alpha/2} = 1.96$$

$$n = \left(\frac{1.96 \times 200}{40} \right)^2 = 96 \text{ day}$$

این ترتیب سؤال مورد نظر را پاسخ می دهیم:

معادله (۱۳.۶) یک فرمول بسیار مفیدی برای محاسبه تعداد نمونه ها در آمار است. بدین منظور برای تعیین تعداد نمونه هایی که نیاز داریم باید واریانس (توزیع نمونه ها) را بدانیم که معمولاً شناخته شده نیست و می توان تخمین زد به صورت (۱۳.۷) اما وقتی از معادله ۱۳.۷ استفاده می شود (به جای

یکی از نکات مهمی که باید از جدول فوق دریافت این است که اساس بحث ما فقط به پایه اولین ۲۰۰ نمونه پایه‌ریزی می‌شود و اگر زمان اجرای (درواقع برای ۲۰۰ نمونه) شیشه‌سازی کافی نباشد نتیجه درستی را ارائه نخواهد داد. همچنین برنامه با یک طول اجرا به راحتی باید برای صدها هزار مورد نیز کافی باشد. تجربه و تحلیل بازی قمار با کمک نکات احتمالی قابل توجه مشکل نمی‌باشد. تمرین ۱۳-۱ نشان می‌دهد که مقدار واقعی قابل انتظار می‌باشد. با مقایسه مقدار تخمین زده شده μ در آخرین ردیف جدول ۱۳-۱ در می‌یابید که مقدار فوق در محدوده تعیین شده می‌باشد:

$$\mu = -0.01414 \\ = 0.01414 + 0.1578 = 0.00164$$

روال فوق که در مثال ۱۳-۳ استفاده شده در جدول ۱۳-۱ نشان داده شده است، می‌تواند اصلاح شود. بنابراین می‌توان مدت زمان اجرا را کمتر نمود.

برای رسیدن به محدوده خاص به جای افزایش n (همانطور که در جدول ۱۳-۱ نشان داده شده) می‌توانیم تقریب تعداد نمونه‌های مورد نیاز n^* را با استفاده از معادله (۱۳-۶) و $sest$ برای se به بستگی به اجرای اولیه مثلاً m مشاهده دارد را اندازه‌گیری کنیم. حال اگر $m \geq n^*$ باشد به نقطه دقت می‌رسیم و در نتیجه اجرا تمام می‌شود. اگر $m < n^*$ باشد باید $(n - m)$ مشاهده بیشتر را بدست آوریم و مجدداً میانگین نمونه x را محاسبه کرد.

واریانس est را با استفاده از معادله (۱۳-۷) بدست می‌آوریم. با جانشین کردن مقادیر جدید $sest$ و n در معادله (۱۳-۸) به محدوده جدیدی از فاصله (U-L) می‌رسیم. اگر محدوده این فاصله به اندازه کافی باریک باشد توقف می‌کنیم وگرنه مجدداً می‌توان روند فوق را به برای نمونه‌های مشابه تکرار کرد.

فلز چارت شکل ۱۳-۱ این روند را توضیح می‌دهد.

طول اجرای یک شیشه‌سازی تصادفی پویا: همانطوری که نشان داده شد (۱) اگر توزیع یکپارخت نباشد با (۲) اگر نمونه‌های y_1, y_2, \dots, y_n مستقل نباشند، معادله (۱۳-۶) برای تعیین اندازه نمونه‌ها مناسب نیست. در بیشتر شیشه‌سازی سیستم‌های تصادفی پویا هر دو مشکل فوق وجود دارد. خروجی‌ها وابسته به یکدیگرند و رفتارهای گذرا وجود دارد و اطلاعات ارتباط زیادی به آنها دارد. به عنوان مثال در سیستم صف همانند سیستم انبار مشکلات بیشتر جلوه می‌کند. حال بهتر است که سیستم چگونه این حالات گذرا برطرف می‌شود.

حذف حالات گذرا: حذف مشکلات به علت روشهای مختلفی برای از بین بردن اثرات ناشی از آنها وجود دارد که رایج‌ترین آن چشم‌پوشی کردن از مقدار اولیه اجرای شیشه‌سازی می‌باشد. اجرا از یک مرحله‌ای شروع شده و بعد از پرورد زمانی مطمئن خامه می‌یابد زمانی که سیستم به یک وضعیت ثبات رسیده. وضعیت سیستم در آن زمان بدون تغییر می‌باشد پس اجراء مجدد شروع شده و آماری از دومین اجرا انجام می‌شود.

$$U = \bar{x} + y_1 - se \frac{\sigma_{est}}{\sqrt{n}} \quad (13.8)$$

$$L = \bar{x} - y_1 - se \frac{\sigma_{est}}{\sqrt{n}}$$

برای m اجرا انحراف معیار تقریبی $sest$ بر سبیل معادله (۱۳-۷) ارائه شده. به عبارتی U و L را به عنوان فاصله‌ای در حدود میانگین تخمین زده شده تعیین کرده‌ایم. \bar{x} در فاصله (U-L) شامل میانگین واقعی μ با سطح اطمینان $(1-\alpha)$ خواهد بود.

حال همین روش شیشه‌سازی را برای ۱۰۰۰ بازی انجام می‌دهیم تا اینکه مقادیر عددی برای x و $sest$ از ۱۰۰۰ مشاهده بدست آید (با استفاده از معادله (۱۳-۱) و سپس U و L را با استفاده از معادله (۱۳-۸) محاسبه می‌کنیم. اگر L و U درحد تعیین شده قرار گیرد، مثلاً $2X0.005 \leq (U-L)$ در نتیجه در می‌یابیم، فاصله‌ای که توسط U و L تعیین می‌شود شامل میانگین واقعی، $(1-\alpha)$ درصد از زمان می‌باشد و همچنین نشان می‌دهد که x بهترین تقریب میانگین واقعی است. اگر L و U درحد تعیین شده قرار نگیرند می‌توان شیشه‌سازی را مثلاً برای ۱۰۰۰ مورد دیگر انجام داد و سری مقادیر $sest$ و U و L را بدست آورد و سپس با موارد قبلی مقایسه نمود. این روند مجدداً برای نمونه‌های دیگر تکرار می‌شوند تا اینکه U و L در محدوده تعیین شده قرار گیرد در این صورت شیشه‌سازی به پایان می‌رسد. محاسبات فوق در یک شیشه‌سازی می‌تواند برنامه‌ریزی شود و در نتیجه کامپیوتر به طور خودکار شیشه‌سازی را بعد از اینکه بازی به تعداد مناسب و کافی انجام شد تمام می‌کند. جدول (۱۳-۱) خلاصه‌ای از محاسبات فوق را نشان می‌دهد.

| n | \bar{x} | $sest$ | U | L | $U-L$ |
|---------|-----------|---------|-----------|----------|--------|
| 1,000 | + 0.05000 | 0.99925 | + 1.1193 | - 0.1193 | 1.2386 |
| 2,000 | + 0.02500 | 0.99994 | + 0.6882 | - 0.1882 | 0.8764 |
| 4,000 | - 0.06500 | 1.00010 | + 0.2449 | - 0.3749 | 0.6198 |
| 8,000 | - 0.16200 | 0.99933 | + 0.0566 | - 0.3816 | 0.4382 |
| 16,000 | - 0.08000 | 1.00000 | + 0.0750 | - 0.2350 | 0.3100 |
| 32,000 | - 0.04000 | 1.00001 | + 0.0652 | - 0.1539 | 0.2191 |
| 64,000 | - 0.00528 | 0.99999 | + 0.0247 | - 0.1503 | 0.1550 |
| 128,000 | - 0.01498 | 0.99989 | - 0.0051 | - 0.2046 | 0.1996 |
| 154,000 | - 0.01578 | 0.99988 | - 0.01079 | - 0.2077 | 0.0998 |

جدول ۱۳-۱ محاسبات با اطمینان ۹۵/۰۰۵ و $(1-\alpha) = 0.05$

مستلزم داشتن اطلاعات قبلی در مورد وضعیت یکنواخت سیستم (ثابت) می‌باشد که ممکن است همیشه در دسترس نباشد. سومین راه اطمینان موضوع آن است که اجراء (RUN) به اندازه کافی طولانی باشد تا بتوان از مسائل مقاداردهی اولیه صرف‌نظر کرد.

مشاهدات همبسته: دومین فرضی که با استفاده از معادله (۱۳.۶) در نظر گرفته می‌شود این است که n مشاهده یا نمونه‌های x_1, x_2, \dots, x_n نظر آماری مستقل هستند. این وضعیت در شبیه‌سازی ثابت غالباً رضایت بخش می‌باشد. اما برای شبیه‌سازی پویا این طور نیست. بهتر است این موضوع را با ارائه مثالی از زمان انتظار در یک سیستم صف روشن گردد:

یک سیستم و صف یک سرویس‌دهنده را با الگوهای ورودی و سرویس مشخص در نظر بگیرید. اگر بخواهیم شبیه‌سازی را به منظور تعیین میانگین زمان انتظار انجام بدهیم، با جمع زمان انتظار n مشتری و تقسیم آن به n (تعداد مشتری) می‌توان میانگین زمان انتظار را محاسبه نمود (در یک اجراء با طول n)

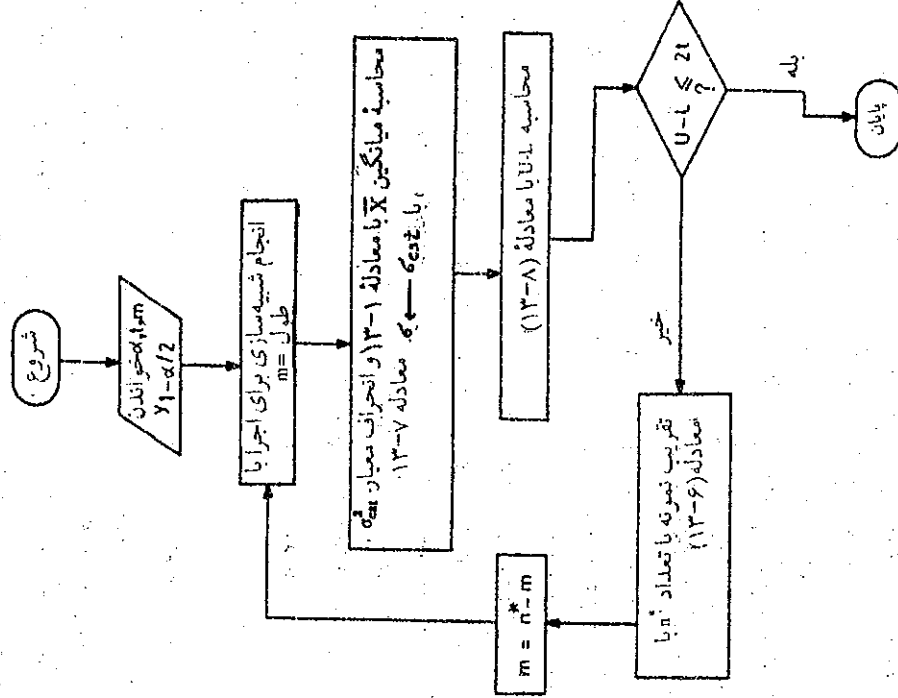
$$\bar{w} = \frac{1}{n} \sum_{i=1}^n w_i \quad (13.9)$$

واضح است که زمان انتظار مشتری‌ها (w_i) مستقل نیستند و در یک لیست انتظار، زمان انتظار K امین مشتری بستگی به زمان انتظار مشتری جلوتر دارد. اگر w_k زیاد باشد w_{k+1} طبیعتاً مقدار بیشتری خواهد داشت. همزمانی اطلاعات بر روی معیارهای اصلی اثر می‌گذارد که به آن همبستگی سریال یا همبستگی گویند. در یک سیستم صف همانطور که عوامل مورد استفاده افزایش می‌یابد (مثلاً افزایش طول میانگین صف) درجه همبستگی نیز افزایش می‌یابد.

همچنین زمانی که از نظر آماری دو نمونه به خوبی وابسته هستند، اطلاعات کمتری را در مورد تعداد اصلی نمونه می‌دهد تا زمانی که مستقل باشند. بنابراین برای بدست آوردن سطح اطمینان بیشتر، نمونه‌های بیشتری را باید در نظر گرفت. قبل از اینکه نمونه‌های مورد نیاز را در یک چنین مواردی جمع کنیم می‌بایست در ابتدا درجه همبستگی داده‌ها را تعیین کنیم.

در یک توالی از مشاهدات x_1, x_2, \dots, x_n مقادیر جدا شده m واحدی بر روی یکدیگر اثر می‌گذارند که می‌توان میزان این اثر را با استفاده از معادله (۱۳.۱۰) اندازه گرفت.

$$r_m = \frac{1}{n-m} \sum_{i=1}^{n-m} (x_i - \bar{x})(x_{i+m} - \bar{x}) \quad (13.10)$$



شکل ۱۳-۱ یافتن دقت آماری مشخص

در عمل به راحتی می‌توان نتایج آمارگیری در یک زمان مشخص از شروع مرحله بدست آمده را حذف کرد. البته برای حذف پروت "قوانین ساده‌ای در دسترس نیست. راه دیگر برای کاهش اثر این مشکل گذرا در آمارها، آغاز سیستم در یک وضعیت ابتدایی نزدیک به حالت یکنواخت (ثابت) می‌باشد. از آنجایی که حالت گذرا به علت تفاوت بین وضعیت یکنواخت و وضعیت ابتدایی اولیه می‌باشد بنابراین هرچه این تفاوت کمتر باشد حالت گذرا کوتاه‌تر خواهد شد. استفاده از روش فوق

و برای حالتی که وابستگی دارند:

$$n = \frac{\sum_{k=1}^M \left(1 + 2 \left(1 - \frac{k}{M+1}\right) r_k\right)}{r^2} \quad (13.12)$$

در جایی که ضریب را همبستگی r_k ها به وسیله معادله (13.10) بدست می‌آید و با استفاده از فرمول (13.12) به جای (13.6) می‌توان اندازه نمونه‌های مورد نیاز در یک وضعیت همبسته را بدست آورد.

مثال 13.4

یک سلسله از 500 مشاهده با یکدیگرند همبستگی سریال دارند. ضریب همبستگی آنها با

$$r_1 = 0.33$$

استفاده از معادله (13.10) بدین ترتیب محاسبه شده است:

$$r_2 = 0.25 \quad r_3 = 0.15$$

و موارد دیگر تفاوت چندانی با صفر ندارند (یعنی صفر هستند) و میانگین آنها $0.5/20$ و واریانس آن $0.9/20$ بدست آمده است. کوچکترین اندازه نمونه را با تقریب ± 2 واحد و سطح اطمینان $0.95/1 - \alpha$ با استفاده از معادله (13.12) بدست آورید:

$$n = \frac{(1.96)^2 \cdot (0.20)}{(2)^2} \left\{ 1 + 2 \left[\left(1 - \frac{1}{4}\right) (0.33) + \left(1 - \frac{2}{4}\right) (0.25) + \left(1 - \frac{3}{4}\right) (0.15) \right] \right\} \\ = 979.61 \approx 1783$$

بنابراین همبستگی نمونه‌ها از $0.33/8$ بیشتر شد. تنها مشکلی که وجود دارد اندازه محاسبه زمان

مورد نیاز در ارزیابی ضریب همبستگی می‌باشد. روش دیگری به نام بلاک‌بندی (blocking) یا روش دسته‌بندی (batching) که برای بدست آوردن داده‌های همبسته استفاده می‌شود وجود دارد.

روش بلاک‌بندی: n مشاهدات x_1, x_2, \dots, x_n به صورت بلاک‌های متوالی دسته‌بندی می‌شوند و هر یک بطول $P = n/b$ است. میانگین بلاک‌ها را به صورت زیر محاسبه می‌کنیم:

$$r_1 = \frac{x_1 + x_2 + \dots + x_P}{P}$$

$$r_2 = \frac{x_{P+1} + x_{P+2} + \dots + x_{2P}}{P}$$

.....

.....

$$r_b = \frac{x_{(b-1)P+1} + x_{(b-1)P+2} + \dots + x_{bP}}{P}$$

با فرض معادله (13.11) r_1 تا r_n امین مشاهده و x میانگین مقدار r_k ها با معادله (13.1) مقدار r_k را است. ضریب کواریانس یا ضریب همبستگی با $\log n$ نامند. برای مورد خاص $\log n = 0$ مقدارش هیچ است اما تقریب r_k ها کواریانس توزیع از r_k ها طبق معادله (13.7) تعیین می‌شود.

با استفاده از معادله (13.10) ضریب r_1, r_2, \dots, r_k به راحتی محاسبه می‌شود. در تمامی سیستم‌های فیزیکی همانطور که n افزایش می‌یابد ضریب r_k کاهش خواهد یافت. زیرا وقتی فاصله بین دو مشاهده طولانی‌تر گردد اثر یک مقدار روی دیگری ضعیف می‌شود.

بنابراین بعد از یک تعداد مشخص M این ضریب به سمت صفر میل می‌کند. $r_{M+1} = r_{M+2} = 0$. این نقطه برش M می‌بایست به اندازه کافی بزرگ باشد تا اینکه ضریب عمده را شامل شود. اما می‌بایست از n کوچکتر باشد قانون قابل قبول $n \geq M^2$ اگر هر یک از این n ها مشخصاً جدا از صفر باشند. (انتخاب یک مقدار دقیق M شامل طرحی است که براساس یک تعدادی برشهای اجرا انجام یا شکل می‌گیرد). اثر همه ضیب همبستگی غیر صفر آزمایش جاری برای محاسبه کردن کواریانس \bar{x} عبارتست از:

$$\sigma_x^2 = \frac{\sigma^2}{n} \left\{ 1 + 2 \sum_{k=1}^M \left(1 - \frac{k}{M+1}\right) r_k \right\} \quad (13.11)$$

اگر رشته از هرگونه ارتباط همبستگی آزاد باشد (به عبارت دیگر داده‌های متوالی مستقل هستند) فقط اولین جمله در معادله (13.11) باقی می‌ماند و σ^2/n کواریانس نمونه با میانگین \bar{x} را می‌دهد.

دوین جمله هم بخشی از همبستگی است. تکرار ارتباط و همبستگی به تغییرپذیری آماری مهمتر در x اشاره دارد و با حالتی که وقتی رشته مستقل است مقایسه می‌شود. با در دست داشتن فرمول (13.11) می‌توانیم حالتی را فرض کنیم که نمونه‌ها استقلال آماری ندارند. برای محاسبه کردن طول اجرا ما عبارت (13.11) را در مکانی از r_k جایگزین می‌کنیم و عبارت جاری را برای طول اجرا بدست می‌آوریم:

$$\frac{\sigma^2}{n} \left\{ 1 + 2 \sum_{k=1}^M \left(1 - \frac{k}{M+1}\right) r_k \right\}$$

۱۳.۲ - تکنیکهای تقلیل واریانس

اکثر آزمایشهای شبیه سازی به منظور بدست آوردن خصوصیات یک مجموعه از قبیل میانگین زمان انتظار برای هر مشتری در یک صف برنامه ریزی شده اند. به این منظور m مشاهده x_1, x_2, \dots, x_m را در نظر می گیریم و میانگین \bar{x} را بدست می آوریم. میزان نزدیک بودن میانگین \bar{x} به میانگین عمومی μ توسط انحراف معیار σ (از میانگین نمونه های \bar{x}) تعیین می شود. هر چقدر اندازه نمونه های n بزرگتر باشد، \bar{x} به μ نزدیکتر خواهد شد، این پدیده را همگرایی تصادفی می گویند.

مشکلی که در مورد پدیده همگرایی تصادفی وجود دارد سرعت کم آن می باشد. در بخش قبلی مطالعاتی در مورد اثر تعداد مشاهدات n با دقت \bar{x} در شرایط مختلف به عمل آمد. دیدیم که اگر نمونه ها از نظر آماری مستقل باشند یا همبسته، و همچنین اگر از مدت اجرای طولانی n یا انعکاسهای n از n به روی $n/2$ هر یک از مشاهدات استفاده کنیم، انحراف معیار میانگین نمونه ها به صورت زیر می باشد.

$$\sigma_{\bar{x}} = \frac{1}{\sqrt{n}} \quad \left(\text{فاکتور } \sigma \text{ مستقل از } n \right)$$

بنابراین برای کاهش انحراف معیار توسط یک عامل f می بایست f^2 را همانند بسیاری از مشاهدات بوجود آوریم. در بسیاری از موارد مخصوصاً در صورت وجود همبستگی یک نمونه که دورتر از نمونه های دیگر می باشد برای بدست آوردن یک سطح اطمینان، محاسبه میانگین با استفاده از فرمول (۱۳.۶) یا (۱۳.۱۳) مورد نیاز است. برای تقلیل واریانس بدون افزایش اندازه نمونه یک تکنیک وجود دارد که در زیر آمده است.

(۱) نمونه های متقابل: فرض کنید می خواهیم دو اجرای شبیه سازی هر یک با طول $n/2$ را برای محاسبه مقدار میانگین نمونه های تصادفی بدست آوریم (یا فرض اینکه n عدد زوج می باشد).

فرض مشاهدات در این دو اجرا به صورت زیر می باشد:

$$y_1, y_2, \dots, y_{n/2}, \quad z_1, z_2, \dots, z_{n/2}$$

$$\bar{y} = \frac{2}{n} \sum_{i=1}^{n/2} y_i$$

$$\bar{z} = \frac{2}{n} \sum_{i=1}^{n/2} z_i$$

$$\sigma_{\bar{y}}^2 = \sigma_{\bar{z}}^2 = \frac{2\sigma^2}{n}$$

واریانس آنها برابر است با:

و میانگین این دو دسته عبارت است از:

اندازه بالای p بزرگ انتخاب می شود که n ها از یکدیگر مستقل باشند. در یک لحظه از زمان p نباید به اندازه ای بزرگ باشد که n کوچک شود (یا فرض $pn=n$) چرا که در آن صورت توزیع \bar{x} نرمال نخواهد بود.

حال n_1, n_2, \dots, n_p را به عنوان نمونه هایی در نظر بگیرید و میانگین انحراف معیار خروجی بالای همانند زیر را محاسبه زیر محاسبه کنید.

$$\sigma_{\bar{x}}^2 = \frac{1}{b} \sum_{i=1}^b \sigma_i^2 = \frac{1}{bp} \sum_{i=1}^p \sum_{j=1}^{n_j} x_{ij}^2 = \frac{1}{n} \sum_{i=1}^n x_i^2$$

واریانس بالای V به صورت زیر فرض شود:

$$\sigma_{\bar{x}}^2 = \frac{\sigma^2}{b}$$

V واریانس، میانگین بلوک V_i را همانند \bar{x} میانگین تمامی اندازه ها در نظر بگیرید. گرچه تقسیم σ^2 ها به صورت بلاکهای مستقل هستند این اجازه را به ما می دهد که از فرمولهای ساده برای محاسبه واریانس میانگین بالای استفاده کنیم. از آنجائی که در روش بالای بزرگ اجرا را در نظر می گیریم و آن را به صورت بلاکهای b با طول p تقسیم می نماییم. در واقع یک آزمایش با طول p و کل دفعات b است و به همین ترتیب حالت نهایی یک اجرا مقدار اولیه اجرای بعدی می شود. این حالت بهتر از زمانی است که تمام اجرای b از یک مقداری اولیه مشابه شروع شده باشد. زیرا وضعیت انتهایی یک اجرا احتمالاً مشخص تر از یک وضعیت ابتدایی خواهد بود گرچه شناسایی وضعیت ابتدایی یک اجرا با وضعیت نهایی اجرای قبلی ارتباطی را بین میانگین اجرای V_i ها بوجود خواهد آورد.

اگر طول بالای b به اندازه کافی بزرگ باشد این همبستگی به علت در نظر نگرفتن بسیاری از معیارها کاهش می یابد. مقدار واقعی طول بالای p با انجام یک سری اجرای آزمایشی به منظور تعیین ارتباط بین V_i ها مشخص می شود. مطالعاتی به منظور مقایسه روش توالی داده های همبسته انجام شده است. البته شواهدی دال بر بهتر بودن یک روش از روش دیگر هنوز در دست نیست. اولین روش، نمونه های بیشتری را نسبت به روش بالای برای بدست آوردن همان سطح اطمینان احتیاج دارد.

زمان مورد نیاز برای محاسبه ضریب همبستگی از مشاهدات متعدد نسبت به زمان محاسبه بیشتر نیز خواهد بود. برای محاسبه زمان یک نمونه b محاسبه ضریب همبستگی تعیین می کند که کدام روش متداولتر است. از آنجائی که این نسبت بستگی به سیستم شبیه سازی دارد، قانون کلی که نشان دهد کدام روش قادر به انجام تمام شبیه سازی است وجود ندارد.

در واقع σ^2 واریانس نمونه‌های اصلی می‌باشد.

میانگین کل یا میانگین دو میانگین عبارتست از:

$$z = \frac{1}{n} \sum_{i=1}^n (y_i + z_i) = \frac{\bar{y} + \bar{z}}{2};$$

واریانس میانگین:

$$\sigma_z^2 = \frac{1}{n} \sum_{i=1}^n [(y_i - \bar{y})^2 + (z_i - \bar{z})^2].$$

و به شکل زیر تغییر می‌یابد:

$$\sigma_z^2 = \frac{\sigma_y^2 + \sigma_z^2}{4} + \frac{1}{2} \text{Cov}(\bar{y}, \bar{z}), \quad (13.3)$$

که $\text{Cov}(\bar{y}, \bar{z})$ به کواریانس متغیرهای تصادفی U و V اشاره می‌کند. اگر دو متغیر U, V مستقل باشند. آنگاه کواریانس آنها صفر خواهد بود. ولی در صورت ما یک همبستگی معکوس (وارونه) بین $n/2$ مشاهدات معرفی می‌کنیم. به این شکل که هرگاه ارزش y_i به سمت پایین میل خواهد کرد و بالعکس. شرط دوم در (13.13) منفی خواهد و واریانس در $n/2$ مشاهدات y_i

$$\frac{y_1 + y_2}{2}, \frac{y_2 + y_3}{2}, \dots, \frac{y_{n/2} + y_{n/2+1}}{2}$$

کمتر از واریانس (اختلاف) متوسط یک اجرای پیوسته برای n مشاهدات خواهد بود. این روش نمونه‌برداری متقابل مانند که به معرفی یک همبستگی منفی بین یک جفت مشاهدات، برای بدست آوردن پاسخ سطح اطمینان آماری می‌پردازد. عمومی‌ترین راه برای بدست آوردن یک جفت نمونه با همبستگی منفی نمایش دو شیشه‌سازی است که به صورت یکسان با استفاده از مجموعه‌های متفاوت از اعداد یکدست تصادفی بین (0 و 1) اجرا می‌شوند.

اگر در اولین اجرا، اعداد تصادفی، $y_1, y_2, \dots, y_{n/2}$ باشند آنگاه در دومین اجرا این اعداد $(1-y_1), (1-y_2), \dots, (1-y_{n/2})$ خواهند بود. پس اگر U بزرگتر از میانگین باشد، $(1-y_i)$ کمتر خواهد بود و بالعکس. پس این دو مجموعه از اعداد تصادفی همبستگی منفی دارند، در نتیجه خروجیهای مربوطه y_1 و $y_{n/2+1}$ دارای همین

خصیصه هستند. هر چند این دو مجموعه اعداد تغییر حالتی را در حین عمل شیشه‌سازی باعث می‌شوند با این حال انتظار داریم که نتایج حتی پس از تغییر حالتی شیشه‌سازی، در همان حالت همبستگی منفی باقی بمانند.

به جز روش استفاده از جفت اعداد تصادفی $(1-y_i)$ که عمومی‌ترین راه در روش نمونه‌برداری متقابل است، روشهای دیگری برای شیشه‌سازی‌های خاص وجود دارد. به طور مثال در سیستم صفی که فقط به یک مشتری سرویس می‌دهد، می‌دانیم تا مشتری $(k-1)$ ام سرویس نگیرد مشتری k ام سرویس نخواهد گرفت. پس زمان ورود مشتری به صف همبستگی مثبت با زمان سرویس‌گیری شخص در صف دارد. لذا زمان سرویس‌گیری، تأثیر مخالف بر روی زمان یک مشتری در سیستم دارد که البته ما می‌توانیم این اختلاف را کاهش بدهیم.

اگر شیشه‌سازی صف را دو مرتبه با شرایط شروع یکسان به شکل زیر انجام دهیم: به قسمی که ابتدا رشته اعداد تصادفی مولد زمانهای بین ورود را در اولین نسخه برای تولید زمانهای سرویس دهی و در دومین نسخه انجام دهیم. و سپس با استفاده از اعداد تصادفی برای تولید زمانهای سرویس در اولین نسخه برای تولید زمانهای بین ورود دومین نسخه بپردازیم.

پس نمونه برداری متقابل به عنوان مفیدترین تکنیک در تمرینها وجود دارد و تنها وسیله مورد نیاز این تکنیک آن است که راهی را معرفی کنیم که تا یک همبستگی منفی بین دو مجموعه ورودی وجود داشته باشد و البته کار مشکلی هم نیست. بهترین منفعت استفاده از نمونه‌برداری متقابل آن است که این تکنیک به هر برنامه شیشه‌سازی کاپیرتری با کمترین تغییرات در برنامه، می‌تواند اضافه شود.

(۲) نمونه‌برداری همبسته: این تکنیک برخلاف تکنیک "نمونه‌برداری متقابل" می‌باشد و همگامی مفید است که قصد تخمین اختلالات در میانگین تحلیل در طراحی یا استراتژی جایگزینی داریم. فرض کنید \bar{A} و \bar{B} نمونه‌های تخمینی هستند که ما آنها را به عنوان میانگین زمان انتظار هر مشتری از دو صف مختلف می‌شناسیم. قصد داریم اختلاف کارایی بین این دو را تقریب بزنیم. اگر \bar{A} و \bar{B} از دو نمونه‌های کاملاً تصادفی بدست آمده باشند. آنگاه اختلاف آنها به شکل زیر است:

$$\bar{D} = \bar{A} - \bar{B}$$

اگر \bar{A} و \bar{B} دو متغیر مستقل آماری باشند آنگاه مقدار اختلاف به وسیله واریانس آنها (\bar{D}) به کمک این فرمول تعیین می‌شود:

بگیریم تا اعداد کافی برای میانگین هزینه بازگشت از سفارش را داشته باشیم. این طول اجرا n به اندازه کافی باید بزرگ باشد تا بتواند میانگین هزینه حمل و میانگین هزینه سفارش مجدد را شامل شود. این عمل تا زمانی که بتوان یک شمای "نمونه برداری نفوذی" شامل یک سری اعداد بزرگ برای وضعیتهای سفارش مجدد با رخداد طبیعی داشت، ادامه خواهد یافت. به طور کلی مشکل ترین بخش استفاده از "نمونه برداری نفوذی" تصمیم گیری برای تعیین شکل مقتضی توزیع داده شده می باشد.

(۴) **متغیر کنترلی:** در شبیه سازی یک سیستم پیچیده، می توان یک نسخه ساده شده از سیستم را جدا کرده و به صورت تحلیلی مقدار پارامتر مرتبط با پارامتر شبیه سازی را حساب کرد. به طور مثال، فرض کنید قصد داریم یک سیستم صف با چند سروس دهنده که یک سیستم پیچیده است را شبیه سازی کنیم و پارامترها طی این عمل میانگین طول صف می باشد. این سیستم بدین صورت عمل می کند که اولین ورودی اولین گیرنده سروس است.

سیستم یک صف با یک سروس دهنده قابل کنترل است. میانگین طول صف طبق معادله (۱۲.۵) است. اگر زمان ورود به سیستم و زمان سروس مطابق توزیع نمائی باشد، پس ما مدل ساده تر را با استفاده از رشته اعداد تصادفی U_1, U_2, \dots, U_n شبیه سازی می کنیم. اگر \bar{x} را تخمین طول صف که در طی شبیه سازی بدست می آید فرض کنیم و \bar{t} مقدار تحلیلی همان پارامتر باشد پس اختلاف آنها است.

$$E = \bar{x} - \bar{t}$$

که به ما خطاهای نمونه برداری را هنگامی که رشته اعداد تصادفی U_1, U_2, \dots, U_n نشان می دهد این خطا (E) ممکن است به عنوان یک تخمین خطای نمونه برداری در شبیه سازی سیستم های پیچیده تر استفاده شوند، اگر رشته اعداد تصادفی همان U_1, U_2, \dots, U_n باشد.

پس طبق آخرین مرحله ای که شبیه سازی سیستم های پیچیده است، از رشته اعداد تصادفی U_1, U_2, \dots, U_n استفاده می کنیم. ارزش پارامتر وابسته به آنها را با \bar{x} بدست می آوریم حال اگر این مقدار را \bar{t} بگیریم مقدار مشاهده شده \bar{t} را به وسیله تفریق E از آن، تصحیح می کنیم پس مقدار پارامتر تخمین زده شده عبارت است از:

$$\bar{z} = \bar{y} - E = \bar{y} - \bar{x} + \bar{t}$$

واریانس \bar{z} در این تخمین جدید کوچکتر از واریانس \bar{t} خواهد بود زیرا \bar{x} و \bar{t} به دلیل آنکه هر دو از یک رشته اعداد تصادفی به عنوان ورودی استفاده کرده اند وابستگی مثبت دارند. این تکنیک استفاده از یک متغیر کنترلی در هر دو سیستم ساده و پیچیده برای معرفی وابستگی مثبت بین پاسخهایشان است. در این صورت کار بسیار راحت می باشد و دیگر نیازی به هیچ نسخه یا تحریکی برای مکانیزم

$$\sigma_B^2 = \sigma_A^2 + \sigma_B^2$$

اما اگر همان دو مجموعه از اعداد تصادفی که در این دو آزمایش استفاده شده اند مستقل نباشند بلکه وابسته باشند، در این شرایط واریانس اختلاف آنها کوچکتر خواهد بود زیرا:

$$\sigma_B^2 = \sigma_A^2 + \sigma_B^2 - 2 \text{Cov}(A, B)$$

و در نتیجه می توان، بدون آنکه زمان اجرای برنامه در کامپیوتر را افزایش بدهیم، واریانس را کاهش داد.

(۳) **نمونه برداری نفوذی:** ایده استفاده از نمونه برداری نفوذی بدین شکل است که از یک روال نمونه برداری که تغییر شکل سنجیده ای روی توزیع اصلی بوجود می آورد استفاده می شود. پس یک نمونه برگزینی برای نمونه برداری بوجود می آید. سپس از یک تابع وزن دار مناسب برای ضرب کردن داده های نمونه استفاده می شود تا تغییر شکل ارائه شده تصحیح گردد.

به طور مثال فرض کنید کارمندان راه آهن می خواهند میانگین سطح تحصیلی افرادی که با قطار سریع السیر مسافرت می کنند را تخمین بزنند. همچنین مشخص است که بخش عظیمی از تغییر سطح تحصیلات مسافران زن از ۱۶ تا ۱۶ سال تحصیلی می باشد و برای مردان این تغییر کمی کوچکتر است. اکثر این مسافران از لیسانس تا بالاترین درجه دکتری هستند (۱۴ سال تا ۱۹ سال تحصیلی). فرض کنید ۸۰٪ از مسافران مرد و ۲۰٪ از مسافران زن هستند و ما از این تعداد یک نمونه ۲۰ تایی انتخاب کرده ایم.

نتیجاً در این نمونه تصادفی ۱۶ نفر مرد و ۴ نفر زن خواهد بود. در هر صورت اگر نمونه زیاد کوچک انتخاب شود هماتقدر بی معنی و بی استفاده خواهد بود که همان نمونه را بزرگ انتخاب کنند. پس بهتر بود نمونه شامل ۱۲ زن و ۸ مرد باشد. در چنین نمونه ای، تخمین بهتری از میانگین را می توان داد. حال می خواهیم وزن مقتضی برای بدست آوردن میانگین کلی را بیابیم. پس لازم است مثالی دیگر را توضیح دهیم.

فرض کنید که در یک سیستم انبار هزینه سفارش واحد بیشتر از هزینه نگهداری واحد است. برای بهترین Q و P ، میانگین هزینه بازگشت از سفارش بخش مهم برای هزینه میانگین کل خواهد بود، اما برگشت از سفارش نادر اتفاق می افتد. پس تحت این چرخه یک تقریب قابل اطمینان برای میانگین هزینه روزانه C مورد نیاز است می باید طول اجرا را که در اینجا با n نشان می دهیم، به اندازه کافی بزرگ

۱۳-۳- خروجی آزمایشی

شرح می‌دهیم که چگونه یک آزمایش شبیه‌سازی تحت مجموعه‌ای از شرایط عملیاتی انجام می‌شود (برای یک مجموعه مقادیر متغیرهای کنترلی). ولی به ما به این مسائل و شرایط تحت عمل توجه نمی‌کنیم بلکه از آنها به عنوان داده‌های ارائه شده استفاده می‌کنیم. در این بحث در مورد مشکلات و مسائل شرایط انتخاب بحث خواهیم کرد.

در یک آزمایش متغیر مستقل را "فاکتور" و متغیرهای وابسته به آن "پاسخ" می‌گویند. برای مثال در مسئله شبیه‌سازی انبار ما ۷ فاکتور (یا متغیرهای طرح) داشتیم که سه هزینه h_1, h_2, h_3 می‌باشند و نقطه سفارش مجدد حد Q_R می‌باشند و دو تابع چگالی احتمال برای تقاضا و زمان هدایت هستند. هزینه کل انبار یعنی C یک پاسخ است. در شبیه‌سازی مسائل انبار، قبلاً می‌خواستیم هزینه C را برای مجموعه‌ای از فاکتورهای مشخص پیدا کنیم به هر حال اگر هدف ما از شبیه‌سازی انبار مقایسه هزینه تحت شرایط عمل خاص است ما در فصل قبل آرا انجام دادیم گرچه ما آنکار سازی و تخمین ارتباط تابعی زیر می‌باشد:

$$C = f(h_1, h_2, h_3, Q_R, p, O, \text{زمان تقاضا})$$

بین پاسخ C و هفت فاکتور ارتباط وجود دارد و کار هنوز پایان نیافته است.

انتخابهای متفاوت ما برای یک فاکتور "سطوح" نامیده می‌شود. اگر در یک آزمایش k فاکتور مختلف و هر کدام با m سطح متفاوت وجود داشته باشد آنگاه مجموع ورودهای مختلف یا نقاط طراحی (که رفتار نامیده می‌شود) برابر با k^m خواهد بود. در صورتی که از تمامی k^m نقاط طراحی استفاده کنیم به این عمل طراحی با فاکتور ریل کامل گویند. البته این عمل بسیار گران قیمت است و زمان زیادی در کامپیوتر می‌گیرد. برای مثال مسئله انبار فرض کنید فقط ۲ سطح برای هر فاکتور وجود دارد پس $2^7 = 128$ رفتار خواهیم داشت و اگر برای هر کدام از ۱۲۸ آزمایش ما طول اجرایی برابر ۱۸۰ روز (۶ ماه) و ۳۰ تکرار برای هر اجرا بگیریم آنگاه $128 \times 180 \times 30$ اجرا مختلف داریم اگر هر اجرا ۱۵ ثانیه طول بکشد ۱۶ ساعت زمان کامپیوتر می‌برد. هر کدام از فاکتورها تأثیری روی جواب خواهد داشت. در آخر در مورد تأثیر دو فاکتور (h_1, C) بر روی یکدیگر بحث خواهیم کرد.

طراحی کسری از فاکتور ریل: گاهی اوقات طراحی تمام فاکتور ریل بسیار گران قیمت خواهد بود که در این صورت باید به دنبال مدلی بود که کمتر از طراحی تمام فاکتور ریل قیمت داشته باشد. سناری که وجود دارد آن است که تأثیر این کاهش رفتار بر روی توانایی تخصیص و تخمین روابط تابعی بین فاکتورها و جوابها دارد چیست؟ یک روش سیستماتیک طراحی کسری از فاکتور ریل می‌باشد. البته بخش بزرگی از نوشته‌های کلاسیک برای طرح آزمایشات به عنوان بخش برای آمارها وجود دارد. زیرا

نمونه برداری وجود ندارد.

(۵) نمونه برداری دسته‌بندی شده: روش دیگر برای تقلیل واریانس بدون داشتن افزایش اعداد نمونه، شکستن آنها به دسته‌ها و طبقه‌های مختلف است به شکلی که در هر دسته حداقل تغییر باشد. سپس تعداد کم مقتضی از نمونه در هر دسته را (اگر اختلاف باشد فقط یک نمونه مورد نیاز است) در نظر می‌گیریم، آنگاه این نمونه‌ها برای بدست آوردن یک تخمین نهائی ترکیب می‌شوند. طبیعی است چنین نمونه برداری بسیار مؤثر از روش نمونه برداری تصادفی است.

روشهای گوناگونی برای دسته‌بندی جمعیتها در سیستم وجود دارد. مؤثرترین آنها دسته‌بندی کردن، به قسمتی است که واریانس برای هر دسته یکی باشد که البته این خود نیاز به داشتن دانش بیشتری در مورد سیستم است و باید دانست چند نمونه برای هر طبقه و دسته داشت که البته این امر کمی مشکل است.

(۶) روانیت روسی و شکاف: فرض کنید می‌خواهیم با انداختن دو تاس در یک لحظه، احتمال آنکه مجموع دو تاس عدد شود را بدست آورد. حال به جای آنکه ابتدا دو تاس را با هم پرتاب کرده و سپس مجموع آنها را بشماریم می‌توان یک تاس را انداخت و دومی را پس از مشاهده نتیجه تاس اول پرتاب کرده اگر دو مرحله اول تاس پیش از عدد ۱ یا ۲ را بیازید نیازی به پرتاب تاس بعدی نیست و نتیجه‌ای عمل را قبول نکرد. و به پرتاب بعدی می‌رویم. بدون تأثیر صحت تخمین‌ها، این عمل باعث می‌شود که نیازی نباشد تا ۴ الی ۶ مرتبه تاس را بیندازیم. به طور کلی در این تکنیک نیاز به تعریف دسته آزمایشات در هر مرحله را داریم و این روش باعث کاهش واریانس ها می‌شود اگر در هر مرحله از آزمایشات تعریف صحیح و درستی دریافتی در هر مرحله آزمایش را داشته باشیم. گرچه این وسیله اساسی برای مدت طولانی توسط مهندسان کنترل کیفیت جهت نمونه برداری ترقیتی استفاده شد است. اما برای آنکه در یک شبیه‌سازی مورد استفاده قرار گیرد به آسانی عملی نیست.

به دلیلی آنکه شبیه‌سازی هنوز یک تکنیک گران قیمت است پس این خود باعث افزایش ارزش تخمین زدن است و باعث افزایش درجه وقت در عمل می‌شود. پس به همین دلیل است که روشهای گوناگونی را جستجو کنیم تا باعث کاهش خطاهای تصادفی گردد. و این در صورتی است که علم کافی و واقعی در مورد سیستم تحت مطالعه داشته باشیم. بدینوسیله اگر اطلاعاتی نداشته باشیم، چیزی هم بدست نخواهیم آورد. البته باید دانست کلیه روشهای (تقلیل الحراف معیار) برای کلیه وضعیتها مناسب نیست. در صورت عدم علم کافی از اینکه کدام تکنیک را به کار ببریم باعث بدتر شدن صحت عمل خواهد شد.

و یک سرویس دهنده را استفاده کرده ایم. این روش برای اعتبار دادن بسیاری از موارد مفید و قابل قبول است. البته مواردی هم وجود دارد که خروجی حتی برای ورودیهای ساده قابل پیش بینی نباشد.

اعتبار دادن به سیستم های موجود:

هرگاه سیستم های شبیه سازی شده در دنیای واقعی وجود داشته باشند، واضح ترین و بهترین کار، استفاده از ورودیهای واقعی و مقایسه خروجیهای آن با سیستم واقعی می باشد. این پروسه کار ساز است گر چه ممکن است در عمل مشکلاتی را نیز به بار آورد. مثلاً ممکن است همیشه داده های ورودی و خروجی را به راحتی بدست نیاوریم (از سیستم هایی که در دنیای واقعی وجود دارند) و یا آنکه اگر داده ورودی و خروجی واقعی را از مدل موجود بدست آوریم، ممکن است برای زمانی طولانی در آن سیستم صادق نباشد. نظر به اینکه داده ها اغلب تصادفی هستند (که از بخش های قبلی فهمیدیم)، آزمایش های طراحی که با نمونه های کوچک کار می کنند مشکل است. آنچه معمولاً انجام شده، مدل را چندین بار، با رشته اعداد تصادفی شبیه سازی کردن و بدست آوردن محدوده اختلافها است. پس اگر مدل معتبر است، خروجیهای واقعی باید، جایی در خروجی مدل قرار گیرد. سوئین مشکل، ارتباط است، بدین معنی که خروجی مدل و خروجیهای سیستم واقعی در واقع از یک جمعیت و گروه هستند. اگر خروجیهای مقایسه شده جزو نمونه باشند (مثلاً میانگین طول صف، زمان انتظار، زمان تلف شده) خود می توانند از یک عدد برای آزمایش های آماری جهت اندازه گرفتن اختلاف بین دو خروجی استفاده کند (خروجی مدل و خروجی سیستم واقعی). مانند آزمایش خی دو (کای دو) که توضیح داده شده و آزمایش گولومگروف - اسمیرنوف، کرامر - فون مایزر و تست لحظه ای مشخص می تواند از آزمایش فرضی برای تصمیم گرفتن، در صورتی که اختلاف مهمی بین مجموعه مشاهدات مستقل وجود داشته باشد استفاده کند. در اکثر آزمایشات شبیه سازی خروجی بدست آمده از سریهای زمانی است (مثل طول صف در زمانهای مختلف) که خود وابسته هستند. حال چگونه دوسری زمانی وابسته و آزمایش آنها را مقایسه کنیم؟ تحلیل هندسی یکی از روشهایی است که برای این کار پیشنهاد شده است.

مدل زمان - اولین اعتبار:

اگر یک مدل به قصد توصیف کردن سیستمی باشد که تاکنون موجود به نوره و اولین بار مرتبه بوجود آمده وظیفه معتبر سازی مدل بسیار سخت تر و پیچیده تر می باشد. زیرا هیچگونه داده هایی برای مقایسه کارایی آنها وجود ندارد. نظر به اینکه سیستم های فرضی به خاطر طبیعتشان براساس

این ابزارهای آماری برای تجربیات در صنعت پیشرفت کرده اند. زیرا به عنوان یک وضعیت ثابت و داده های خروجی مستقل که براساس آزمایش های آماری است وجود دارند. و نه به عنوان آزمایشات شبیه سازی، در نتیجه استفاده شان در طراحی آزمایشات شبیه سازی ضایع نیست.

جستجوی برای قاطع طراحی بهینه: هدف کلی برای تعداد زیادی از مطالعات شبیه سازی تصمیم گرفتن در مورد ترکیب پارامترهای ورودی است که باعث ماکزیمم یا مینییم ساختن خروجی می باشد. برای مثال مقصد نهایی شبیه سازی یک سیستم انبار یافتن خط مشی (P,Q) با کمترین هزینه C، که با هزینه های ثابت تقاضا در زمان هدایت شده است می باشد. روشهای مختلفی در تئوری بهینه سازی موجود است که می گویند چگونه برای یک نقطه بهینه در یک سطح جواب جستجو کرد و ما را به نقطه مطلوب و دلخواه همراه با کمترین محاسبه راهنمایی می کند.

۱۳-۴ - متغیر سازی

چگونه ما اطمینان داریم مدلی که از سیستم ساخته ایم نمایش صحیحی از سیستم شبیه سازی شده می باشد؟ سوال مهمی وجود دارد که قبل از آنکه به مطالعه شبیه سازی دست بزنیم باید به آن جواب داد و این که بدون یک مدل معتبر هیچ چیزی برای ما وجود ندارد. در واقع بدون برقراری و بر وجود آوردن یک مدل قابل اعتبار نتایج لازم و (اگر نتایج شبیه سازی را بپذیریم) مایوس کننده می باشد. اما مدل معتبر چیست؟

ما همیشه باید مدلی معتدل از سیستم واقعی را درخواست کنیم. سطوح قابل قبول تقریبی از سیستمی به سیستم دیگر و از شبیه سازی به شبیه سازی دیگر متفاوت خواهد بود و یک مدل کلی و جامع به عنوان نماینده مدلهای برای شبیه سازی وجود ندارد. پس می توان گفت مدلهای متفاوتی برای شبیه سازیهای مختلف وجود دارد و تنها چیزی که در اختیار داریم، راهنماییایی برای یک مدل با اطمینان و قابل قبول است. تلاشهای معتبر سازی به دو بخش تقسیم می شوند: اعتبار مدل انترازی و اعتبار پیاده سازی آن است. اولین قسمت شامل آزمون کلیه فرضیه هایی است که از سیستم واقعی به مدل مفهومی منتقل می شود. بخش بزرگی از قضاوت و علم سیستم واقعی در این مرحله خلاصه می شود، باید دانست اعتبار مدل انترازی بسیار مفهومی است. آزمایش اعتبار یک پیاده سازی مفهومی و کار ساده تری است که شامل بررسی منطقی، فلوجارت و برنامه کامپیوتری برای اطمینان پیاده سازی کامل مدل است. ساده ترین روش برای اعتبار یک پیاده سازی استفاده از مسائل نمونه است. ورودی نمونه باید به گونه ای انتخاب شوند که خروجیهای وابسته تحلیلی بدست آید.

بطور مثال قبلاً از توزیع ورودی مخصوص برای اعتبار فلوجارت و برنامه کامپیوتری یک مدل صف

فرضیات وجود دارند، مدل شبیه‌سازی وابسته به اعتبار این فرضیات می‌باشد. رهنمودهایی برای امتحان اعتبار چنین سیستم‌هایی وجود دارد که عبارتند از:

(۱) اعتبار زیر سیستم: ممکن است برای یک سیستم، دیگری که قابل مقایسه با آن باشد وجود نداشته باشد اما در همان سیستم، زیر سیستم‌هایی شناخته شده که بتوان اعتبار آنها جداگانه امتحان کرد وجود داشته باشد.

(۲) اعتبار داخلی: اگر تغییرات داخلی زیاد باشد معمولاً روش مورد استفاده رد می‌شود. سیستم تصادفی با انحراف معیار بالا به علت پروسه‌های داخلی، تغییرات خروجی را به علت تغییرات ورودی داخلی محو می‌کند. تست مورد نظر توسط تکرار یک شبیه‌سازی با قوای اعداد تصادفی متفاوت انجام می‌شود و سپس انحراف معیار خروجی محاسبه می‌شوند. اگر انحراف معیار زیاد باشد مدل رد می‌شود.

(۳) آتالیز حساسیت: آتالیز حساسیت شامل تغییر مختلف پارامترها یا متغیرهای ورودی در یک لحظه در پاسخ مدل می‌باشد. آتالیز حساسیت نشان می‌دهد که سیستم نسبت به کدام پارامترها حساستر است. سیستم معمولاً به پارامترها که دقیق نیست پاسخ می‌دهد، بنابراین احتیاجی به دقت زیاد نیست.

بنابراین به راحتی می‌توان ضریب اطمینان مدل بکار گرفته شده را تعیین کرد.

(۴) اعتبار ظاهر: اگر مدل برخلاف عرف و منطق باشد رد خواهد شد و (حتی اگر مثل یک سیستم واقعی عمل کند) اگر سیستم مدل را به عنوان یک مدل معقول نپذیرد خواهد شد، تست‌های تعیین صحت فوق و یا دیگر تست‌ها نمی‌توانند به طور کامل درست بودن یک مدل را تعیین کنند. بنابراین تنها در میزان ضریب اطمینان تغییر ایجاد خواهد شد. اشتباهات مدل باید از اشتباهات اجرا (اشتباهات برنامه) تمیز داده شود.

بنابراین در ابتدا می‌بایست قبل از نوشتن شبیه‌سازی از صحت مدل اطمینان داشت (در عمل گر چه تعیین صحت مدل ریاضی بدون آزمون توسط کامپیوتر مسیر نیست) البته این موضوع به علت ریاضیات مدل می‌باشد که دلیلی برای شبیه‌سازی آن در ابتدای امر است. تعیین صحت مدل مقرون به صرفه نمی‌باشد چرا که با در نظر گرفتن صرف زمان زیاد نتایج قابل توجهی را حاصل نمی‌کند تنها اشتباهات قابل توجه موجود در مدل و همچنین ضریب اطمینان نسبت به نتایج شبیه‌سازی را به ما می‌دهد.

بخش پنجم

زبانهای شبیه‌سازی گامپوتوری

فصل چهاردهم - زبانهای شبیه‌سازی

فصل پانزدهم - مدل‌سازی مقدمه‌ماتی SLAM

فصل شانزدهم - گره‌های منابع و دروازه‌ها

فصل هفدهم - گره‌های تصمیم‌گیری و منطقی

فصل هجدهم - ورودی‌ها و خروجی‌ها

فصل نوزدهم - مدل‌سازی شبکه باروین تعریفی کاربر

فصل چهاردهم

زبانهای شبیه سازی

در فصول گذشته بیان شد که شبیه سازی کامپیوتر اساساً یک تکنیک آزمایشی برای مطالعه رنج وسیعی از سیستمهای مورد استفاده را فراهم می کند. هدف آزمایش مشاهده رفتار سیستم (و یا یک مدل مشخص آن) در یک محیط واقعی یا شبیه آن است. مدل مجرد سیستمی که باید شبیه سازی شود بفرم برنامه های کامپیوتری تبدیل و رفتار سیستم بوسیله خروجی های اجرای برنامه ارائه می گردد. در این کتاب از زبان های خاص عمومی مانند FORTRAN برای شبیه سازی عام بطور گسترده استفاده نشده است، با وجود اینکه اینها بهترین زبانهای موجود در صورت نیاز استفاده کرد می یابند از زبانهای ALGOL، PL/I و COBOL یا هر زبان همه منظوره دیگر در صورت نیاز استفاده کرد ولی به علت آن که مشکلات درگیری زبانهای همه منظوره جهت این امر بسیار زیاد است صرفاً روالهای شبیه سازی ارائه شد.

ممکن است مشاهده کرده باشید که یک مدل شبیه سازی نه چندان پیچیده (مانند صف دو سرورس دهنده یا یک شبکه فعالیتها) می تواند جهت برنامه نویسی، خطایابی و تعیین اهداف با مشکل روبرو باشد. یک برنامه نویس که بطور متوالی شبیه سازی انجام می دهد بهتر است که یک زبان سطح بالای تک منظوره را فرا بگیرد تا برنامه نویسی شبیه سازی را راحت انجام دهد. یادگیری یک زبان شبیه سازی باعث افزایش سرعت تولید برنامه شبیه سازی و سادگی استفاده آن می شود.

یک زبان شبیه سازی در کنار هدف برنامه نویسی اهداف مفید دیگری نیز ارائه می کند و یک قالب مفهومی ایجاد می کند که در آن سیستم آنالیز می تواند با ساختن مدل سیستم و مطالعه آن سوالهای اساسی خویش را بیابد. یک زبان شبیه سازی ترمینولوژی شرح یک سیستم پویا و پیچیده را فراهم می آورد.

هرچند محدوده وسیعی از زبانهای شبیه سازی ارائه شده وجود دارند ولی نمی توان هیچ زبان خاصی را به عنوان "بهترین"، "مفیدترین" یا "عمومی" یافت. بیشتر این زبانها برای محدوده خاصی از کاربردها مناسبند. اگر ما یک سیستم نظامی را شبیه سازی کنیم باید زبان MILITRAN انتخاب گردد اما اگر یک مدل دینامیکی جهانی را شبیه سازی می کنیم شاید زبان DYNAMO برگزیده شود و برای

که به وسیله معادلات دیفرانسیل شرح داده می‌شوند و حل تحلیلی آنها بسیار مشکل بود. این سیستمها بوسیله یک بلاک دیاگرام ارائه و سپس روی کامپیوتر آنالوگ شبیه‌سازی می‌شدند.

به محض ورود کامپیوترهای دیجیتال با توجه به مزایایش (از قبیل دقت زیادتر و آزادی از مقیاس بندی) جایگزین کامپیوترهای آنالوگ شد. از اینرو برنامه‌های خاصی روی کامپیوترهای دیجیتال جهت عملکرد آن مانند یک کامپیوتر آنالوگ پیاده‌سازی گردید. شبیه‌سازهای آنالوگ دیجیتال (برنامه‌ها) نام گرفت که منظور آنها کامپیوتر آنالوگ یا چک نتایج یک شبیه‌سازی آنالوگ روی یک کامپیوتر دیجیتال بود. این زبان شبیه‌سازی، (که یک کامپیوتر دیجیتال یک کامپیوتر آنالوگ را تداعی می‌کرد) به عنوان یک "زبان شبیه‌سازی ساخت یافته-بلاک" سیستم پیوسته نامیده شد و کاربر نیاز به ارائه یک دیاگرام از کامپیوتر آنالوگ داشت و سپس این دیاگرام وارد سیستم می‌شد. پس از مدتی "زبانهای شبیه‌سازی براساس-عبارت" نیز توسعه یافت.

۱۴.۳- زبانهای شبیه‌سازی ساخت یافته-بلاک پیوسته

از اولین زبانهایی بود که جهت شبیه‌سازی ارائه گردید. برای شبیه‌سازی سیستمهایی طراحی شد که می‌توانند به صورت دیاگرامهای بلاک آنالوگ ارائه شوند لذا کاربرد آنها محدود بود. اولین سیمولاتور آنالوگ دیجیتال بوسیله Selfridge در سال ۱۹۹۵ روی کامپیوتر IBM 701 پیاده‌سازی گردید که از قوانین سیمومون برای انتگرال استفاده کرد. سیمولاتور Selfridge بوسیله زبانهای مشابه (بلاک-گر) در اواخر ۱۹۵۰ و ۱۹۶۰ دنبال شد.

بهترین موارد شناخته شده عبارتند از: DEPI (سال ۱۹۵۸ روی کامپیوتر 204 DATATRON، از روش راگ-کوتا برای انتگرالگیری استفاده شد)، DEPI-4 (سال ۱۹۵۹ برای کامپیوتر IBM 704)، DYSAC (در سال ۱۹۶۱ روی کامپیوتر CDC 1604، از روش چهارجمله‌ای راگ-کوتا استفاده کرد)، DAS (سال ۱۹۶۲ روی کامپیوتر IBM 7090، از فرمول اولیور برای انتگرال استفاده کرد) و MIDAS (سال ۱۹۶۳)، MIDAS، بعنوان یک مفسر مشتر و در سال ۱۹۹۶ یک نگارش کمپایلری آن ارائه شد. این زبان روی کامپیوترهای بسیاری از قبیل IBM مدل‌های 1620، 7040 و 7090 همچنین UNIVAC مدل‌های 1107 و 1108 موجود است.

در نیمه ۱۹۶۰ که استفاده از کامپیوتر دیجیتال برای شبیه‌سازی سیستمهای پیوسته توسعه یافت، زبانهای شبیه‌سازی بلاک گرا منتشر و توسعه زبانهای جدید ادامه یافت. اکنون یک زبان شبیه‌سازی پیوسته بلاک-گرا یا یک مثال شرح می‌دهیم:

در یک زبان ساخت یافته-بلاک کاربر اول بلاک دیاگرام سیستمی که باید سیموله شود را طرح

شبیه‌سازی یک سیستم نمونه برداری داده‌ها ممکن است BLODIB استفاده شود ولی برای شبیه‌سازی زمان بندی کارها ممکن است SLAM یا GPSS را ترجیح دهیم.

علاوه بر آن طبیعت سیستمی که شبیه‌سازی می‌کنیم و ساخت افزار یا نرم افزار موجود نیز در انتخاب یک زبان تعیین کننده هستند. اگر فقط یک متنی کامپیوتر داریم، نمی‌توانیم زبان مانند HLL انتخاب کنیم. SIMSCRIPT را برای آزمایشات شبیه‌سازی پیاده‌سازی نمائیم و وجود یک زبان بهتر از انتخاب آن است زیرا تمامی زبانها روی همه کامپیوترها موجود نمی‌باشند. در این فصل مروری بر زبانهای شبیه‌سازی بدون شرح جزئیات آنها داریم. جزئیات آنها در راهنماهای مختلف یافت می‌شود و مثالهایی نیز از بعضی ارائه می‌گردد (زبان SLAM در فصول آتی شرح داده خواهد شد).

۱۴.۱- زبانهای شبیه‌سازی گسسته و پیوسته

در یک نظر زبانهای شبیه‌سازی به دو دسته گسسته و پیوسته تقسیم می‌گردند و مطابق این امر بیشتر زبانهای شبیه‌سازی در یکی از این دو دسته قرار می‌گیرند. زبانهای شبیه‌سازی پیوسته برای شبیه‌سازی مدل‌های پیوسته و زبانهای شبیه‌سازی گسسته برای مدل‌های گسسته طراحی شده‌اند. اخیراً تعداد کمی زبان طراحی شده‌اند که برای هر دو مدل‌های گسسته و پیوسته به خوبی مناسب است. این زبانها "به زبانهای شبیه‌سازی ترکیبی" موسومند که بطور خاص برای سیستمهایی نوشته شده‌اند که بعضی متغیرهای آنها تغییرات پیوسته و متغیرهای دیگر آنها تغییرات گسسته دارند. البته این متغیرهای گسسته و پیوسته با یکدیگر مجاوره یا ارتباط دارند. برای مثال، تعداد مسافران در یک بالابر هواپیما بطور گسسته تغییر می‌کند درحالی که فاصله هواپیمای برخاسته از زمین یا سرعت آن تغییرات پیوسته دارد. GASP IV شاید بهترین زبان شناخته شده ترکیبی شبیه‌سازی است.

مثالهای دیگر عبارتند از:

SLAM (Simulation Language for Alternative Method)
 GSL-A (Generalized Simulation Language)
 CLASS (Composite Language Approach For System Simulation)
 PROSE (Problem Level Programming System)

۱۴.۲- زبانهای شبیه‌سازی پیوسته

قبل از استفاده گسترده کامپیوترهای دیجیتال برای شبیه‌سازی سیستمهای پیوسته، از کامپیوترهای آنالوگ استفاده می‌شد. سیستمهای شبیه‌سازی شده عموماً سیستمهای مهندسی بودند

(تغذیه‌کننده پازاریت) با نرخ ثابت شناخته شده $\left(\frac{dH}{dt} = K_1 H\right)$ تولید می‌شود اگر هیچ پارازیتی نباشد که آن را بیلد. محصوله پازاریت P با نرخ $\left(\frac{dP}{dt} = -K_2 P\right)$ کاهش می‌یابد اگر هیچ ازدحامی برای زنده کردن آن نباشد. کاهش تعداد H ها و افزایش تعداد پارازیتها به عنوان تابعی "قابل شمارش" بین ازدحام‌کننده‌ها (H) و پارازیتها (P) است. این سیستم بوسیله یک جفت معادله دیفرانسیل غیرخطی بیان می‌شود:

$$\frac{dH}{dt} = K_1 H - K_2 H P$$

$$\frac{dP}{dt} = -K_2 P + K_3 H P$$

فرض کنید که چهار مقدار ثابت بصورت زیر باشند:

$$K_1 = 0.05 / \text{ساعت}$$

$$K_2 = 0.10 / \text{هر ساعت}$$

$$K_3 = 0.0002 / \text{ساعت}$$

$$K_4 = 0.0002 / \text{ساعت}$$

هر ۵۰۰۰ ساعت که یک تغذیه تولیدی بلعیده و یک پارازیت متولد می‌شود، نتیجه آن تعدادی معین پارازیت و تغذیه ازدحام است.

ما مقدار اولیه جمعیت تغذیه و پارازیت را $H(0)$ و $P(0)$ قرار می‌دهیم و می‌خواهیم دو جمعیت آنها را به عنوان تابعی از زمان تعیین کنیم. سیستم بوسیله بلاک دیاگرام شکل ۱۴-۲ ارائه شده است (بلاکهای شماره ۱۹، ۲۶، ۲۷).

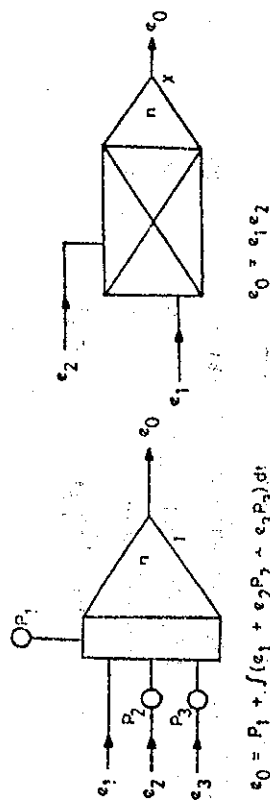
جهت ارائه این بلاک دیاگرام به کامپیوتر، برنامه‌نویس باید یک سری از دستورات هدایتی را ارائه کند. یک روش عمومی به‌صورت زیر ارائه می‌گردد:

| شماره بلاک | نوع بلاک | ورودی ۱ | ورودی ۲ | ورودی ۳ |
|------------|----------|---------|---------|---------|
| ۲۶ | I | ۷ | ۲۶ | ۰ |
| ۱۹ | I | ۷ | ۱۹ | ۰ |
| ۷ | X | ۲۶ | ۱۹ | ۰ |

یعنی بلاک شماره ۲۶ ورودی ترمینال ۱ آن به خروجی ۷ و ورودی ترمینال ۲ آن به خروجی بلاک ۲۶ متصل است و مانند آن.

می‌کند و سپس او باید دستورات اتصال برای چگونگی اتصال بلاکها ارائه نماید. یک زبان بلاک-گرا (مانند CSMP 1130 IBM) تعدادی بلاکهای تابعی استاندارد (بیاده شده بفرم سایرترین‌ها) فراهم می‌آورد. این زبان دارای عناصر عمومی شبیه‌سازی می‌باشد که عبارتند از: انگرالیها (I)، ضرب‌کننده (X)، جمع‌کننده ($+$)، عناصر تابعی خاص مانند محدودکننده (L)، عنصر تاخیر (U)، فاصله (D) و نگهدارنده صفر (Z).

علاوه براین بلاکهای ثابت، زبان تعدادی عنصر برای استفاده در اختیار کاربر قرار می‌دهد که کاربر می‌تواند تابع ارتباطی آن را تعریف کند. برای مثال CSMP 1130 دارای نوع بلاک مختلف و ۵ بلاک قابل تعریف کاربر را فراهم می‌آورد. به عنوان مثال شکل ۱۴-۱ دیاگرام دو بلاک در CSMP 1130 را نشان می‌دهد.



شکل ۱۴-۱ بلاکهای انگرالگیر (I) و ضرب‌کننده (X) (از ارائه‌کننده شماره بلاک است)

بسته به نوع بلاک، این بلاکها در زبان ممکن است صفر، یک، دو یا سه ورودی داشته باشند. برای مثال در شکل ۱۴-۱ انگرالگیر سه ورودی و ضرب‌کننده دو ورودی دارد. یک بلاک ثابت، هیچ ورودی ندارد و خروجی هر بلاک یک تابع مقداری عددی متغیر مستقل (زمان)، ورودی‌ها و پارامترها است. هر یک از عناصر (یا بلاکها) بوسیله یک سمبل مشخص زبان (مانند IX) ارائه می‌گردد که بستگی به نوع آن دارد و شماره n برای اشاره به یک بلاک ضروری بکار می‌رود.

برای شبیه‌سازی سیستم پیوسته مورد نظر اول سیستم باید بوسیله یک بلاک دیاگرام ارائه شود. این بلاک به صورت یک برنامه کامپیوتری (در زبان بلاک‌گرا) بوسیله سه نوع دستورات ارائه می‌گردد: (۱) دستورات ساختاری، دستوراتی که اتصال بلاکها را تعریف می‌کند. (۲) دستورات پارامتریک، مقادیر عددی که با عناصر بلاک، عملیات تابعی بلاکها را نشان می‌دهد، تعریف می‌کند. (۳) دستورات تولید تابع، ورودی/خروجی عناصر مولد تابع را تعریف می‌کند.

مثال ۱-۱۴ به عنوان نمونه یک مثال کلاسیک سیستم-پارازیت بیان می‌شود. ازدحام H

مثال بالا استفاده ساده و برنامه‌نویسی آسان این نوع زبان را نشان می‌دهد. یک فرد می‌تواند یک یا دو ساعت استفاده از این زبان را یاد بگیرد. با یک زبان محاوره‌ای، بلاک-گرا یک کامپیوتر دیجیتال ارزان می‌تواند به عنوان یک کامپیوتر آنالوگ/هائیرید با مرزهای اضافی دقت بیشتر و آزادی از مقیاس‌بندی ظاهر شود. عیب اصلی یک زبان بلاک-گرا انعطاف‌ناپذیری آن است.

سیستم مورد شیبه‌سازی اول باید بصورت بلاک دیاگرام ارائه شود. بنابراین یک زبان بلاک-گرا طبعاً دیاگرامهای بلاکی نوع آنالوگ، ارائه سیستم را استفاده می‌کند و اغلب پیچیدگی نوشتن معادله‌ها را ندارد. برای این منظور مهندسين کثیرل سیستمها بلاکهای ساده ارائه می‌کنند به عنوان مثال دیگر یک زبان قدرتمند مناسب بلاک‌گرا در زیر بیان می‌شود:

Merritt and Millers Own Block Structured Simulation

Language-Unpronounceable Acronym For: MOBSSL-UAF

یک زبان شیبه‌سازی ساخت‌یافته-بلاکی محاوره‌ای است که برای سیستمهای بیروسته بکار می‌رود. مانند وضعیت زبانهای ساخت‌یافته-بلاکی دیگر (از قبیل (1130 CSMP, FACTOLUS, MIDAS از قبیل (1130 CSMP, MOBSL, زبان است که برای یادگیری ساده می‌باشد و نیازی به مهارت و تجربه برنامه‌نویسی ندارد. MOBSL از نسل 1130 CSMP است IBM است اما سراسری و ویژگیهای بیشتری دارد. از قبیل: بلاکهای I/O, پردازنده‌گرایی، یک متغیر ثابت، بلاک حافظه، توابع بسیار و بلاکهای دیجیتال به آنالوگ و آنالوگ به دیجیتال برای محاسبات هائیرید.

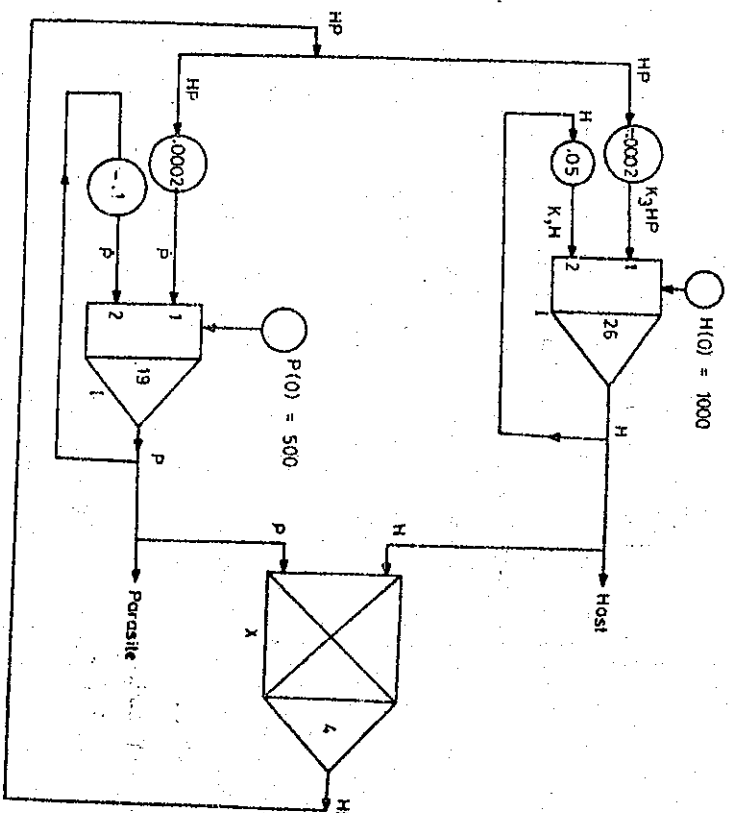
MOBSL-UAF در دانشگاه کالیفرنیا جنوبی در سال ۱۹۶۹ توسعه یافت و روی تعدادی کامپیوتر

از قبیل PDP-11/40, IBM 360/44, CDC 3600, IBM S/370 پیاده‌سازی شد.

۱۴-۲- زبانهای براساس-عبارت

یک زبان شیبه‌سازی پیوسته-بلاک-گرا به علت دارا بودن یک فرهنگ کوچک خاص محدودیت داشت. این زبان به دستورات نوع FORTRAN اجازه استفاده نمی‌داد و بنابراین هیچ ارتباط یا معادله‌ای در سیستم مستقیماً نمی‌توانست بیان شود. سیستم اول باید بصورت بلاک دیاگرام مشابه آنالوگ ارائه شود. بنابراین زبانهای بلاک‌گرا مناسب تکرار آنالوگ و مدل‌هایی است که سیستم پیوسته-بلاک دیاگرام بیان می‌شود و بصورت یک سری از معادلات نمی‌تواند باشد.

حدود سال ۱۹۶۵ بعد از ارائه MIDAS (که بلاک‌گرا بود)، این محدودیت تشخیص داده شد و MIDAS پیشرفته‌تر با نام MIMIC برای کامپیوترهای خانواده IBM 7090 نوشته شد. در MIMIC یک نفر می‌تواند دستورات مشابه FORTRAN بنویسد. (همچنین با ارائه بلاک دیاگرامها) و معادلات منطقی،



شکل ۲-۱۴ بلاک دیاگرام سیستم تغذیه-پارازیت

مقادیر اولیه و مقدار پارامترها به شکل زیر بیان می‌شوند:

| مقادیر اولیه و پارامترها | بلاک | مقدار اولیه | پارامتر ۱ | پارامتر ۲ |
|--------------------------|------|-------------|-----------|-----------|
| | ۲۶ | ۱۰۰۰/۰ | -۰/۰۰۰۲ | ۰/۰۵ |
| | ۱۹ | ۵۰۰/۰ | ۰/۰۰۰۲ | -۰/۱ |

در نهایت باید دوره شیبه‌سازی، توقف، متغیرهایی که باید نمایش داده شوند و چگونه به عنوان خروجی ظاهر گردند (روی چاپگر، پلاتر یا CRT و مانند آن ساخته شود) را مشخص کرد جزئیات بسیاری در نظر گرفته نشده است که اینجا در راهنمای هر زبان معین است.

مزیت اصلی یک زبان بلاک‌گرا سادگی و طبیعی بودن آن است که یک سیستم پیوسته دینامیکی را شیبه‌سازی می‌کند.

الف) روشهای انتقالگیری:

در بیشتر زبانهای براساس-عبارت روشهای مختلف موجودی می‌باشند که کاربر از میان آنها می‌تواند انتخاب کند و شامل هر دو روشهای با اندازه-گام ثابت و روشهای با اندازه-گام متغیر می‌باشند. برای مثال در S360 CSMP پنج روشین با اندازه ثابت موجود است: ثابت رنگ-کو-تایمپسون، هذلولی، مثلثی و دو جمله‌ای آدافر دو روشین با اندازه-گام متغیر نیز موجود است: پنج جمله‌ای Milne or Predictor-Correcter و چهار جمله‌ای رنگ-کو-تا.

گام انتقالگرا^۴ در اتنای اجرای برنامه بطور اتوماتیک تغییر می‌کند تا ناحیه خطای تعیین شده کاربر را تأمین کند. اگر هیچکدام از این ۷ روش نتواند نیازهای S360 CSMP را تأمین کند او می‌تواند یک روشین انتقالگرا^۵ CENTRL استفاده و هر روش دیگر انتقالگرا که می‌خواهد تعریف کند.

برخلاف S360 CSMP بعضی زبانهای کوچکتر فقط یک سابروشین انتقالگرا دارند. برای مثال BEDSOCS از روش انتقالگرا چهار جمله‌ای رنگ-کو-تا-مروسون با اندازه-متغیر استفاده می‌کند. GASP-PLI و DYNAMO یک سابروشین انتقالگرا براساس یک نگارش از روش چهار جمله‌ای رنگ-کو-تا با گام-متغیر دارند و روش انتقالگرا اولتر را استفاده می‌کند.

ب) توانایی اتوماتیک دستورات:

هر برنامه‌نویسی می‌داند که اگر متغیرهای ظاهر شده در سمت راست یک عبارت مقداردهی تعریف نشده باشند (در دستور قبل آن) دستور مقداردهی در برنامه قابل قبول نیست. بطور مشابه برای اطمینان در برنامه شبیه‌سازی ضروری است که دستورات موجود در یک توانایی صحیح قرار گرفته باشند. گرچه در یک برنامه شبیه‌سازی نوشته شده برای یک سیستم پیچیده، این عمل بیشتر برعهده برنامه‌نویس می‌باشد. بنابراین، بیشتر زبانهای شبیه‌سازی براساس-عبارت دارای حالتی هستند که بطور اتوماتیک دستورات ورودی را برای محاسبه مقضی مرتب می‌کنند. لذا کاربر از اعمال دستورات ورودی رها است و نیازی ندارد که نگران قرارگرفتن آنها در توانایی صحیح باشد.

پ) زبان میانجی (واسط):

بیشتر زبانهای براساس-عبارت به عنوان توسعه یافته بعضی زبانهای سطح بالا شناخته می‌شوند. برای مثال BEDSOCS یک توسعه از زبان BASIC است. بطور مشابه DSL90 و CSMP توسعه‌ای از FORTRAN هستند. بنابراین توانایی FORTRAN نزد کاربر S360 CSMP موجود است. عبارات در S360 CSMP می‌تواند با دستورات FORTRAN مخلوط گردد. توابع کتابخانه‌ای استاندارد FORTRAN مانند سینوس، کسینوس، تانژانت و قدرمطلق در دسترس برنامه‌نویس است. یک پیش‌پردازش ابتدا تمامی دستورات غیر FORTRAN (از S360 CSMP) را به FORTRAN تبدیل می‌کند. این دستورات

جبری و دیفرانسیلی (به عنوان ارائه سیستمی که سیموله می‌شود) می‌تواند مستقیماً بصورت دستورات یا عبارات در زبان نوشته شود.

به عنوان مثال سیستم تغذیه-پارازیت بخش ۱۴.۳ که بوسیله معادلات (۱۴.۱) و (۱۴.۲) ارائه شده بیان می‌گردد. سیستمی که سیموله می‌شود از سری دستورات زیر در یک زبان شبیه‌سازی سیستم پیوسته براساس-عبارت بنام BEDSOCS استفاده می‌کند:

ثابت‌ها

10 LET K1 = 0.05

20 LET K2 = 0.10

30 LET K3 = 0.0002

40 LET K4 = 0.0002

وضعیت‌های اولیه

50 LET H = 1000

60 LET P = 500

100 DYNAMIC

150 EQUATIONS

معادلات

160 DER H = K1*H - K3*H*P

170 DER P = -K2*P + K4*H*P

180 DISPLAY H/100, P/100

200 EQUEND

300 DYEND

زبانهای شبیه‌سازی پیوسته مانند BEDSOCS (Bradford Educational Simulation Language For)

Continuous Systems) را زبانهای شبیه‌سازی سیستمهای پیوسته براساس - عبارت یا براساس -

دستورالعمل می‌نامند. زبان براساس-عبارت به ساختار بلاکی آنالوگ نیاز ندارد و مستقیماً معادلات

دیفرانسیل مدل را پیاده‌سازی می‌کند. تعداد زیادی زبانهای براساس-عبارت طراحی و پیاده‌سازی

گردید. ابتدا بخشی از عمومی‌ترین آنها را شرح می‌دهیم و سپس به موردی بر بعضی از زبانهای

شناخته‌شده آنها می‌پردازیم:

جایی که S پراگماتور لایلاس است.

تکیه ساده ملسازی آن توابع انتقال تعریف یک متغیر جدید Y است:

$$\frac{Y(s)}{X(s)} = \frac{1}{s^2 + cs + d}$$

جایی که صورت کسر واحد است اما همان مخرج استفاده شده است. سپس، برای بالاترین جمله مشتق حل می‌کنیم؛

$$s^2 Y = X - cs Y - d Y.$$

این معادل است با دستور فامنه زمانی:

$$\ddot{Y}(t) = X(t) - c \dot{Y}(t) - d Y(t)$$

که بر سبیل دستورات CSMP به صورت زیر مدل می‌شود:

$$S2Y = X - C * SY - D * Y.$$

سپس بر سبیل دو انتگرال متوالی $Y(t)$ را بدست می‌آوریم. از این‌رو

$$Z(s) = s^2 Y + as Y + b Y$$

در نهایت $Z(t)$ را بر سبیل ترکیب با دستور زیر از CSMP بدست آوریم:

$$Z = S2Y + A * SY + B * Y$$

کاربر ممکن است یک ماکرو برای ارائه ارتباطات تابعی آن با نام واحد FILTER تعریف کند. ماکرو جدید در زبان CSMP به صورت زیر تعریف می‌گردد:

```
MACRO Z = FILTER (A, B, C, D, X)
  S2Y = X - C * SY - D * Y
  SY = INTEGRAL(0.0, S2Y)
  Y = INTEGRAL(0.0, SY)
  Z = S2Y + A * SY + B * Y
ENDMAC
```

این ماکرو می‌تواند در شریع برنامه CSMP قرار گیرد و از آن استفاده شود.

(ج) انتخاب‌های ارائه پلاک-گی:

بسازی از زبانهای براساس عبارت و ترکیبهای زبان پلاک-گی را نیز در خود لحاظ می‌کند. یکی کردن دفرع کاربر که یکی سیستم خویش را بصورت معادلات مدل می‌کند و دیگری که سیستمش را

همراه با دستورات FORTRAN (که با دستورات CSMP مخلوط شده است) تشکیل یک برنامه می‌دهد. سپس بر سبیل کامپایلر FORTRAN کامپایل و اجرا می‌شوند.

از مزایای داشتن یک زبان واسط قدرت، انعطاف و سادگی پیاده‌سازی می‌باشد. قیمتی برای این پرداخت می‌شود و زمان پیش پردازش را افزایش می‌دهد زیرا زبان اول باید به زبان واسط ترجمه شود و سپس قبل از اجرا به زبان ماشین تبدیل گردد. درحالی که مثلاً یک برنامه نوشته شده در MMIC (زبانی که کامپایلر برنامه شییه‌سازی را مستقیماً بدون زبان واسطی که ماشین تبدیل می‌کند) زمان ترجمه کثوری از برنامه به زبان DSL90 یا CSMP (S360) دارد.

جنب دیگر استفاده از یک زبان واسط آن است که برنامه‌نویس باید جهت استفاده از یک زبان شییه‌سازی، کارایی لازم در یادگیری زبان واسط را نداشته باشد.

(ت) پیش‌نگاره ورودی و خروجی:

در یک زبان شییه‌سازی سیستم پیوسته براساس عبارت، ورودی از فرم و شکل رها است. کاربر نیاز به فرمت‌دادن به آن ندارد. بطور مشابه، خروجی نیز با یک فرمت ثابت نمایش داده می‌شود (چاپ می‌شود).

فقط باید متغیرهایی که نمایش داده می‌شوند تعریف گردند و متغیرهای مستقل (زمان) را افزایش داد. برای این منظور بعنوان مثال در برنامه BEDSOS برای پارزیت H و P که باید روی صفحه نمایش داده شود را با مقیاس ۱۰۰ در هر واحد ارتفاع صفحه نمایش، با یک دستور DISPLAY H/100,P/100 تعریف می‌کند. این پیش‌نگاره ورودی/خروجی برنامه‌نویس را از کار فرمت دادن آنها رها می‌سازد.

(ث) ماکرو:

یک مزیت مهم موجود در بسازی از زبانهای براساس عبارت توانایی تعریف تابع قوی است که ماکرو نام دارد. این امر به کاربر اجازه می‌دهد که بلاکهای عملیاتی بزرگتر از توابع ابتدایی موجود زبان بسازد و بنابراین یک زیربخشی از یک پلاک دیگرام شییه‌سازی ایجاد کند. ماکرو یک بار تعریف و به دفعات استفاده می‌شود.

با مثال زیر بعنوان یک ارائه، نشان می‌دهیم که این امر در CSMP چگونه استفاده می‌شود. فرض کنید می‌خواهیم یک سیستم کنترل که دارای چندین تابع انتقال است شییه‌سازی کنیم که تمامی آنها دارای شکل زیر هستند:

$$\frac{Z(s)}{X(s)} = \frac{s^2 + as + b}{s^2 + cs + d}$$

CSSL : Continuous System Simulation Language در سال ۱۹۶۷ تعریف شد که ویژگیهای مناسب یک زبان شبیه‌سازی در ارتباط با نواحی کاربردی، روش برنامه‌نویسی و کامپیوترهای دیجیتال، ترکیب گزید. توانایی بسیار بالا برای رفع نیازهای کاربر دارد و دارای کنترل خطا و خطایابی است. انتگرالگیری را می‌تواند با استفاده از چندین روش انجام دهد.

نگارندهای بسیاری از آن ارائه شد که عبارتند از: CSSL III روی CDC 6400 و CSSL IV: XDS روی کامپیوترهای CDC 3600/370 و UNIVAC 1108/1110، DEC 10 و CDC 6600/7600. یک نگارش CSSL بنام RSSL روی CDC 7600 پیاده‌سازی شد. زبانهای HYTRAN و SL-1 نیز نگارندهای CSSL هستند.

DARE : Differential Analyzer Replacement یک سری از زسانهای شبیه‌سازی سیستمهای پیوسته برای پردازش‌های batch و online می‌باشد. یک نگارش آن DAREP توسط وسیله دانشگاه آریزونا توسعه یافت. DAREP روی کامپیوترهای CDC 6000 و CDC 10 ارائه شد. با استثنای یک سری کم‌روشنیهای وابسته ماشین، سیستم بطور کامل با FORTRAN می‌شد. برنامه سازولار است و از روشهای براساس - FORTRAN با سیستم مستقل برای نگارش و پیاده‌سازی فایلها استفاده می‌شود.

کاربر می‌تواند الگوریتمهای انتگرالگیری یا توابع کتابخانه‌ای جدید تعریف کند. معادلات مسئله به شکل یک متن ریاضی می‌تواند داده شود. داده ورودی توسط یک کارتهای داده‌ای رها از فرمت تعیین می‌شود. خروجی‌ها روی چاپگر و پلاتر تعریف گردیده‌اند. روشهای انتگرالگیری آن عبارتند از: رانگ - کوکرامسون، چهارجمله‌ای رانگ - کوکوتا، سه جمله‌ای رانگ - کوکوتا، دو جمله‌ای آدمز، اولتر و دوروش (Gear, Pope) برای حل سیستم‌هایی با معادلات دیفرانسیل. زبان می‌تواند با استفاده از کامپیوتر PDP-11 بطور محاوره‌ای استفاده شود. نگارندهای دیگر DARE روی PDP-11 و روی LSI-11 (میکرو کامپیوتر LSI-11) است.

DYNAMO : DYNAmo (موجود روی CDC 3600, 370, 7090 و UNIVAC 1107, 1108 B5500) زبانی با بیشترین توسعه و مورد استفاده برای شبیه‌سازی علوم اقتصادی و اجتماعی است که در اواخر ۱۹۵۰ در MIT توسعه یافت. از اولین جمله دیفرانسیل برای تقریباً هر مسئله پیوسته مدل شده استفاده می‌کند. سیستمهای دینامیک (پویا) بصورت معادلات سطوح و نرخ و متغیرهای پیوسته ارائه می‌شوند. متغیرهای سطح (پول، جمعیت و مانند آن) وضعیت سیستم در زمان خاص را می‌دهد. جریانها به وسیله معادلات نرخ و متغیرها بیان می‌گردد. این متغیرها و معادلات چگونگی سرعت تغییر سطح را تعریف می‌کند.

براساس یک سری پلاکهای مرتبط مدلسازی شده است، امکان دارد بسیاری از زبانهای براساس عبارت مانند CSSL، CSMP و S360 این قدرت را دارند.

این زبانها از مزایای دیگری نیز برخوردارند. برای مثال BEDSOCS فقط اولین جمله معادلات مشتق را دارد و زبانهای بسیاری جملات بالاتر معادلات نوشته شده را قبول می‌کنند. برای این منظور در یک چنین زبانی

$$\frac{d^3x}{dt^3} = y$$

می‌تواند بطور مستقیم نوشته شود بصورت:

$$\text{DER } 3(x)=Y$$

و نباید آن را به سه معادله مانند زیر تبدیل کنیم:

$$\text{DER}(X)=Z1$$

$$\text{DER}(Z1)=Z2$$

$$\text{DER}(Z2)=Y$$

در این فضای محدود نمی‌توان شرح کامل یک زبان شبیه‌سازی براساس عبارت را ارائه کرد و می‌توان با راهمندی زبان آنها را یافت. حال بعضی ویژگیهای یک تعدادی زبانهای شناخته شده در این سطح را بیان می‌کنیم:

DSL (Digital Simulation Language): یک زبان کلاسیک و شناخته شده زبانهای شبیه‌سازی سیستمهای پیوسته است. این زبان بوسیله IBM توسعه یافت که ابتدا در سال ۱۹۴۶ به عنوان DSL/90 برای کامپیوترهای 7090/7094 و کامپیوترهای مختلف ارائه شد که عبارتند از: DSL/40 (برای IBM 7040)، DSL/44 (برای IBM 360/44)، DSL/1130 (برای IBM 1800)، DSL/1800 (برای IBM 360) و DSL/360 (برای IBM 360). CSMP (SYSTEM/360) یک پیاده‌سازی از DSL برای بعضی کامپیوترهای IBM می‌باشد. DSL یک توسعه از زبان FORTRAN است.

S/360 CSMP: زبان CSMP (S/360 Continuous System Modeling Program) می‌تواند شرح بلاکگرایا براساس عبارت را پشتیبانی کند. با بیان قیل یک پیش‌پردازش با FORTRAN به عنوان زبان واسط دارد. برنامه کاربر می‌تواند شامل دستورات و سابروتینهای FORTRAN باشد. تواناییهای این زبان مواردی است از قبیل: هفت روش انتگرالگیری، بیش از ۵۰ تابع شبیه‌سازی، ماکروها، آشکارسازی و خطایابی عالی، دارای فرمت ورودی/خروجی کم و ابزارهای مختلف که برای یادگیری مشکل نیست ولی گران است.

سیستم های کوچک شبیه سازی انجام داده باشند.

(۲) زبانهای شبیه سازی سیستم گسترده بسیار مسئله گرا است. یک زبان بسیار طبیعی و مناسب

شبیه سازی یک کلاس سیستمهای گسترده ممکن است برای کلاسهای دیگر سیستم ها مناسب نباشد.

(۳) برنامه نویس زبان آشنا با آن و هماهنگی با کلاسهای مسائل طراحی ممکن است در مرکز کامپیوتر وجود نداشته باشد. علاوه بر آشنایی باید تکنیکهای طراحی در زبان بهبود یافته، مستندات بهتری داشته و شبیه سازی عمومی تری یابد.

هر زبان شبیه سازی گسترده باید مفاهیم و دستورات زیر را فراهم کند برای (الف) ارائه وضعیت یک سیستم در یک نقطه از زمان (مدلسازی ثابت)؛ (ب) حرکت یک سیستم از وضعیتی به وضعیت دیگر (مدلسازی پویا) و (ج) انجام موارد عمومی مانند تولید اعداد تصادفی، آتالیز داده ها و تولید گزارش. موارد بخش (ج) در تمامی زبانها یکسان است.

موارد (الف) و (ب) در زبانهای مختلف بطور متفاوت بحث می شود. از این نقطه نظر، زبانهای شبیه سازی سیستم گسترده به سه دسته اصلی تقسیم بندی می گردند:

(۱) زبانهای پیش آمد گرا (Event-oriented)،

(۲) زبانهای فعالیت گرا (Activity-oriented) و

(۳) زبانهای پروسه گرا (Process-oriented) (مشمول بر جریان-تراکنش Transaction).

(۱) زبانهای پیش آمد گرا: در یک زبان پیش آمد گرا هر پیش آمد برسبیل یک وقوع آتی در زبان شبیه سازی ارائه می شود و باید وقتی که یک سری وضعیتهای موجود است برای وقوع زمان بندی گردد. با وقوع یک پیش آمد وضعیت سیستم تغییر می کند. زبانهای این ساختار برای مدل کردن پروسه هایی است که برسبیل تعداد زیادی موجودیت شکل می گیرند. دو زبان شناخته شده این گروه زبانها SIMSCRIPT و GASP هستند.

(۲) زبانهای فعالیت گرا: در یک زبان فعالیت گرا رخدادهای گسترده در پیشبرد زمان بندی نمی شوند. آنها برسبیل یک برنامه که شامل شرح وضعیتهای است که هر فعالیت می تواند انجام گیرد، ایجاد می گردند. این وضعیتها قبل از پیشبرد زمان شبیه سازی موز می شوند و اگر تمامی شرایط وجود داشته باشد، افعال مقتضی انجام می گیرد. بر این اساس زبانهای فعالیت گرا دو جزء اساسی دارند: یک جزء تست و یک جزء عمل.

یک زبان فعالیت گرا برای استفاده باید شرح داده شود اگر مدل مشخصات زیر را دارد: (۱) مدل یک زمان بندی از نوع پیش آمد بعدی یا زمان متغیر دارد، (۲) پروسه ها با درجه محاوره زیاد شبیه سازی می شوند اما دارای تعدادی زیادی موجودیتها با رخداد بی قاعده پیش آمدها نمی باشند و

با استفاده از این دوزنوع بسیاری سیستم ها به عنوان سیستم های دارای فیدبک مدل می شوند.

کامپایلر DYNAMO مدل نوشته شده، بفرم سطح، نرخ و معادلات کمکی، با استفاده تنهای خاص را قبول می کند.

توابع کامپایلر عبارتند از: الف) کنترل خطا، ب) توانی معادلات مطابق مفاهیم دینامیک سیستم، پ) کامپایل برنامه مدل را تعریف می کند و ت) اجرای برنامه خروجی ها را بفرم جدول یا گرافیکی بیان می دارد. زبان دارای بخشهای محاسباتی بفرم آماری یا ریاضی می باشد. برای حل سیستمهای با فیدبک و مدلسازی آنها مناسب است.

DYNAMO III مانند زبانهای دیگر شبیه سازی در طی سالها زیاد توسعه یافت و تمامی قابلیتهای DYNAMO II را به علاوه ساختار آرایه ها دارا می باشد.

MIMIC: با بیان قبلی MIMIC جزو اولین زبانهای شبیه سازی پروسه براساس عبارت می باشد که در سال ۱۹۶۵ برای خانواده IBM 7090 توسعه یافت. هنوز به عنوان یک زبان عمومی برای حل مسائل مهندسی و زبان جاگزینی DYNAMO می باشد. دارای کد روانی است و فقط یک سابروتین انتگرالگیری با روش چهار جمله ای، رنگ-کو تا و با گام متغیر دارد. برنامه MIMIC مستقیما به برنامه زبان ماشین ترجمه می شود. بنابراین MIMIC از نظر زمان پیشبرد ارزش، دارای ارزش می باشد و روی کامپیوترهای سری UNIVAC1107, 1108 و CDC 6000 پیاده سازی شده است.

۱۴.۵- زبانهای شبیه سازی سیستم گسترده

گرچه بسیار برنامه های کوچک و بزرگی برای شبیه سازی سیستم های گسترده وجود دارد که در زبانهای همه منظوره مانند FORTRAN نوشته شده اند، زبانهای خاص شبیه سازی سیستم های گسترده طراحی گردیده و به صورت عمومی در آمده اند. این زبانها دارای امکانات بسیاری هستند که عبارتند از: تولید اتوماتیک جویبارهای اعداد شبه تصادفی برای هر توزیع آماری، جمع آوری اتوماتیک داده ها، تحلیل های آماری و تولید گزارش، تشخیصهای مناسب، پشتیبانی خودکار صف و مانند آن.

به علاوه یک زبان خوب شبیه سازی دارای مدل خاص جهت طراحی آسان می باشد که مدلسازی را ساده تر می سازد و باید راهنمایی برای شرح سیستم های دینامیکی، مفاهیم قدرتمند و رفتار آن ارائه کنند. بنابراین یک زبان شبیه سازی باید بتواند حتی در سطح مدلسازی نیز راهنما برای کمک به کاربر داشته باشد.

سه مسئله مهم در استفاده از این زبانها وجود دارد: (۱) برنامه نویس باید یک زبان جدید یاد بگیرد که ممکن است انگیزه ای برای انجام آن نداشته باشد اگر تمرینات شبیه سازی را روی موارد کم یا

به عنوان یک زبان عمومی با مزایای شبیه‌سازی پیش آمد - گسسته می‌باشد که به علت توان عمومی و مبنای FORTRAN آن، SIMSCRIPT بسیار پیاده‌سازی و توسعه یافت و روی کامپیوترهای زیر ارائه گردید:

سری‌های IBM 7090, 7094, 360, 370 GE 625, 635; CDS 3600, 3800, 6000, 7000; HONEYWELL 600, 6000; UNIVAC 490, 1107, 1108; PHILCO 210, 211, 212; RCA 70, 45, 50

زبان SIMSCRIPT می‌تواند به راحتی یادگیری و در سطح مختلف بکار رود و سطح یک SIMSCRIPT قابل مقایسه با یک زبان الگوریتمیک ساده مانند BASIC است. کاربرد آن دارای حالتی زیر است: داده‌های بدون فرمت و سادگی برنامه‌نویسی، خروجی‌های اختیاری و تبدیل خودکار. برای مثال، دستورات ساده زیر در SIMSCRIPT بیان می‌شود:

```
READ X, Y AND N
PRINT 1 LINE WITH X, Y, XY, N**2 AS FOLLOWS
X = 0.0, Y = 0.0, XY = 0.0, N SQUARED = 0.0
```

دستور اول مقادیر X, Y, N را از کارت پانچ می‌خواند. دستور دوم X, Y, XY, N را انجام و نمایش می‌دهد.

سطح بعدی می‌تواند به عنوان یک زبان FORTRAN-PLUS بیان شود. تواناییهای اضافی در نواحی ساختارهای داده‌ای یافت می‌گردد و زبان اجازه ساختار انتطافی آرایه (آرایه‌های دوبعدی با تعداد مختلف عناصر در هر ردیف) و ساختارهای درختی پیچیده را می‌دهد. هر برنامه شامل برنامه اصلی و ساب‌روتینهایی است که بازگشتی نیز هستند. توانایی‌های ورودی-خروجی مانند FORTRAN است ولی مولد گزارش بسیار خوبی دارد.

سطح سوم به SIMSCRIPT II دستورالعملی شبیه ALGOL (یا PLI) فراهم می‌آورد که سه سطح اول SIMSCRIPT عنوان یک زبان روالی جبری مطرح است.

سطح چهارم SIMSCRIPT II قدرت تعریف و پیاده‌سازی موجودیتها، صفات و سری‌ها را می‌دهد. یک موجودیت یک عنصر برنامه مشابه یک متغیر است. وقتی یک موجودیت ایجاد می‌شود می‌تواند به عنوان تعریف عمومی یک کلاس موجودیتها تفسیر گردد. موجودیتهای خاص دارای مقداری هستند که صفات نام دارند و مبین یک وضعیت خاص موجودیت است. صفات نامگذاری می‌شوند و نه شماره‌گذاری. برای مثال: ممکن است EMPLOYEE به عنوان یک موجودیت تعریف و AGE و SALARY صفات آن باشند. هر EMPLOYEE صفات AGE و SALARY خاص خودش را دارد.

موجودیتها می‌توانند دو نوع باشند: دائمی (برای تمامی زمان اجرای برنامه تعریف می‌شود) و

(۳) وقوع پیش آمدها بوسیله برنامه‌های مرور فعالیت کنترل می‌شود. دو مثال این دسته عبارتند از: CSL (روی کامپیوترهای IBM 7090/94, 1620, 200/2200, IBM 400/1400, 200/2200, IBM 7090/94, 1620) MILITRAN (روی ماشینهای IBM 7090/94).

(۳) زبانهای پروسه-گرا: مزیت کلیدی پروسه‌گرایی آن است که روتین یک پروسه ترکیبی از تعدادی سگمنت برای نمایش توالی فعاليتها می‌باشد. هر سگمنت به عنوان یک برنامه کنترل مستقل وجود دارد. با دریافت کنترل، فقط دستورات ترکیبی سگمنت اجرا می‌شوند و سپس کنترل برگردانده می‌شود. بنابراین مدل به صورت یک سری از اتفاقيه‌ها (پروسه نامیده شده) تعریف می‌شوند و محاورات بین پروسه‌ها بوسیله روتین پروسه انجام می‌شود. بعضی از زبانهای متعلق به این دسته عبارتند از: SIMULA (بطور مختصر بیان شده)، ASPOL (روی سری CDC 6000)، SIMUFOR، SIMULATED (UNIVAC 1107, 1108) SOL و SIMPLE/1، (IBM 360/370).

زبانهای جریان-تراکنش گرا: زبانهای جریان-تراکنش گرا زیر گروهی از زبانهای پروسه-گرا را تشکیل می‌دهند که جریان فعاليتها از طریق بلاکهای تعریف شده خاص، عبور می‌کند. مدل سیستم بوسیله فلوچارت شامل بلاکهای زبان ارائه می‌گردد. برنامه تراکنش‌ها را ایجاد می‌کند، در بلاکها جاری می‌شوند و آنها را از طریق فلوچارت عبور می‌دهند. نوشتن یک برنامه به کشیدن فلوچارت تقلیل می‌یابد و برخلاف زبانهای براساس-دستور بیان شده موارد (۱) و (۲) و (۳) فوق، این زبانها فلوچارت-گرا هستند. بهترین زبان شناخته شده این دسته GPSS است.

قابل توجه است که دسته‌بندی زبانهای شبیه‌سازی سیستم گسسته هرگز مطلق نیست. راههای زیاد دسته‌بندی عبارتند از:

سیستمهای مفسری یا کامپایلری هستند، دستور-گرا یا بلاک-گرا می‌باشند، یا برای چه دسته از مسائل مناسبند. حال شرحی بر سه زبان عمومی شبیه‌سازی سیستم گسسته SIMSCRIPT و GPSS و SIMULA خواهیم داشت و به بیان زبان SLAM نیز می‌پردازیم.

۱۴-۶ SIMSCRIPT

SIMSCRIPT توسط شرکت RAND در اواخر سال ۱۹۶۰ توسعه یافت و اولین برنامه آن در ۱۹۶۲ انجام شد، این زبان دستخوش تغییرات بسیاری شد و بهبود آن منجر به SIMSCRIPT 1.5 گردید. در سال ۱۹۶۸ نگارش کامل SIMSCRIPT II ارائه شد و در سال ۱۹۷۲ نگارش SIMSCRIPT II.5 انتشار یافت. این زبان بسیار شبیه FORTRAN می‌باشد و در حقیقت با زبان FORTRAN به عنوان زبان واسطه پیاده‌سازی شده است.

استفاده می‌شود. SIMSCRIPT II دوازده تابع برای تولید نمودهای شبه تصادفی برای توزیعهای زیر

دارد:

یکتوان، نرمال، پواسون، نمایی، بتا، ارتگ، نرمال لگاریتمی، دو جمله‌ای، گاما، ویبل و توزیع

یکتوان صحیح.

زبان همچنین برای جمع‌آوری اطلاعات آماری از اجرای یک شبیه‌سازی امکاناتی فراهم می‌آورد.

دو دستور ACCUMULATE و TALLY برای جمع‌آوری خودکار و تحلیل منطقی دستورات برنامه در PREAMBLE استفاده می‌شود. یک سابروتین FORTRAN می‌تواند در این زبان موجود باشد و بعضی توانایی‌ها امکان‌سازی خطای برنامه بوسیله کمپایلر را دارد.

۱۴.۷ GPSS - (General Purpose Simulation System)

GPSS یکی از زبانهای قدیمی شبیه‌سازی گسترده می‌باشد که بوسیله Gordon توسعه یافت و در سال ۱۹۶۲ ارائه گردید.

اولین نگارش آن روی کامپوترهای IBM 704, 709, 7090 پیاده‌سازی شده که نگارندهای آن عبارتند از: GPSS II, GPSS v2, GPSS360, GPSS360, GPSS III, (۱۹۷۶), GPSS V و GPSS سری کامل تمامی پیاده‌سازیهای GPSS است و یک برنامه در GPSS360 می‌تواند تحت GPSS اجرا شود و در تمامی این نگارندها مفهوم اساسی ساختار بلاک دیگرامی باقی مانده است.

GPSS برای آنالیزهایی که اساساً برنامه‌نویس کامپوتر نبرند طراحی گردید و مناسب مدل‌سازی ترافیک و سیستمهای صف می‌باشد. یک برنامه‌نویس GPSS برنامه بطریق برنامه‌نویس SIMSCRIPT انجام نمی‌دهد. بجای آن، او یک بلاک دیگرام (شبکه‌ای از بلاکهای متصل) می‌سازد که هر کدام یک تابع شبیه‌سازی - گرافیکال می‌دهند. GPSS ۲۸ نوع مختلف بلاک دارد که می‌توانند به‌فعالت تکرار شوند و هر کدام یک نام دارند کار خاص انجام می‌دهند.

حرکت موجودیتها موقت سیستم تراکتش (Transaction) نام دارد. مثالهای تراکتشها عبارتند از مشتری، پیامها، قطعات ماشین، دوچرخه و مانند آن. بلاکهای نمونه عبارتند از (۱) GENERATE تراکتشها را ایجاد می‌کند، (۲) QUEUE صفی از تراکتشها ایجاد و آمادگی صف را نگه می‌دارد، (۳) TABULATE زمانی را که تراکتش از نقطه زمانی ورود به سیستم شبیه‌سازی شده طی کرده است را جدول‌بندی می‌کند و (۴) TERMINATE از بین‌بردن و جمع‌آوری تراکتشها از سیستم است.

توانایی پیش‌آمد‌ها در سیستم شبیه‌سازی شده بوسیله حرکت تراکتشها از بلاکی به بلاکی دیگر

موقتی (طور دینامیکی که به عنوان محصول برنامه است). موجودیتهای موقتی بطور فیزیکی از طریق دستورات خاص ایجاد و از بین می‌روند. موجودیتهای دائمی صفاتشان را به صورت آریه‌هایی در اختیار دارند. موجودیتها ممکن است به عنوان موقت یا دائمی در نظامنامه تعریف PREAMBLE برنامه شوند.

GPSS در

هر دو موجودیتهای موقتی و دائمی می‌توانند به سری‌های خودشان مبتنی باشند. SIMSCRIPT از اشاره‌گرهایی (Pointers) برای اتصال موجودیتهایی که اعضای سری‌ها هستند، استفاده می‌کند. برای فراموشی پیاده‌سازی این سری‌ها موجود است. ایجاد یک موجودیت، فایل کردن سری، خارج کردن یک موجودیت از یک سری، جستجوی یک سری و غیره. همچنین فراموشی خاص تشخیص یا سری‌های مختلف وجود دارد مانند: اولین ورودی خروجی یا مرتب‌سازی FIFO ورودی اولین خروجی LIFO عناصر سری مطابق مقادیر بعضی از صفات آن.

کار اصلی سطح پیچ نگهداری مدل‌های دینامیک است. یک توشه برای شرح دینامیک سیستمها، کنترل دینامیک آنها، مدل‌سازی آمارها و برای خطایابی و تحلیل مدل دارد. با بیان قبلی، رفتار وابسته با زمان بوسیله پیش‌آمدها مدل می‌شوند که با وضعیت در نقاط گسترده در زمان شبیه‌سازی تغییر می‌کند و بوسیله اجرای روتین یک پیش‌آمد تعیین می‌گردد. زمان شبیه‌سازی بوسیله روتین زمان‌بندی کنترل می‌شود و زمان‌بندی پیش‌آمدها بوسیله سری پیش‌آمدها یا جدول پیش‌آمدها انجام می‌گیرد.

جدول پیش‌آمدها یک صف با عناصر مرتب شده بر اساس زمان وقوع می‌باشد و پیش‌آمدی که در جلوی صف قرار دارد زودتر اجرا می‌شود. یک پیش‌آمد خاص یک موجودیت موقت است که صفاتی مانند زمان پیش‌آمد و نوع پیش‌آمد را دارد. این‌ها مطابق زمان زمان‌بندی شده وقوع، مطابق سری پیش‌آمدها ایجاد و در آن قرار می‌گیرد. هر فعالیت در پیش‌آمد SIMSCRIPT II بوسیله دو پیش‌آمد شروع و پایان ارائه می‌گردد.

"SIMSCRIPT II به یک مدل برای تعریف بصورت پیش‌آمدهای جدا و زمان‌بندی یک به منظور پیش‌آمد ویژه" در سری پیش‌آمدها، نیاز دارد. پیش‌آمدها می‌توانند تولید شوند و بصورت داخلی یا خارجی (از راه بوسیله کارهای داده) زمان‌بندی گردند. در اولین وضعیت ما یک دستور به عنوان مسبب یک پیش‌آمد داخلی را بوسیله پیش‌آمد دیگر می‌نویسیم، برای مثال در دستور:

SCHEDULE A DEPARTURE AT TIME.V+8.5

یک پیش‌آمد با نام DEPARTURE زمان‌بندی می‌شود که در زمان شبیه‌سازی مقدار متغیر TIME.V به علاوه ۸/۵ واحد زمانی رخ دهد. در وضعیت دوم، کارت‌های داده پیش‌آمد را زمان‌بندی می‌کند.

زمان‌بندی پیش‌آمدها عموماً با محاسبه یک زمان تأخیر بوسیله سابروتینهای نمونه‌برداری آماری

تشخیص داده می‌شود. ممکن است تراکنش‌های مختلفی در بلاک دیاگرام وجود داشته باشد اما هر کدام از آنها در هر مقطع از زمان در همان بلاک تعیین می‌شود. یک بلاک می‌تواند دارای یک یا چندین تراکنش همزمان باشد و روی آنها کار کند. معمولاً تراکنش‌ها موجودیهای موقت هستند که تولید می‌شوند، در بین بلاکها حرکت می‌کنند و بوسیله برنامه GPSS جمع‌آوری می‌شوند. بعضی از تراکنش‌ها دائمی هستند و در سیستم شبیه‌سازی در انبای تمامی پیروی اجرای شبیه‌سازی باقی می‌مانند.

بلاکهایی که فقط یک واحد در هر زمان را پشتیبانی می‌کنند تسهیلات (Facilities) نامیده می‌شوند. مثالهایی از آن یک ماشین یا یک فرد است. تراکنش ممکن است یک کار باشد که باید بوسیله ماشین یا فرد انجام شود. نوع بلاکی که همزمان بوسیله یکی یا بیشتر از یک تراکنش می‌تواند استفاده شود یک انباره (Storage) نام دارد که مثال آن گروهی از ماشینها هستند.

GPSS پیشبرد زمان را بوسیله بلاک ADVANCE پشتیبانی می‌کند. وقتی که یک تراکنش به این بلاک وارد می‌شود، یک زمان عمل محاسبه و جهت ایجاد زمان یک بلاک خروجی به زمان جاری اضافه می‌گردد. وقتی که زمان خروج رسید تراکنش را در صورت امکان به بلاک بعدی چارت حرکت می‌دهد. تراکنش‌ها ممکن است صفاتی که برای تصمیم‌گیری منطقی داخل یک بلاک استفاده می‌گردد را متصرف شوند. برنامه GPSS توانایی جمع‌آوری داده‌های آماری در پروسه شبیه‌سازی مانند تعداد تراکنش‌های داخل انباره یا طول یک صف را دارد.

محاسبات ساده ریاضی می‌تواند با استفاده از دستورات متغیری انجام شود. برخلاف SIMSCRIPT در GPSS توابع اساسی ریاضی وجود ندارد (مانند توابع لگاریتمی). GPSS می‌تواند تعدادی متغیر تصادفی ابتدائی تولید نماید. خروجی برنامه‌ها بدون نقاضای مشخص از کاربر اعداد تصادفی به فرم استاندارد را فراهم می‌کند. نوشتن گزارشات جدول‌بندی شده بصورت گرافیکی می‌تواند تولید شود. خطاهای برنامه‌نویسی در انبای کمپایل بخوبی اجرا، آشکار می‌شود و قدرت تولید پیامهای خطا برای کاربر را دارد.

به علت فلوجارت‌گرا بودن GPSS، شاید بسیار سریع‌تر از هر زبان دیگر یادگیری شود. یک برنامه GPSS دارای کد کوتاه‌تر از برنامه‌های مشابه در زبانهای دیگر است. GPSS مناسبترین سیستم برای شرح واحدها بصورت جریانی (عناصر پویا مانند مردم، ماشین‌ها، تلفن و مانند آن) از طریق یک بلاک دیاگرام ثابت می‌باشد. اما اگر مدل تولید بصورت جریان تراکنش‌ها را از طریق بلاکهای شبکه موجود باشد، استفاده از آن بسیار مشکل است. GPSS یک زبان عمومی شبیه‌سازی گسسته مانند SIMSCRIPT نیست.

۱۴.۸ - SIMULA (Simulation Language)

SIMULA توسعه 60 ALGOL است و بنابراین یک زبان برنامه‌نویسی همه منظوره می‌باشد. در سال ۱۹۶۵ روی 360/370 CDC 3000، Burroughs B5500 قرار گرفت. هدف آن ایجاد عملیات شبیه‌سازی بدون از دست دادن مزایای یک زبان همه‌منظوره است. GPSS و SIMSCRIPT دو زبان عمومی شبیه‌سازی گسسته در امریکا و SIMULA بیشتر در اروپا عمومیت یافت. دستورات آن بفرم دستورات ALGOL می‌باشد و لذا کاربر باید دانشی از ALGOL داشته باشد.

ایده اصلی در SIMULA افزودن مفاهیم برنامه‌های مجتمع بنام پروسه (Process) به ALGOL بود که اساساً بصورت موازی عمل کنند. شرح یک پروسه دارای بخش ثابتی است که یک توالی از تعریف صفات می‌باشد و یک بخش دینامیک که توالی از دستورات است و "قانون عمل" نام دارد و بیان رفتار دینامیکی پروسه می‌باشد.

پروسه‌ها عملیاتشان را در گروههایی انجام می‌دهند که به آن "فاز فعال" گویند. یک پروسه ضمن حمل داده‌ها کنش action روی آنها را اجرا می‌نماید. قانون عمل تعریف کنشی است که باید انجام گیرد. قانون عملیات پیش‌آمدهائی هستند که باید در برهه‌ای از زمان شبیه‌سازی روی دهند و همچنین مکانیزمی است که پیش‌آمدهای متعلق به پروسه را فعال و غیرفعال می‌کند. در یک مطالعه شبیه‌سازی ممکن است چند پروسه موردنظر دارای ساختمان داده‌ای و قانون عملیات یکسان باشند ولی در مقادیر صفتهای مرتبط با ساختمان داده‌ها متفاوت هستند. یک چنین گروهی از پروسه‌های یک "دسته پروسه" نام دارد.

در یک برنامه SIMULA تعریف یک دسته پروسه در یک بلاک (block) انجام می‌گیرد که اساس SIMULA و راهی برای تجربه برنامه است. بلاک یک بخش مستقل از یک برنامه می‌باشد، ساختمان داده به عنوان بلاک آغازین است و دستورات بلاک قانون عمل را تعریف می‌کند. سپس بلاک توجیه به بیان شکل ساختمان داده و اعمال کنش‌ها (actions) است. وقتی یک بلاک اجرا شود، شبه‌بلاک تولید می‌گردد.

SIMULA توانایی آن را دارد که چندین بلاک را تولید کند و ممکن است در یک زمان موجود و با یکدیگر در حال محاوره باشند. برای هر کدام از این بلاکها و ساختمان داده‌ای آنها یک نام در نظر می‌گیرد و آن را به کمک توانایی بالا در پردازش لیست‌ها انجام می‌دهد. برنامه SIMULA شامل مجموعه بلاکهایی است که اجرای کامل آن به عنوان توالی فازهای فعالیتهای مرتبط به شبه‌بلاکهای مختلف، می‌باشد.

۱۴.۱- فاکتورهای انتخاب یک زبان شبیه‌سازی سیستم گسسته

فرض کنید که برای شبیه‌سازی از یک زبان همه‌منظوره مانند C یا FORTRAN استفاده نکنیم، از چه زبان شبیه‌سازی برای شبیه‌سازی یک سیستم گسسته استفاده شود؟ با بیان قبیل نگرش مختلف زمان‌بندی پیش‌آمدها را دیدیم. اساس آنها مفاهیم کلیدی است و برای فرموله کردن پروسه‌ها در مدل شبیه‌سازی، اجرای آن و برنامه‌نویسی روند متفاوتی دارند. بطور واضح انتخاب مناسب یک زبان شبیه‌سازی بسیار مسئله‌گر است و دارای پارامترهای گوناگونی می‌باشد. ابتدا محورهای مقایسه زبانها ارائه و سپس سه زبان SIMULA, SIMSCRIPT و GPSS را مقایسه می‌کنیم.

(۱) قدرت برنامه‌نویسی و عمومیت زبان: زبان باید قادر باشد که محدودیت‌های وسیع سیستم‌های کاربردی را بررسی کند. باید توانایی ارائه الگوریتمیک قوی، بخوبی موازای گرایش زبانهای شبیه‌سازی را داشته باشد که عبارتند از: محاسبات عددی، مفاهیم شبیه‌سازی برای بیان سیستم توانایی پشتیبانی مجموعه‌ها، لیست‌ها و صفها؛ مولدهای تصادفی و جمع‌آوری داده‌ها و نمایش معطوف داده‌ها. گرچه کاربرد عمومی (توانایی) یک زبان بستگی به توانایی کاربر برای قرار دادن فضای مسئله در سازمان زبان دارد.

(۲) راحتی استفاده: راحتی استفاده شامل ترجمه طبیعی سیستم تحت مطالعه به یک برنامه کامپیوتر مرتبط، تقلیل در تلاشهای خطایابی، مستندسازی مناسب و طراحی آزمایشات مناسب است. یکی از مسائل مناسب درجه بالای توانایی خواندن، راحتی تعریف و تقلیل نیازهای مستندسازی است.

تقلیل تلاش خطایابی تعریف زبان به خوبی کیفیت پیاده‌سازی آن ارتباط دارد. اغلب نیاز داریم که تمامی خطاهای برنامه‌نویسی در مرحله کمپایل ظاهر شود و در مرحله اجرای سیستم نیز پیامهای مناسب پوشش داده شوند. خطایابی خوب در پیاده‌سازی یک زبان شبیه‌سازی مفید و مهم است زیرا این امر مسیب توسعه استفاده از پروژانش لیست‌ها و ساختمان داده‌ای پویا می‌باشد.

طرح آزمایشات باید در ترکیبات اجرایی کامپیوتر و گزارشات عددی نتایج قابل انعطاف توأم باشد. اینجا یک تعادل بین تقاضاهای زبان و راحتی یادگیری و استفاده این مفاهیم وجود دارد. راحتی استفاده زبان بر سبیل رهایی فرمت ورودی و فرمهای پیش‌فرض خروجی است که محدوده وسیعی از مسائل را پشتیبانی می‌کند.

(۳) کارایی ماشین: معمولاً هزینه توسعه یک مدل و کدنویسی بخش مهمی از هزینه شبیه‌سازی است. با اینحال کارایی زمان اجرا مهم است و در جایی که اجراهای متتابع بسیاری نیاز دارد یا برنامه شبیه‌سازی در پربودهای زیاد شبیه‌سازی استفاده می‌گردد. کارایی ماشین وابستگی به سطح

67 SIMULA ۱۰ زیر برنامه اعداد تصادفی برای طرح نمونه‌های مختلف با توزیعهای آماری شامل: یکتراخت، نمائی، نرمال، پواسون و ازانگ دارد. گرچه جمع‌آوری خودکار آماری در SIMULA نیست (که در GPSS وجود دارد)، یک زیر برنامه "acorn" موجود است که جمع‌آوری زمانی مناسب مقدار هر متغیر را انجام می‌دهد. معمولاً برنامه‌نویس SIMULA هر نیاز الگوریتمیک آماری را بر مبنای زبان

ALGOL 60 تعریف می‌کند.

۱۴.۹- SLAM (System for language Alternative Modeling)

SLAM یک زبان سطح بالایی شبیه‌سازی بر اساس FORTRAN است و اجازه زمان‌بندی پیش‌آمدها یا گرایش محاوره پروسه‌ها یا ترکیبی از هر دو نگرش را می‌دهد. زمان‌بندی پیش‌آمدها در SLAM کاملاً مشابه GASP می‌باشد و گرایش محاوره پروسه‌ها در SLAM بسیار مشابه با GPSS است. در اینجا بیشتر به بخش گرایش محاوره پروسه‌ها در SLAM می‌پردازیم. این زبان توسط Pritsker و Regden در سال ۱۹۷۹ توسعه یافت.

برای استفاده از SLAM باید محاوره پروسه‌ها، شبیه‌ساز یک شبکه شامل گره‌ها و انشعاب‌ها که پروسه‌های سیستم را ارائه می‌کنند، می‌سازد. آشایی که در سیستم جریان دارند موجودیت می‌نامند (موجودیت در SLAM شبیه تراکشن در GPSS). صلازدهن یک پروسه توالی پیش‌آمدها و فعالیتها است که با یک موجودیت جریان یافته مواجه نمی‌شود. یک مدل شبکه SLAM سیستمی است از تمامی مسیرهایی که یک موجودیت از آن عبور می‌کند. برای اجرای مدل، شبیه‌ساز مستقیماً شبکه را به دستورات کامپیوتر تبدیل می‌کند و به پروژانده SLAM می‌دهد.

SLAM بطور خودکار الگوریتم پیش‌رو زمان/زمان‌بندی پیش‌آمدها را پشتیبانی می‌کند، عملیات مجموعه مانند افزایش یا حذف موجودیتها را پشتیبانی و جمع‌آوری آمارها و تولید نمونه‌های تصادفی را انجام می‌دهد. در SLAM مجموعه‌ها، فایلها نامیده می‌شوند. با پشتیبانی خودکار فایل، SLAM به راحتی می‌تواند بروش FIFO یا روش موردنظر صفها را مدیریت کند. برخلاف GPSS، SLAM دارای توابع مولد تصادفی پیش ساخته جهت محدوده وسیعی از توزیعهای آماری می‌باشد. یک شبکه SLAM شامل گره‌ها (nodes) و انشعابها (branches) است. یک انشعاب گذر زمان را ارائه می‌کند که این ارائه یک فعالیت می‌باشد. به علاوه یک انشعاب ممکن است تعداد محدودی سروس دهنده را معرفی نماید. گره‌ها برای ارائه پیش‌آمد زروود (گره CREATE)، تاخیر یا انتظار مشروط (گره QUEUE)، پیش‌آمد خروج (گره TERMINATE) و اعمال دیگر سیستم است. ویژگیهای بیشتر آن در جدول ۱۴-۱ آمده است.

SIMULA

(الف) دارای قدرت الگوریتمیک قابل مقایسه با ALGOL،
 (ب) جمع آوری آمارهای داده ها در رفتار سیستم،
 (پ) سهولت ورود-خروجی قابل مقایسه با SIMSCRIPT،
 (ت) امکان توسعه زبان جهت ساخت زبان مسئله گرای خاص،
 (ث) SIMULA یک زبان پیچیده است اما سادگی استفاده دارد،
 (ج) سهولت خطایابی بسیار خوب،
 (چ) توانایی خواندن و ساختار دهی خوب برنامه،
 (ح) امکان طراحی آزمایشی قابل مقایسه با SIMSCRIPT و
 (خ) کارایی ماشین کمپایلر 67 SIMULA بخوبی کمپایلر SIMSCRIPT
 علاوه بر این سه زبان شبیه سازی گسترده تعدادی دیگر از زبانهای شبیه سازی وجود دارند که بسیار استفاده می شوند. یکی از آنها GASP (General Activity Simulation Program) است که سال ۱۹۶۰ در آمریکا توسعه یافت و پیاده گردید. در اواخر سال ۱۹۶۰، GASP II توسط KIVIAT و PRITSKER طراحی و پیاده سازی شد. زبان GASP IV می تواند شبیه سازی هر دو سیستم پیوسته و گستره را پشتیبانی می کند (۱۹۷۰). GASP II روی کامپیوترهای مختلف پیاده سازی شد از قبیل: GE 225,415، IBM 1130 و GASP. CDC 3400 شامل مجموعه وسیعی از ساب روتینهای FORTRAN است که به عنوان دستورات برنامه نویسی سطح بالا عمل می کنند. برای استفاده از GASP II و GASP IV یک فرد باید برنامه نویسی FORTRAN را بخوبی بشناسد. زبان دیگر گستره CSL (Control and Simulation) نام دارد که یک زبان شبیه سازی سیستم ترکیبی است. زبان دیگر گستره CSL می تواند به عنوان language، یک زبان براساس FORTRAN می باشد که در سال ۱۹۶۱-۶۲ توسعه یافت. این زبان مانند 67 SIMULA در اروپا بسیار گسترش یافت و به وسیله IBM پشتیبانی گردید. CSL می تواند به عنوان زبان فعالیت-گرا دسته بندی شود.

سومین زبان SIMPLI است که زبان شبیه سازی براساس PL/I می باشد و توسط شرکت IBM در سال (۱۹۷۲) توسعه و پیاده سازی گردید. روی IBM System/360 تعریف شد و توسعه p/I با امکان شبیه سازی گسترده شامل پردازنده لیست ها است.

شکل ۳-۱۴ بطور مختصر دسته بندی زبانهای شبیه سازی را نشان داده و در جدول ۱-۱۴ مقایسه ای بین زبانهای FORTRAN، GASP، IL5، SIMSCRIPT و GPSS V، SLAM انجام گرفته است.

شبیه سازی

پیاده سازی کمپایلر دارد که براساس ساختار و محتوای زبان است. دولاک اندازه گیری کارایی ماشین زمان اجرا و حافظه مورد نیاز می باشد که هر دو در اثنای پیش پردازش و زمان اجرا مهم هستند.

(۴) در دسترس بودن: این یک نگرش عملی به هر زبان کامپیوتری است که زبان باید موجود باشد و توسط بسیاری از کامپیوترها برای استفاده پشتیبانی گردد. پس نیاز به آن دارد که زبان مستقل از ماشین باشد، پیاده سازی پیش پردازنده های آن با ماشینهای مختلف باید سازگار و لیست برنامه از ماشینی به ماشین دیگر قابل نقل و انتقال باشد.

مشکل است که یک مقایسه بی تعصب از زبانهای موجود ارائه کنیم و یک دلیل آن است که افراد در پیاده سازی مهارتهای متفاوتی دارند و دلیل دوم آن که مستندسازی موضوعی بوسیله زبان یا ساختار کامپیوتر فراهم می شود. در بخش زیر به بررسی سه زبان مهم و شرح موارد مربوطه می پردازیم:

SIMSCRIPT

(الف) زبان شبیه سازی همه منظوره مستقل از ماشین،

(ب) توانایی الگوریتمیک قابل مقایسه با ALGOL یا PL/I،

(پ) مفاهیم شبیه سازی نسبتاً کم و بسیار عمومی،

(ت) جمع آوری عالی داده ها،

(ث) موارد ورودی و خروجی خوب،

(ج) حفاظت (آتشکارسازی خطا) ضعیف،

(چ) انعطاف خوب طراحی آزمایشی،

(ح) کارایی بالای ماشین و

(خ) یادگیری سخت تر از GPSS.

GPSS

(الف) مناسب مسائل صف،

(ب) سهولت محاسباتی ضعیف،

(پ) ورودی-خروجی غیر قابل انعطاف،

(ت) عدم امکان توسعه زبان،

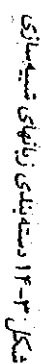
(ث) راحتی یادگیری و استفاده،

(ج) سهولت خطایابی خوب و

(چ) کارایی پایین ماشین علت سیستم مفسری بودن GPSS.

مدلسازی مقدماتی SLAM

برای مدل‌سازی سیستم‌های بسیار بزرگ، در صورت امکان آن را به سیستم‌های کوچک‌تر تقسیم می‌کنیم و با انتخاب مدل‌های مستقل زیر سیستم‌ها می‌توان رفتار کلی سیستم را مدل‌سازی کرد. این امر مستلزم مدل ذهنی واضح از مسئله است و این مدل ذهنی نیاز به تعریف دقیق مسئله دارد. مدل ذهنی



| زبان | | | | |
|----------|----------|-----------|----------|------------|
| SLAM | GPSSV | SIMSCRIPT | GASP | (c)FORTRAN |
| عالی | عالی | خوب | خوب | خوب |
| عالی | عالی | خوب | متوسط | ضعیف |
| همه | صاف | همه | همه | بیج |
| بله | خیر | بله | بله | خیر |
| بله | بله | بله | خیر | خیر |
| بله | خیر | بله | بله | خیر |
| بله | خیر | بله | عالی | ضعیف |
| خوب | متوسط | خوب | خوب | ضعیف |
| عالی | متوسط | عالی | عالی | ضعیف |
| خوب | ضعیف | خوب | خوب | متوسط |
| خوب | متوسط | خوب | خوب | متوسط |
| خوب | ضعیف | خوب | خوب | عالی |
| خیلی خوب | خیلی خوب | متوسط | خیلی خوب | خیلی خوب |
| خوب | عالی | خوب | خوب | ضعیف |
| متوسط | کم | زیاد | کم | کم |

● سهولت یادگیری
● سهولت متهوری کردن مسئله
● وابستگی جزیی سیستمها

● دید مدام سازی
● زمانبندی پیش آمدنها
● محاسبه پورده ها
● سیستم پیوسته

● پشتیبانی
● نمودار داری تصادفی پیش ساخته
● توانایی جمع آوری امار
● توانایی پردازش لیست
● راحتی گزارش استاندارد
● راحتی طراحی گزارش خاص
● اهداف خطایابی

● زمان اجرای کامپیوتر
● مستند سازی یادگیری زبان و ارجاع
● خود مستند سازی کند
● قیمت

جدول ۱-۱۴ مقایسه چندین زبان شبیه‌سازی و زیاتهای همه‌منتظره

شالوده‌آسانی ارائه رفتار سیستم و مدل‌سازی کامپیوتری را می‌سازد. با ترسیم مدل ذهنی سیستم موردنظر در قالب موجودیتهایی تعریف می‌شود که از یک شبکه متصل از گره‌ها و شاخه‌ها (گرافیکی) عبور می‌کنند. لذا قدمای مهم آن عبارتند از:

تعیین عناصری از سیستم به عنوان موجودیت مانند: مشتری، ماشین اطلاعات، برگه‌های سفارش و غیره (سبب تغییر در سیستم)؛

تعیین متغیرهای وضعیت سیستم

و تعریف موضوعاتی که سبب تغییر متغیرهای وضعیت است.

مثلاً در سیستم بازرسی رادیوها، وضعیت سیستم با دو متغیر "تعداد رادیوهای منتظر بازرسی" و "تعداد بازرسی‌های مشغول کار" ارائه می‌شود و رادیو به عنوان موجودیت با عبور از شبکه بازرسی موجب تغییر وضعیت تعداد بازرسی و تعداد رادیوهای منتظر می‌گردد. البته انتخاب موجودیت ممکن است همیشه به این وضوح و شفافیت نباشد و یا برای یک موجودیت واقعی در سیستم موجودیتهای گوناگون تعریف شود. می‌توان موجودیت را در یک شبکه به صورت زیر تعریف نمود:

"هر موضوع مبین تغییرات در سیستم را موجودیت نامند که اغلب در اثر حرکت موجودیتها تغییرات صورت می‌گیرند"

به ازای عبور موجودیتها از عناصر شبکه فعالیتهای متفاوتی صورت می‌گیرد. ترکیب فعالیتهای گوناگون سیستم در یک شبکه لازم است. این شبکه پروسه‌های مختلفی که یک موجودیت انجام می‌دهد یا بر روی آن صورت می‌گیرد را نشان می‌دهد. ابتدا این مسئله به فرم گرافیکی (مدل گرافیکی) ترسیم می‌گردد و سپس به برنامه SLAM تبدیل می‌شود. اگر سیستم به نرم‌افزارهای طراحی مدلهای گرافیکی مجهز باشد که آن ابزار برنامه شبیه‌سازی را تولید نماید، مدل‌سازی گرافیکی اولین و آخرین گام شبیه‌سازی می‌شود.

در یک سیستم صف:

مشتریان به سیستم وارد می‌شوند،

انتظار می‌کشند،

سرویس می‌گیرند و

در نهایت از سیستم خارج می‌شوند.

این توالی از پیش‌آمدها و فعالیتهای انجام یافته را پروسه نامند. موجودیتها باید از طریق این پروسه‌ها جریان یابند و صفات آنها دستخوش تغییرات لازم گردد. موجودیتهای سیستم ممکن است در طول مسیر شبکه بعضی از صفاتشان (یا کل آن) تغییر کند ولی وجه تمایز موجودیتها همین صفات

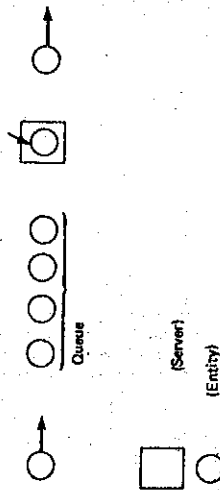
می‌باشد. مثلاً زمان ورود هر کدام از موجودیتها به عنوان یک صفت متمایز کننده قابل طرح است. در این پروسه، سرویس دهنده‌ها برای انجام سرویس روی موجودیتها را به عنوان منابع سیستم تعریف می‌کنند. برای مدل‌سازی منابع، انتظار برای منابع، دریافت منابع، فعالیت در قبال دریافت منابع و تداوم جریان موجودیتها از گره‌ها و شاخه‌ها استفاده می‌کنیم. لذا می‌توان بیان کرد که شبکه SLAM ارائه پروسه‌ها و روند حرکت موجودیتها در آنها است.

۱۵.۱- مدل SLAM

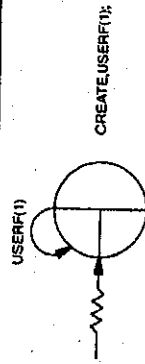
در گذشته بیان گردید که مدل ارائه یک سیستم است، در قالبهای گوناگون انجام می‌گیرد و اغلب دارای اشکال گرافیکی است. مسئله ابتدا جهت تمرکز و بررسی بهتر به مدل گرافیکی تبدیل می‌گردد. مدلهای ماشینیتهای متناهی و شبکه‌های پتری، ارائه آنها در قالب گرافیک و روابط به عنوان مدلهای کارآ معرفی شد. در این فصل مدل SLAM جهت مدل‌سازی مسائل و شبیه‌سازی آن معرفی می‌شود. ابتدا به عناصر مدل، مدل‌سازی و برنامه شبیه‌سازی SLAM II بطور مقدماتی می‌پردازیم و سپس به تدریج آنرا کامل می‌کنیم.

۱۵.۲- عناصر ابتدایی مدل SLAM

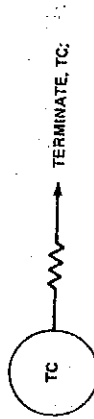
زبان SLAM مسائل مربوط به صف را به راحتی مدل‌سازی و حل می‌نماید. عناصر ابتدایی مدل SLAM در راستای مدل‌سازی صف بیان می‌شود. یک سیستم دارای بخشهای زیر است:



این مدل صف را می‌توان یک سیستم تعبیر قطعات الکتریکی فرض کرد که قطعات سیستم در مدل صف همان مشتری تلقی می‌شود. قطعات برای دریافت سرویس تعبیر به سرویس دهنده وارد می‌شوند و اگر سرویس دهنده مشغول کار باشد، صف ایجاد می‌گردد. مدل گرافیکی SLAM سیستم صف در شکل ۱۵-۱ نشان داده شده است:



۱۵.۲.۲ - گروه جمع کننده (قطع کننده) موجودیتها TERMINATE



شکل ۱۵-۳ سمبل گرافیکی گروه TERMINATE و دستور SLAM آن

از طریق این گروه، موجودیتها از سیستم جمع آوری می شوند که پارامترهای آن به شرح زیر است:

JBL: برچسب نام گروه است. نام دلخواه است و می توان ذکر نکرد.

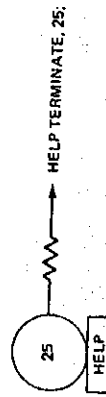
TC: شمارنده تعداد موجودیتهای دریافتی جهت قطع برنامه شبیه سازی است. چنانچه ذکر نشود این گروه فقط یک چاه یا جمع کننده است ولی اگر تعداد آن مشخص شود پس از ورود TC مشتری، برنامه

شبیه سازی را قطع می کند.

در واقع این گروه موجودیتها را از بین می برد می تواند روی قطع برنامه شبیه سازی نیز اثر بگذارد.

مثال ۱۵-۵

گروه HELP پس از جمع آوری و از بین بردن ۲۵ موجودیت برنامه شبیه سازی را قطع می کند.



مثال ۱۵-۶ گروه قطع جمع آوری و نابودی موجودیتها را انجام می دهد و اثری بر قطع برنامه ندارد.



مثال ۱۵-۷ این گروه نیز فقط وظیفه انهدام نهادها را بطور نامحدود برعهده دارد.

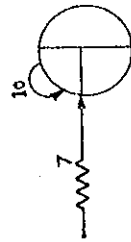


شیوه تولید موجودیت: اگر زمان تولید اولین موجودیت TF باشد، دومی در زمان TF+TBC، سومی در TF+2TBC و ... می آیند

$$TF, TF+TBC, \dots, (TF+(MC-1)TBC)$$

زمان TBC می تواند به صورت ثابت یا تصادفی فرض شود. اگر ثابت فرض شود تولیدات به فواصل زمانی ثابت TBC تولید می شود ولی اگر تصادفی باشد، موجودیت بعدی در زمان TBC بعد از تولید فعلی است.

$$ATRB(MA) = \text{زمان ایجاد موجودیت}$$

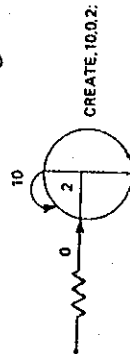


مثال ۱۵-۱

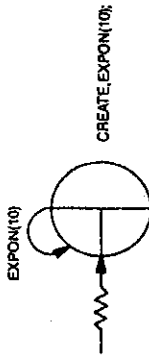
یعنی اولین تولید در زمان ۷ شبیه سازی باشد و بعدی به فواصل زمانی ۱۰ واحد تولید شوند یعنی: 7, 17, 27, 37, ...

تولید مولد نامحدود است و زمان ورود هر کدام در مشخصه شماره ۲ آنها نسبت می گردد. به یک شاخه خروجی هدایت می شوند و فقط می توانند از یک شاخه خارج شوند حتی اگر چند شاخه از آن خارج شود. اگر M بیشتر تعداد از شاخه ها باشد از تمامی آنها این موجودیتها عبور می کند. یعنی کپی موجودیت از آنها عبور می کند.

مثال ۱۵-۲ زمان تولیدات موجودیتها ۰ و ۱۰ می باشد.



مثال ۱۵-۳ خروجی دارای توزیع نمایی با میانگین ۱۰ است و زمان تولید اولین موجودیت چون ذکر نشده صفر است و محدودیت تولید ندارد.



مثال ۱۵-۴ خروجی دارای تولید تعریف شده طبق نظر کاربر است که از تابع USERF پیروی می کند و باید نوشته شود.



مثال ۹-۱۵ صف با گنجایش نامحدود و مقدار اولیه صفر یا صف خالی.



مثال ۱۰-۱۵ صف خالی با ظرفیت ۴ و ذخیره در فایل ۱.



مثال ۱۱-۱۵ مانند مثال ۹-۱۵ و ذخیره در فایل ۱.

آیا اگر صف پر باشد، موجودیتهای ورودی از بین می‌روند یا نه؟ پاسخ این سؤال بستگی به انتخاب سیاست مورد نظر برای این رفتار است. که در این رابطه سه گونه رفتار داریم:

- ۱- موجودیتهای ورودی از بین نبروند (پیش‌فرض در زبان SLAM).
 - ۲- موجودیتهای ورودی به محل دیگری هدایت شوند^(۱). در واقع مرز و مانعی قابل نشتند داریم.
 - ۳- تا خالی شدن صف یا خروج حداقل یک نفر کلیه فعالیت‌های مستثنی به صف متوقف شود.^(۲)
- رفتارهای فوق را در مدل می‌توان تعیین کرد. چنانچه صف به صورت قبل مطرح شود (پیش‌فرض)، موجودیتهای ورودی را در مدل می‌توان تعیین کرد. چنانچه صف به صورت BALK یا Block تعریف شود پیش‌فرض چنین خواهد بود.
- اگر بخواهیم موجودیت ورودی به صف پر را به‌گه دیگری هدایت کنیم به این امر در مدل SLAM

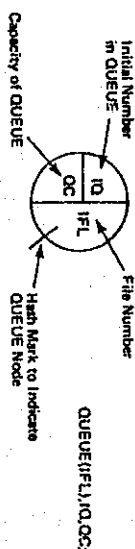
QUEUE(IFT), IOQC,BALK(NBL),SLBL

-----> BALK (NBL)

شکل ۵-۱۵ سیمبل گرافیکی SLAM و دستور BALK

مثال ۸-۱۵ مانند مثال ۵-۱۵ است.

۱۵.۲.۳- گروه صف QUEUE



شکل ۴-۱۰ سیمبل گرافیکی گروه QUEUE و دستور SLAM آن

گروه صف، محلی برای انتظار موجودیتهای (مشتریان) منتظر سروس است. سروس‌دهنده‌ها جهت سروس، مشتریان را از صف انتخاب می‌کنند. اگر حداقل یک سروس دهنده یکبار باشد، فوراً سروس دادن را آغاز می‌کند. در این مدل بین پایان سروس موجودیت و شروع سروس موجودیت بعدی زمانی در نظر نمی‌گیریم.

تقدم و تأخر سروس‌گیری بستگی به سیاست صف دارد. این تقدم خارج از مدل تعریف می‌گردد که می‌تواند براساس زمانی دود به صف باشد یا براساس هر کدام از خانه‌های بردار مشخصه اولویت‌بندی انجام گیرد. (آمارهایی برای صف مانند طول صف، متوسط زمان انتظار و ... به صورت اتوماتیک می‌تواند ثبت شود) مانند:

FIFO: اولین ورود برای سروس می‌شود.

LIFO: آخرین ورود برای سروس می‌شود.

HIGH(K): انتخاب براساس بیشترین مقدار مشخصه Kم بردار صفات برای سروس است.

LOW(K): انتخاب براساس کمترین مقدار مشخصه Kم بردار صفات برای سروس است.

پاراوترهای صف:

IO تعداد موجودیتهای اولیه صف در زمان شروع شییه‌سازی، پیش‌فرض آن صفر است.

IOC ظرفیت صف، حداکثر تعداد موجودیتهایی که در یک صف قرار می‌گیرند، پیش‌فرض نامحدود است.

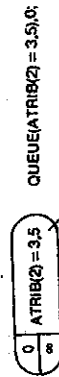
IFL شماره فایل ذخیره موجودیتهای منتظر و مشخصات آنها است.

مثال ۱۵-۱۵ صفی با گنجایش ۲ و مقدار اولیه ۱ با نام QUE1

QUE1 QUEUE(3), 1, 2, BLOCK;

شماره فایل صف می تواند به عنوان صفت ثبت شود، مانند: $ATTRIB(I)=J, K$ یعنی شماره صف I شماره فایل را دارد که این شماره از J تا K مجاز است.

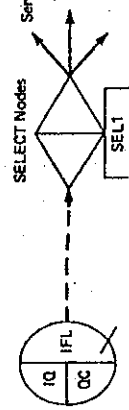
معادل سه صف است صف ۳ و ۴ و ۵:



Select (Select node label)

BALK (Nobe lable)

SEL 1 (تصمیم گیری) است.



مثال ۱۵-۱۶

QUE1 QUEUE(3), 1, 2, BLOCK, TYPE1;



۱۵.۲.۴ - شاخه فعالیت

DUR, PROB or COND → ACTIVITY (N/A, DUR, PROB or COND, NLBL, ID)

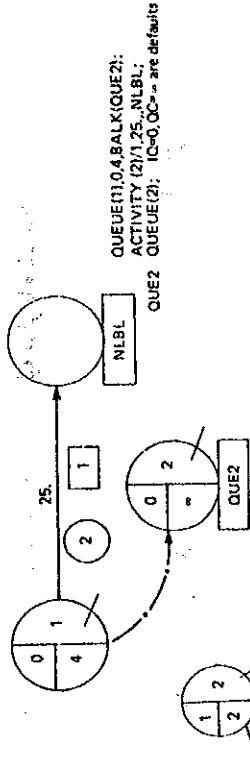


شکل ۱۵-۷ سمبل گرافیکی ACTIVITY و دستور SLAM آن

این شاخه ها برای مدل سازی فعالیتها مورد استفاده قرار می گیرند که این فعالیت فقط توقف مدت زمانی به عنوان تأخیر است و فعالیت خاصی صورت روی مشتری نمی گیرد. فعالیت سرورس برای گذر و جریان موجودیهای همزمان محدودیت بوجود نمی آورد و بطور همزمان به تعداد سرورس دهنده ها موجودیت عبور داده می شود.

هر فعالیت از یک گره شروع و به گره دیگری پایان می یابد. پس شروع و پایان آن با گره است. اگر فعالیتها مختلفی از یک گره خارج شوند ممکن است نیاز به تصمیم گیری یا ایجاد شرط یا احتمال داشته باشیم. این تعریف ها در شاخه فعالیت قابل تعیین است. در این مدل هر فعالیت را با یک شماره می شناسیم و می توان از آنها این که شماره دارند گزارش خروجی دریافت کرد. با این شماره می توان

مثال ۱۵-۱۲ در صورت پر بودن صف ۱: موجودیهای ورودی به صف ۲ هدایت می شوند.



مثال ۱۵-۱۳ در صورت پر بودن صف به گره جمع آوری ارسال می شوند.

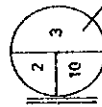
اگر بخواهیم که موجودیهای ورودی در صورت پر بودن صف از بین نروند و به محل دیگری نیز هدایت نشوند. در واقع از ورود آنها جلوگیری شود، به آنها اجازه حرکت ندهند و کلیه فعالیتها متوقف به آن صف متوقف شوند. این امر از طریق تعریف BLOCK به صورت زیر امکان پذیر است و ممکن است فقط فعالیتها سرورسی بلوک شوند:

QUEUE(IFL), IQ, OC, BLOCK, SLBL;

BLOCK

شکل ۱۵-۶ سمبل گرافیکی BLOCK و دستور SLAM آن

مثال ۱۵-۱۴ صف بلوک شده با ظرفیت ۱۰ و مقدار اولیه ۲: QUE1 QUEUE(3), 2, 10, BLOCK;



در حالتی که صف پر باشد، موجودیها و فعالیتها منتهی به گره مدتی به حال تعلیق در می آیند و تا مدتی که فضای خالی ایجاد نشود این وضعیت تداوم دارد. به محض ایجاد فضای خالی، موجودیت می تواند وارد صف شود و فعالیتها مربوطه برای بقیه رها می گردند.

STOPA برابر مقدار 1 تعیین شود. در این لحظه زمان فعالیت پایان می‌یابد و موجودیتها از این شاخه خارج می‌گردند. خارج می‌گردند حتی اگر در فعالیتهای دیگری نیز موجودیتهایی با کد 1 وارد شده باشند، آن فعالیتهای نیز پایان می‌پذیرند.

با وجود تغییر NTC با بردار مشخصه موجودیت یا عدد تصادفی، موجودیتهای با کد متفاوت به جریان می‌افتند.

ACTIVITY, STOPA(ATTRIB(1));

موجودیتهایی خارج می‌شوند که کد آنها در خانه اول بردار مشخصه شان باشد. با مقداری $STOPA=1$ موجودیتهای نوع 1 از فعالیت آزاد می‌شود و اگر مقداری $STOPA=2$ باشد موجودیتهای نوع 2 از فعالیت آزاد می‌شوند و غیره.

تعریف احتمال برای شاخه فعالیت:

احتمال به صورت عدد ثابت بین صفر و یک، متغیر یا عبارت می‌تواند تعیین شود. لازم نیست

مجموع احتمالات فعالیتهایی که از یک گره خارج می‌شوند برابر 1 باشد:

$MNO(2)/100$ یا $ATTRIB(2)$ یا $xx(3)$ یا 0.7

تعریف شرط برای شاخه فعالیت:

قابل تعریف شرط به این صورت است.

Value . Operator . Value

Value می‌تواند عدد ثابت، متغیر زبان یا متغیر تصادفی باشد و Operator یک عملگر منطقی استاندارد زبان فورتن به شرح زیر است:

| | |
|------|-----------------|
| LT < | کوچکتر |
| LE ≤ | کوچکتر یا مساوی |
| EQ = | مساوی |
| NE ≠ | نامساوی |
| GT > | بزرگتر |
| GE ≥ | بزرگتر یا مساوی |

مثال ۱۷-۱۵) TNOW متغیر زمان جاری شیه‌سازی (یعنی زمان جاری شیه‌سازی بزرگتر یا مساوی

TNOW, GE, 100.0

است. ۱۰۰

وضعیت فعالیت و موجودیتهای عبوری آنها را کنترل کرد.

برای هر فعالیت به جای یک شماره می‌توان چندین شماره تعریف کرد که این امر از طریق گره ATTRIB امکان دارد.

پارامترهای شاخه فعالیت عبارتند از:

DUR: دوره یا مدت زمان فعالیت.

PROB: احتمال انتخاب فعالیت برای عبور از آن.

COND: شرط انتخاب فعالیت برای عبور از آن.

IN: حداکثر تعداد موجودیتهایی که همزمان از فعالیت عبور می‌کند یا تعداد سروس دهنده‌های موازی.

NA: شماره فعالیت.

NLBL: برچسب گره‌بندی آن.

زمان فعالیت: زمان فعالیت می‌تواند توسط عدد ثابت، متغیر یا توزیعی آماری دلخواه تعیین گردد. اگر این مقدار متغیر شود مدل مقدار صفر را برای آن فرض می‌کند. ضمناً زمان یک فعالیت می‌تواند تابعی از عملکرد یک سیستم یا بخشی از آن باشد. مثلاً انجام یک فعالیت بستگی به خروج موجودیتهای گره خاصی داشته باشد. یعنی در واقع زمان پایان سروس بستگی به خروج موجودیتی از جای دیگر دارد که این بخش همان توانایی مدل در جستجوی فعالیتهای است. که در این حالت زمان فعالیت به صورت REL(NLBL) تعریف می‌شود. یعنی زمانی که یک خروجی از گره NLBL خارج می‌گردد، لحظه پایان سروس این فعالیت خواهد بود (مدلسازی به روش جستجوی فعالیتهای).

مدت زمان یک فعالیت می‌تواند به مقادیر تعیین شده در گره ASSIGN جهت پارامتر خاص STOPA) مرتبط شود. اگر زمان یک فعالیت به صورت STOPA(NTC) تعریف گردد، هر موجودیت (مشتری) با کد مشخص NTC وارد فعالیت می‌شود (یک عدد صحیح است). NTC می‌تواند یک عدد ثابت یا متغیر با زمان باشد (در صورت اعشاری بودن به نزدیکترین عدد صحیح گرد می‌شود). اگر در گره ASSIGN مقدار STOPA برابر کد یک مشتری یا یک دسته از موجودیتهای تعیین شود (این موجودیتهای قبلاً در فعالیت یا فعالیتهایی که زمان آن با STOPA (NTC) تعیین شده است، وارد شده‌اند) زمان فعالیت برای آن موجودیتهای پایان می‌یابد.

مثلاً چنانچه دستور به صورت زیر باشد:

ACTIVITY, STOPA(1);

یعنی تمامی موجودیتهایی که وارد این فعالیت می‌گردند منتظر بمانند تا در گره ASSIGN مقدار متغیر

خانه اول بردار مشخصه مشتری فعلی کوچکتر از یک عدد تصادفی جویبار شماره ۲ است. $ATTRIB(1).LT.DRAND(2)$

موجودیهای فایل شماره ۷ مساوی ۱۰ است. $NNQ(7).EQ.10$

این شرط ترکیبی است از اپراتورهای AND و OR می‌توان استفاده کرد.

در عبارات AND بر OR اولویت دارد:

توجه: شرطهای با منطق پیچیده که نیاز به استفاده از پرانتز دارد، مجاز نیست. برای این عمل می‌توان از تابع $USERF$ که توسط کاربر تعیین می‌گردد استفاده کرد و شرطها را در آن تعریف نمود:

$USERF(1).GE.0.$

۱۵.۲.۵ - گروه مقداردهی ASSIGN



شکل ۸-۱۵: سمبل گرافیکی گروه ASSIGN و دستور SLAM آن

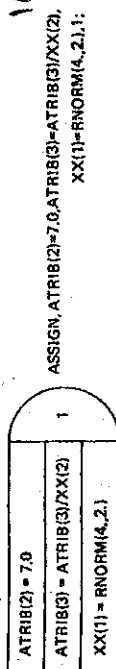
از گروه ASSIGN برای مقداردهی متغیرهای زبان، بردار مشخصه موجودیت (مشتری) یا مشخصه‌های دیگر استفاده می‌شود. این متغیرها می‌توانند دارای تغییرات پیوسته و یا گسسته باشند. متغیرهایی که در این گروه مقداردهی می‌شوند عبارتند از: $XX(I)$ ، $SS(I)$ ، $DD(I)$ ، II ، $ATTRIB(I)$ مقداردهی متغیر STOPA.

متغیرهای مقداردهی شده می‌توانند به عنوان زمان فعالیت، شرط فعالیت و مانند آن در برنامه‌نویسی استفاده گردند. در شبه‌سازیهای ترکیبی (پیوسته و گسسته) تعیین مقدار متغیرها به عنوان قسمتی از تغییرات مدل تلقی می‌شود. مقداری که برای متغیرها تعیین می‌کنند می‌توانند یک عدد ثابت، متغیرهای مذکور، متغیرهای وضعیت در شبکه، متغیر زمان جاری شبه‌سازی، تابع توزیع

احتمال یا یک مقدار از تابع $USERF$ باشد.

در سمت راست آن یک عبارت ریاضی می‌باشد که تا ۱۰ عمل محاسباتی برای آن پیش‌بینی شده است. عبارت ریاضی شامل اعداد ثابت و متغیرهای جداگانه ۱-۱۵ و ۲-۱۵ است. در محاسبات اعمال ضرب و تقسیم بر جمع و تفریق اولویت دارد.

مثال ۱۵-۱۷



می‌توان با متغیر کمکی II نیز مقداردهی کرد:

$ASSIGN, II = ATTRIB(2), ATTRIB(3) = XX(II);$
 $ASSIGN, II = UNIFORM(0.,10.), ATTRIB(4) = II;$

مقداردهی $XX(I)$ را از متغیر تصادفی نمایی با میانگین $ATTRIB(I)$ و جویبار اعداد تصادفی شماره ۲ انجام می‌دهیم:

$ASSIGN, XX(1) = EXPON(ATTRIB(1), 2);$

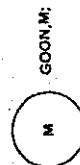
زمانی که مشتری از گروه ASSIGN عبور کند و بخوایم خانه شماره ۳ بردار مشخصه آنرا مقداردهی کنیم دستور زیر بکار می‌رود:

$ASSIGN, ATTRIB(3) = 4;$

۱۵.۲.۶ - گروه اشعاب GOON

اگر بعد از فعالیتی موجودیت بخواهد از آن خارج شود، سه‌وال این است که از کدام شاخه‌ها خارج شود؟ از چند شاخه خارج شود؟ می‌توان گره‌ای برای این امر در SLAM یافت این گره به صورت زیر است: (شکل ۹-۱۵)

شاخه‌ها می‌توانند دارای شرط یا احتمال باشند. مقدار M تعداد شاخه‌های خروجی همزمان را نشان می‌دهد. اگر M از N بزرگتر باشد، مشخصاتان بطور موازی از تمامی شاخه‌های عبور می‌کنند.



شکل ۱۵-۹: سمبل گرافیکی گروه GOON و دستور SLAM آن (به شاخه خروجی).

اگر مقدار M کوچکتر از N باشد، از N تا N اولین شاخه‌ها که شرط آنها صادق باشد عبور می‌کنند.

۱۵.۲- گروه جمع آوری آمار COLCT

| TYPE | IO,H | M |
|-------|-------------------------------|---|
| COLCT | TYPE, ID, NCLE, FLOW, HWID, M | |

شکل ۱۵-۱۰ سمبل گرانیکی گروه COLCT و دستور SLAM آن

از این گروه برای جمع آوری مشاهدات آماری از متغیرهای برنامه استفاده می شود. از ۵ نوع از متغیرهای (type) برنامه جمع آورهای آماری صورت می گیرد که عبارتند از:

۱- FIRST (زمان اولین موجودیت ورودی به گروه): از هر بار اجرای مدل یک مقدار جمع آوری می شود.

۲- ALL (زمان ورود کلیه مشتریان): از زمان ورود کلیه مشتریان مختلف ورودی به گروه جمع آوری آماری، متوسط گیری و سایر اطلاعات محاسبه می شود.

۳- BETWEEN (زمان بین ورود مشتری): زمان ورود اولین مشتری به عنوان مرجع انتخاب می شود و از زمانهای بین ورود مشتریان بعدی جمع آوری آمارها صورت می گیرد.

۴- INT (NATR) (آمارهای فاصله زمانی): جمع آوری آمارهای مربوط به مشاهداتی از اختلاف زمانی ورود مشتری با مقدار مشخصه به شماره NATR است یعنی از زمان TNOW-ATRIB (NATR) جمع آوری اطلاعات صورت می پذیرد.

۵- متغیرهای زبان SLAM جمع آوری آمارها از مقدار متغیر زبان SLAM هر بار که مشتری وارد گروه شده انجام می گیرد.

گروه GOCT با جمع آوری آماری از ۵ نوع متغیرهای فوق متوسط گیری، انحراف معیار و حتی هسته گرام ارائه می کند. شرح پارامترهای این گروه متناقضاتی نیست.

برای ترسیم هسته گرام تعداد خانه های هسته گرام، حد پایین اولین خانه و طول هر خانه مشخص می گردد:

تعداد خانه های هسته گرام = NCLE

حد پایین اولین خانه هسته گرام = FLOW

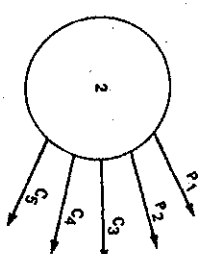
عرض هر خانه هسته گرام = HWID

تعداد سلولهای هسته گرام برابر (NCLE+2) است زیرا:

(-∞, ∞, FLOW), (FLOW, FLOW + HWID), ..., (FLOW + HWID × NCLE, ∞)

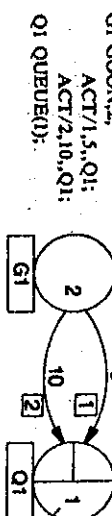
مثال: NCLE=5, FLOW=0 و HWID=10 که محاسبات آماری و جمع آوری آنها در یکی از فواصل

مثلاً اگر مقدار M برابر ۲ باشد و ۵ شاخه موجود داشته باشیم، ۲ تا از اولین شاخه ها که شرایط عبور مشتریان آنها صادق باشد انتخاب می گردند.



در شکل فوق هم احتمالات (P_i) و هم شرطهای (Q_i) برای انشعاب داریم. هر مشتری که وارد این گروه می شود بدون تغییر از آن عبور می کند. حداقل اگر از ۲ شاخه بدون تغییر عبور می کند که ابتدا احتمالات P_i و P_i و پس شرطهای Q_i و Q_i بررسی می گردند و به ترتیب ۲ تا می که ابتدا شرط آنها صادق است انتخاب می شوند.

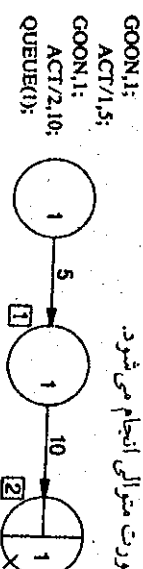
MODEL OF 2 ACTIVITIES IN PARALLEL



مثال ۱۵-۱۸ (شاخه موازی)

به صورت موازی از ۲ شاخه عبور می کند به صف Q₁ دو موجودیت از ۲ شاخه با مشخصه های یکسان وارد می شوند.

MODEL OF 2 ACTIVITIES IN SERIES



مثال ۱۵-۱۹ (شاخه سریال)

از یک شاخه عبور داده و به صورت متوالی انجام می شود.

| متغیر | تعریف |
|------------|---|
| ATTRIB(1) | خانه II بر دار مشخصه جاری |
| II | متغیر شناور عدد صحیح که II به عنوان اندیس یا آرگومان تعریف می شود |
| TNOW | زمان حال شبیه سازی |
| SS(I) | متغیر وضعیت I |
| DD(I) | مشق متغیر SS(I) |
| XX(I) | سیستم یا بردار شناور global |
| ARRAY(I,Y) | سیستم آرایه شناور (global) |

جدول ۱۵-۱ متغیرهای زبان SLAM

در مدل سازی پیوسته از متغیرهای SS(I) و DD(I) برای تعریف معادلات دیفرانسیل یا تغییرات پیوسته مدل استفاده می شود که متغیر D(I) مشتق متغیر SS(I) است.

متغیر TNOW حاوی زمان جاری شبیه سازی می باشد و مقدار آن با ساعت شبیه سازی افزایش می یابد و مدل ساز از طریق این متغیر با ساعت شبیه سازی مرتبط می شود.

متغیرهای دیگر نیز وجود دارند که با استفاده از آنها می توان حالت های مختلف سیستم را داشت که به آنها متغیرهای حالت نیز می گویند. مانند: NNRS و NNQ و NNGAT و NNCNT و NNACT که مقادیر آنها بطور خودکار توسط SLAM بروز می آیند. حالت های از قبیل: تعداد مشتریان موجود یک فایل، وضعیت فایلهای، حالت منابع و دروازه های و مانند آن می باشد. مثلاً NNCNT(2) یعنی تعداد مشتریانی که از فعالیت شماره 2 (در لحظه جاری شبیه سازی TNOW) عبور کرده اند.

توابع:

توابع بسیاری در زبان SLAM وجود دارد و ضمناً کاربر نیز می تواند توابع مورد نظر خود را بداند تعریف و شماره گذاری نماید:

USERF (N)

پارامتر N شماره تابع می باشد که به زبان فرترن برنامه نویسی آن انجام می گیرد.

تابع تابع (VALUE, X, Y, XVALUE) تابع GGIBLN (IRX, IRY, XVALUE) تابع جستجو نام دارد. این تابع مقدار Y متناظر با X را برمی گرداند. IRX و IRY شماره سطرهایی از ماتریس ARRAY هستند.

شبیه سازی

۴۳۶

زیر قرار خواهند گرفت و با شمارش عناصر آنها در فواصل زیر، هیستوگرام مربوطه ترسیم می گردد: (50,50), (40,40), (30,30), (20,20), (10,10), (0,0), (-50,0).

اگر تعداد سلولهای هیستوگرام مشخص نشود ترسیم آن صورت نمی گیرد. شرح پارامترهای گره:

ID: نامی که برای گزارش خروجی انتخاب می شود، حداکثر ۱۶ کاراکتر القبا عددی است که با یک حرف القبا آغاز می گردد.

II: سمبل هیستوگرام است که در حقیقت سه فیلد HLOW و HNU و HNUD دارد.

M: خروجی به M شاخه ارسال می شود.

TYPE: همان 5 نوع متغیرهایی است که بیان گردید.

مثال ۱۹-۱۵ جمع آوری آمارها از زمان ورود مشتری و بدون هیستوگرام جمع آوری گردد عنوان مورد نظر گزارش PROJECT COMP است.

FIRST PROJECT COMP COLCT, FIRST, PROJECT COMP;

مثال ۲۰-۱۵ جمع آوری آمار از مدت زمانی است که مشتری در سیستم می باشد. زمان ورود آنها به سیستم در ATRIB(3) ذخیره شده است. گزارش با عنوان خروجی SYSTEM TIME است.

INT(3) SYSTEM TIME 1 COLCT, INT(3), SYSTEM TIME, J;

مثال ۲۱-۱۵ از متغیر آزاد XX(2) جمع آوری آماری صورت می گیرد و در خروجی نام SAFETY STOCK دارد. هیستوگرام خروجی با ۲۰ سلول، خانه با طول ۵ و خانه اول (۱۰، ۵۰) می باشد. خروجی آن از دو شاخه عبور می کند.

XX(2) SAFETY STOCK, 20/10/5 2 COLCT, XX(2), SAFETY STOCK, 20/10/5, 2;

۱۵.۲.۸ - متغیرهای تصادفی و توابع زبان SLAM

متغیرها:

بعضی متغیرها در جدول زیر شرح داده شده اند:

| Variable/Function | Definition |
|-------------------------|--|
| NNACT(I) | Number of active entities in activity I at current time |
| NNCNT(I) | The number of entities that have completed activity I |
| NNCMTG(ATE) | Status of gate ATE at current time : 0 = open;
1 = closed |
| NNQ(I) | Number of entities in file I at current time |
| NNSCRES) | Current number of units of resource type RES available |
| NNUSERRES) | Current number of units of resource type RES in use |
| USERFN) | A value obtained from the user-written function USERF with user function number N (See Chapter 9) |
| GGTBLN(RX,IRX,XVALUE) | A value corresponding to XVALUE obtained from a table lookup with the Kth point in the table defined by (ARRAY(IRX,K),ARRAY(IREX,K)) |
| BETA(THETA,PHI,IS) | A sample from a beta distribution with parameters THETA and PHI using random number stream IS |
| DRAND(IS) | A pseudo-random number obtained from random number stream IS |
| DRORBN(CI,V,IS) | A sample from a probability mass function where the cumulative probabilities are in row IC of ARRAY and the corresponding sample values are in row IV of ARRAY using random number stream IS |
| ERLNG(EMN,XK,IS) | A sample from an Erlang distribution which is the sum of XK exponential samples each with mean EMN using random number stream IS |
| EXPON(XMN,IS) | A sample from an exponential distribution with mean XMN using random number stream IS |
| GAMA(BETA,ALPHA,IS) | A sample from a gamma distribution with parameters BETA and ALPHA using random number stream IS |
| NPSSN(XMN,IS) | A sample from a Poisson distribution with mean XMN using random number stream IS |
| RNORM(XMN,STD,IS) | A sample from a normal distribution with mean XMN and standard deviation STD using random number stream IS |
| RLOGN(XMN,STD,IS) | A sample from a lognormal distribution with mean XMN and standard deviation STD using random number stream IS |
| TRIAG(XLO,XMODE,XHI,IS) | A sample from a triangular distribution in the interval XLO to XHI with mode XMODE using random number stream IS |
| UNFRM(XLO,UHI,IS) | A sample from a uniform distribution in the interval XLO to UHI using random number stream IS |
| WEIBL(BETA,ALPHA,IS) | A sample from a Weibull distribution with scale parameter BETA and shape parameter ALPHA using random |

جدول ۲-۱۵ متغیرها و توابع SLAM

توزیع یکپارچه: منبیم و ماکزیم تولید یکپارچه ULO و UHI است:

$$\text{UNFRM}(XLO, UHI, IS) \\ ULO = \mu - \sqrt{3}\sigma \\ UHI = \mu + \sqrt{3}\sigma$$

توزیع ویبل: پارامتر مقیاس و پارامتر شکل BETA و ALPHA هستند:

$$\text{WEIBL}(BETA, ALPHA, IS)$$

توزیع ارلنگ: با پارامترهای EMN و XK عبارتست از:

$$\text{ERLNG}(EMN, XK, IS) \\ EMN = d^2/\mu \quad XK = \mu/EMN$$

توزیع گاما: با پارامترهای ALPHA و BETA می شود:

$$\text{GAMA}(BETA, ALPHA, IS) \\ BETA = \frac{\sigma^2}{\mu} \quad ALPHA = \frac{\mu}{BETA}$$

مثال:

| K | ARRAY (3, k) | ARRAY (5, k) |
|---|--------------|--------------|
| 1 | 100 | 10 |
| 2 | 200 | 20 |
| 3 | 300 | 15 |
| 4 | 400 | 10 |
| 5 | 1000 | 0 |

فرض $IRX=3$ و $IRY=5$ باشد، مقدار Y در $ARRAY(5,1)$ و متناظر مقدار X در $ARRAY(3,1)$ می باشد.

XVALUE مقدار متغیر مستقل X را که تابع بدین مقدار متغیر Y متناظر آن است، تعیین می کند.

چنانچه XVALUE خارج از محدوده جدول مقدار $GGTBLN(3,5,400)$ را برمی گرداند.

باشد، اگر از ۱۰۰ کمتر باشند (حد پایین یعنی) ۱۰ را برمی گرداند و اگر از ۱۰۰۰ بیشتر باشد (حد بالا یعنی) ۰ را برمی گرداند.

توابع متغیرهای تصادفی و معادلات آنها در فصول ابتدائی شرح داده شد اکنون توابع SLAM آنها را بیان می کنیم.

تابع توزیع تصادفی بواسن، نرمال، نرمال لگاریتمی دارای پارامترهای میانگین و انحراف معیار است که در SLAM به عنوان میانگین μ و STD به عنوان انحراف معیار می باشد.

$$\text{PNORM}(XMN, STD, IS)$$

مانند:

$$\text{RLOGN}(XMN, STD, IS)$$

$$\text{NPSSN}(XMN, IS)$$

توزیع منطقی: منبیم، ماکزیم و مداتبع XLO و XMODE، است.

$$\text{TRIAG}(XLO, XMODE, XHI, IS)$$

و جمع آوری آمارهای مورد نظر را می توان با دستور زیر ارائه داد:

COLCT, PROC-TIME;

مثال ۱۵-۲۳

EQUIVALENCE / XX (1), INVENTORY / XX (2), REORDER / UNFRM (4, 6),

REVIEW TIME ;

می توان در دستور زیر به جای متغیرهای SLAM نام های فوق را قرار داد:

ACTIVITY, REVIEW TIME, INVENTORY . LE . REORD-PT;

REVIEW TIME برای تابع تصادفی UNFRM(4,6) استفاده می شود.

۱۵.۴ دستور آرایه ARRAY

این دستور برای تعیین مقدار یک سطر از یک ماتریس بکار می رود (INITIALIZE). تعداد عناصر یک سطر ARRAY می تواند تغییر کند و از این رو جدول به یک جدول قدیمی اشاره می کند.

شکل دستور:

ARRAY (IROW, NELEMENTS) / initial values / repeats;

IROW تعریف یک عدد ثابت صحیح برای تعیین سطر آرایه است.

NELEMENTS تعداد عناصر این سطر و initial value ثابتی که برای سطرهای سطر در نظر

گرفته ایم است.

مثال ۱۵-۲۴

ARRAY (2, 4) / 5, 4, 2, 7, 3;

ARRAY (2, 1) = 5

ARRAY (2, 2) = 4

ARRAY (2, 3) = 2

ARRAY (2, 4) = 73

مثال ۱۵-۲۵ اگر ATRIB(1) هر موجودیت به عنوان نوع JOB تعریف شود، ATRIB(2) به عنوان گام بعدی JOB و ATRIB(3) شماره ماشین برای گام بعدی JOB باشد، جدول ARRAY مقدار آن

شماره ماشین به عنوان آرایه است که هر ردیف آن با یک نوع JOB و ستون آن با گام JOB می باشد.

سپس

یک شماره ماشین به مشخصه ۳ براساس نوع JOB و مقدار گام فعلی JOB می دهد.

اگر از دستور EQUIVALENCE استفاده کنیم دستور خوانا تر می شود:

توزیع بتا: با پارامترهای THETA و PHI عبارتست از:

$$\mu = \frac{\mu_B - BMIN}{BMAX - BMIN}$$

تولید نمونه در محدوده ۰ تا ۱ است.

BMAX و BMIN نقاط انتهایی توزیع بتا می باشند.

$$\delta^2 = \frac{\delta^2_B}{BMAX - BMIN}$$

پس آرگومان تابع می شود:

$$TETA = \frac{\mu}{\delta^2} (1 - \mu) - \mu$$

$$PHI = TETA \left(\frac{1 - \mu}{\mu} \right)$$

به عنوان مثال در محدوده BMIN تا BMAX و BETAT و BETAT معادله زیر باید استفاده شود:

$$BETA = BETA(THETA, PHI) * (BMAX - BMIN) + BMIN$$

در زبان SLAM علاوه بر توابع پیش ساخته، کاربر می تواند با تابع توزیع مورد نظر خودش تصادفی DPROBN(IC, IV, IS)

دلخواه تولید نماید: مقدار احتمالات تجمعی، در ردیف IC آرایه ضرایب و مقدار نمونه متناظر در ردیف IV از آرایه ARRAY شده است.

استفاده از این تابع شبیه استفاده از جدول جستجوی GGTLN است.

تولید اعداد شبه تصادفی از جویبار اعداد تصادفی IS

DRAND(IS)

۱۵.۳ دستور معادل (برابری) (EQUIVALENCE)

این دستور جهت استفاده از نامهای مورد نظر (باسمی) و واضح طبق علاقه کاربر برای متغیرهای زبان می باشد:

تکرار / نام مورد نظر و متغیر زبان SLAM

متغیرهایی که در این دستور قرار می گیرند: II, XX, ARRAY, DD, SS, ATRIB, متغیرهای

تصادفی SLAM و یک مقدار ثابت. نامهایی که کاربر می تواند انتخاب نماید حداکثر ۱۲ کاراکتر با شروع یک حرف الفبا است.

مثال ۱۵-۲۲ دستور زیر نام معادل PROC-TIME را برای ATRIB(1) قرار می دهد:

EQUIVALENCE / ATRIB (1), PROC . TIME ;

نام متغیر در گزارش خروجی (انتخابی کاربر) حداکثر ۶۱ حرف.

ID نام متغیر در گزارش خروجی (انتخابی کاربر) حداکثر ۶۱ حرف.
ENTRY/FILE, ATRIB(1), ATRIB(2), ..., ATRIB(MATR)/repeats; دستور

برای تعیین موجودیت اولیه فایلهاست.

شماره فایل.

ATRIB(1): مقادیر بردار مشخصه اولین ورودی به فایل (برای بعدی ها باید تکرار شود).

از برای توجه به شروع دیگری از مشخصات جدید ورودی.

PRIORITY/FILE, ranking/repeats;

تعیین اولویت خروج مشتریان یک فایل (مانند صفها) یا همان سیاست قابل است.

دستور PRI

شماره فایل

(LVEF(N) و HVEF(N), LIFO, FIFO) شامل

اولی: با مشخصه N که بیشترین است.

اولی: با مشخصه N که کمترین است.

MONTR, option, TRST, TSEC, variables;

دستور MONTR

CLEAR option و CLEAR TRACE یعنی آمارهای زمان TRST در COLET منظور نشود.

یعنی (در صورت Trace) آمارها از زمان TRST تا TSEC می باشد.

TRACE: لیستی از وقایعی که باید چاپ شود از زمان TRST شروع و تا زمان TSEC خاتمه می یابد.

CLEAR: برای جمع آوری آمارها در یک دوره گذراست.

دستور FIN

FIN: انتهای دستورات ورودی (SLAM) می آید.

دستور SIMULATE

SIM: انتهای دستورات ورودی برای یک اجرا است.

دستور SIM برای یک اجرا اگر FIN آخرین دستور باشد نیاز نیست.

۱۵.۶ مدل سازی با مدل SLAM

مثال ۱۵-۲۶ (صف و در سروس دهنده موازی) - یک بانک با دو کارمند به مشتریان سروس

می دهد. زمان سروس هر کارمند برای مشتریان بین ۱۲ واحد زمانی است. مشتریان در صورت

مشغول بودن کارمندان به صف انتظار می روند تا یکی از دو کارمند بی کار شود. حداکثر گنجایش صف

انتظار مشتریان بانک ۱۰ نفر است. ورود مشتریان از تابع زمانی با میانگین ۱۰ پیروی می کنند. زمان

EQUIVALENCE / ATRIB(1), JOB TYPE/

ATRIB(2), JOB STEP/

ATRIB(3), MACHINE;

MACHINE = ARRAY (JOB TYPE, JOB STEP)

۱۵.۵ - دستورات ورودی

تعیین دستورات کنترل (دستورات ورودی) از قدم های مهم است. این دستورات اطلاعات لازم از قبل قابل ها، مشخصه های هر موجودیت، متغیرهای آزاد، مقدار اولیه صف ها، تقدم با سیاست صف و غیره را برای شبیه ساز فراهم می کند.

دستورات: ENTRY, TIMST, INITIALIZE, MONTR, PRIORITY, FINAL, LIMIT, GENERATE

(سه حرف اول هر دسته برای SLAM کلیات می کنند).

خلاصه دستورات:

GEN, NAME, PROJECT, MONTH/DAY/YEAR, NNRRNS;

دستور GEN

نام مدل ساز ذکر می شود.

PROJECT نام پروژه شبیه سازی می باشد.

MONTH/DAY/YEAR تعیین تاریخ سال/روز/ماه است.

NNRRNS تعداد دفعات شبیه سازی (اجراء RUN) است.

دستور LIMIT

تعداد فایل ها.

MATR تعداد مشخصه های هر مشتری.

MENTRY تعداد مشتریان فعال در شبکه.

دستور INTLC

برای تعیین مقداردهی متغیرهای زبان II SLAM است.

دستور INIT

زمان شروع شبیه سازی.

TTFIN زمان پایان شبیه سازی.

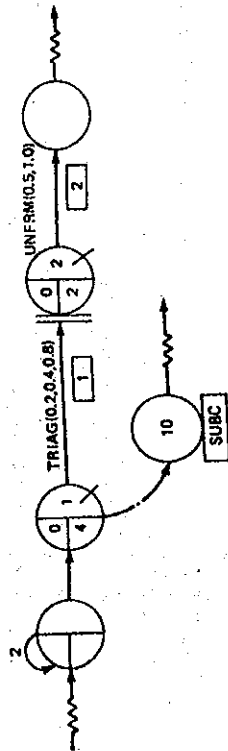
دستور TIMST

دستور جمع آوری آمارهای پیوسته از یک متغیر تصادفی است.

VAR جمع آوری آمار از VAR.

مثال ۱۵-۲۸ یک شرکت عمل تعمیر و نگهداری قطعاتی را انجام می دهد. در هر دو واحد زمانی یک قطعه نیاز به تعمیرات دارد. زمان تعمیرات تابع مثلی با پارامترهای $(\mu/\sigma) = (200/40)$ و به علاوه زمانی بین $0/5$ و 2 واحد زمانی است. قطعات جهت سرویس به یک صف با گنجایش ۴ قطعه وارد می شوند. که چنانچه صف پر باشد قطعات به محل دیگری هدایت می گردند. پس از انجام بخش اول تعمیرات به صف جهت بخش دوم می روند که صف دوم گنجایش حداکثر ۲ قطعه دارد و چنانچه صف پر باشد هیچ قطعه ای نباید وارد شود و فعالیتها متوقف شود. هر دو صف در ابتدا خالی هستند و چنانچه ۱۰ قطعه جهت تعمیرات به خارج هدایت شود، عملیات شبیه سازی پایان می یابد.

مدل سیستم:



```

NETWORK:
CREATE,2;
BALK FROM QUEUE OF FIRST OPERATION IF 4 ITEMS ARE WAITING
QUEUE(1),0,4,BALK(SUBC);
ACT/1,TRILOG(0.2,0.4,0.8);
SERVICE ACTIVITY 1 IS BLOCKED IF SECOND QUEUE HAS
2 ITEMS WAITING
QUEUE(2),0,2,BLOCK;
ACT/2,UNIFORM(0.5,1.0);
TERM;
TERMINATE THE SIMULATION IF 10 ITEMS SUBCONTRACTED
TERM,10;
SUBC
ENDNET;

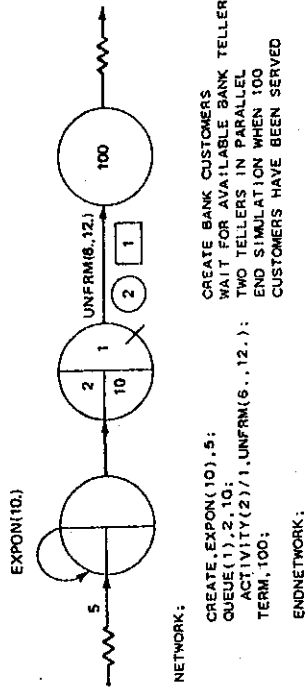
```

شکل ۱۵-۱۳ مدل SLAM سیستم تعمیر و نگهداری قطعات

مثال ۱۵-۲۹ عملیات بررسی محصولات یک شرکت تولیدی بوسیله یک بازرسی انجام می گیرد. زمان بازرسی تابعی از تعداد محصولات منتظر در صف ورودی فایل شماره ۱ و تعداد محصولات معیوب و منتظر بازرسی (در فایل شماره ۲) است. 70% قطعات پس از بازرسی سالمند و 30% معیوب که دوباره برای تعمیر و بازرسی بازگردانده می شوند. زمان ارسال برای قطعات سالم ۸ واحد زمانی و ارسال قطعات معیوب به صف مربوط ۶ واحد زمانی است.

ورود اولین مشتری در ۵ است. در ابتدای شروع کار بانک ۲ نفر در صف انتظار قرار دارند. چنانچه بانک ۱۰۰ مشتری را سرویس دهد، عملیات بانکی متوقف می شود.

مدل سیستم:

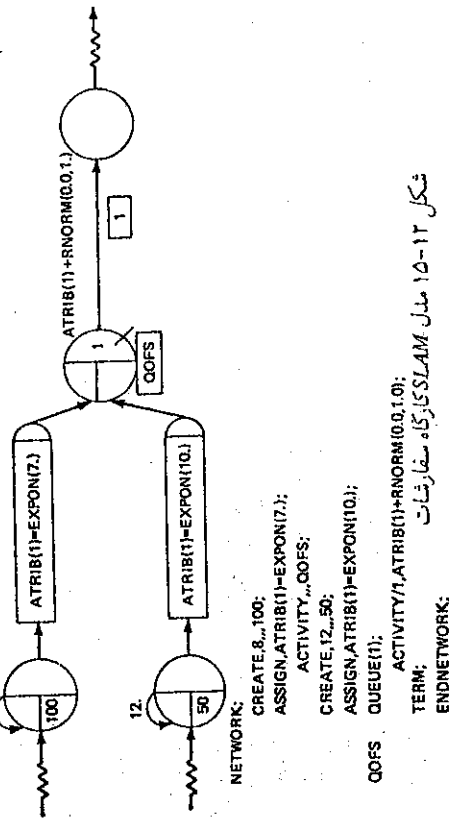


شکل ۱۵-۱۱ مدل SLAM صف و سرویس دهند: موازی

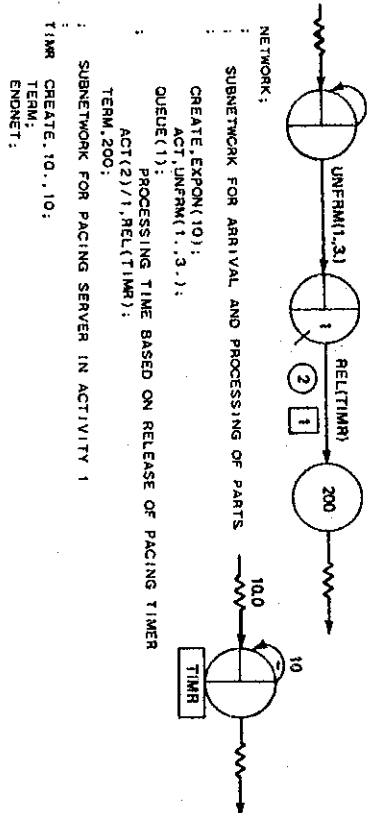
نکته: مولد با تولید بی نهایت و با شروع زمان یک شاخه خروجی دارد.

مثال ۱۵-۲۷ کارگاهی دو مرکز سفارش گیری دارد که در مرکز اول با فواصل زمانی ۸ و در مرکز دوم با فواصل زمانی ۱۲ واحد زمانی سفارش دریافت می شود. سفارشات به ماشینی می رود که زمان سرویس آن چنانچه از مرکز اول کار باید دارای توزیع نمایی با میانگین ۷ و اگر از مرکز دوم بیاید توزیع نمایی با میانگین ۱۰ می باشد. خطای تخمین زمان سرویس برای تمامی کارها با توزیع نرمال و با میانگین صفر و انحراف معیار ۱ است. اگر ماشین مشغول باشد کارها در صف به طول نامحدود قرار می گیرند. حداکثر کار دریافتی از مرکز اول ۱۰۰ سفارش و از مرکز دوم ۵۰ سفارش می باشد.

مدل سیستم:



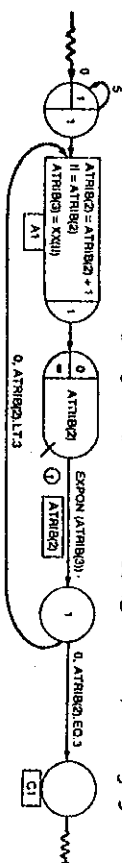
شکل ۱۵-۱۲ مدل SLAM یکگاه سفارشات



شکل ۱۵-۱۵ مدل SLAM عملیات مونتاژ یک ایستگاه کاری

در این مدل هرگاه یک مشتری از گره $TIMER$ خارج شود زمان سرورس فعالیت مشتری در فعالیت ۱ خاتمه می‌یابد. مانند یک (CLOCK) کلاسی که عملیات خروجی صف به آن وابسته است.

مثال ۱۵-۳۱ مشتریان یک سیستم بفراسصل زمانی ۵ واحد وارد می‌شوند و سه مرحله سرورس دارند (مرحله ۱، مرحله ۲، مرحله ۳). زمان سرورس تابع توزیع پواسون است که پارامتر آن در هر مرحله تغییر می‌کند. وقتی عملیات سرورس پایان می‌یابد که مشتری هر سه مرحله را طی کرده باشد. مدل با استفاده از گره صف (مدل خاص) و یک شاخه فعالیت سرورس).

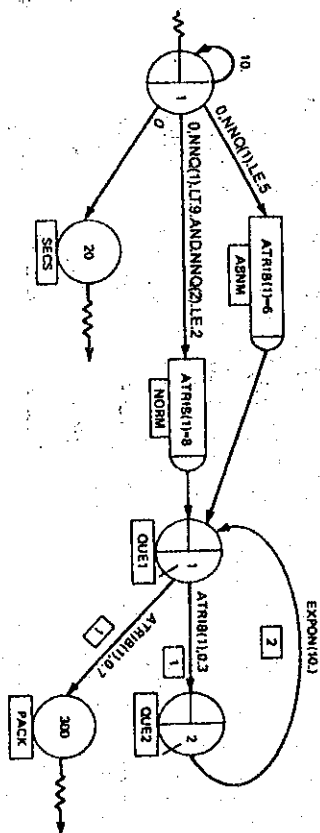


```

NETWORK:
CREATE, 5, 0, 1, 1;
ATTRIB(2)=NEXT SERVER STATION, ATTRIB(2)=MEAN*SERVICE TIME
A1 ASSIGN, ATTRIB(2)=ATTRIB(2)+1, I1=ATTRIB(2), ATTRIB(3)=XX(1);
SET FILE NUMBER=SERVER STATION NO.
QUEUE(ATTRIB(2)=1, 3);
ACT(1)/ATTRIB(2)=1, 3, EXPON(ATTRIB(3));
GOON, 1;
ACT(0, ATTRIB(2), LT, 3, A1; GO TO NEXT SERVER STATION
ACT(0, ATTRIB(2), EQ, 3, CI; COMPLETED THREE SERVICES
TERM;
ENDNETWORK;

```

شکل ۱۵-۱۶ مدل SLAM مثال ۱۵.۱۳ و ارائه گره صف جدید



شکل ۱۵-۱۴ مدل SLAM عملیات بررسی یک شرکت تولیدی

در مدل فوق گره مولد چون پارامتر M آن برابر ۱ است، خروجی به یکی از سه شاخه می‌رود. اگر تعداد قطعات قابل ۱ (قطعات منتظر) کمتر یا مساوی ۵ عدد باشد، شاخه یک انتخاب می‌شود که برای نشان دادن عبور از این شاخه و این وضعیت خانه اول برادر مشخصه مقدار ۶ می‌گیرد (در گره $ABNM$). شاخه دوم زمانی انتخاب می‌گردد که تعداد قطعات منتظر تغییر کمتر یا مساوی ۹ و هم قطعات معمول کمتر از ۲ باشد. (گره $NORM$) خانه اول برادر مشخصه برابر ۸ قرار دارد یعنی زمان سرورس این قطعه ۸ واحد زمانی است. در صورتی که این ۲ شرط برقرار نباشد شاخه سوم انجام می‌پذیرد و بازای ۲۰ قطعه عبوری از آن به گره $SEGS$ شبیه‌سازی خاتمه می‌یابد و یا آنکه ۲۰ قطعه سالم به گره $PACK$ وارد شود. محصولات خراب وارد صف $QUEUE$ شده و منتظر شروع تعمیرات (فعالیت ۲) می‌گردند.

```

EQU(VALANCE/ATTRIB(1), ESERV, 7;
RANDOM(0, 0, 1, 0); NOISE;
NETWORK:
CREATE AND ASSIGN ESTIMATES OF PROCESSING TIME
CREATE, 8, 1, 100;
ASSIGN, ESERV=EXPON(7);
ACTIVITY, 1, 60;
ASSIGN, ESERV=EXPON(10, 1);
PROCESSING QUEUE
QUEUE(1);
ACTIVITY TIME IS ESTIMATE + ERROR IN ESTIMATE
ACTIVITY(1), ESERV+NOISE;
TERM;
ENDNETWORK;

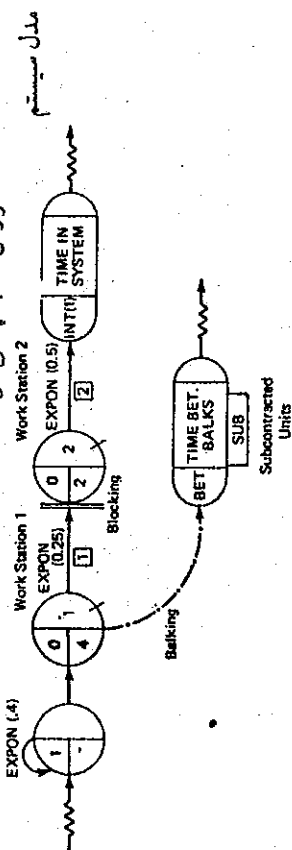
```

مثال ۱۵-۳۰ خط مونتاژی که عملیات مونتاژ آن در یک ایستگاه کاری انجام می‌شود، فقط در

پایان فواصل ۱۰ دقیقه‌ای و آن هم وابسته به یک تأییر است. قطعات ورودی با توزیع نمایی (میانگین ۱۰) وارد و بین زمان ۱ الی ۳ واحد طول می‌کشد که به صف مونتاژ برسد. پس از انتظار داخل صف در صورتی که تأییر علاقت دهد سرورس قطعات انجام و در صورت مونتاژ ۲۰۰ قطعه عملیات خاتمه می‌یابد که عمل مونتاژ با ۲ مونتاژگر انجام می‌شود.

می دهند. صف دوم چنانچه برای بخش دوم سرویس پر باشد. از ورود به این صف جلوگیری شود تا حداقل یک جای خالی در آن ایجاد گردد (BLOCKING).

آمارگیری از مدت زمان پسریدن قطعه در سیستم $INT(1)$ یعنی $INT(1)$ و تحت عنوان $TIME IN SYSTEM$ گزارش آن چاپ می شود.



گزارش خروجی:

شکل ۱۵-۱۸ مدل SLAM یک شرکت تعمیراتی

SLAM II SUMMARY REPORT

SIMULATION PROJECT SERIAL WORK STATIONS BY PRITISOR
DATE 7/14/1983 RUN NUMBER 1 OF 1

CURRENT TIME 0.3005-03
STATISTICS MARKS CLEARED AT TIME 0.0002-00

STATISTICS FOR VARIABLES BASED ON OBSERVATION

| MEAN VALUE | STANDARD DEVIATION | COEFF. OF VARIATION | MINIMUM VALUE | MAXIMUM VALUE | NUMBER OF OBSERVATIONS |
|----------------------------|--------------------|---------------------|---------------|---------------|------------------------|
| TIME IN SYSTEM 0.2781E-01 | 0.1720E-01 | 0.4625E-01 | 0.1002E-01 | 0.7191E-01 | 595 |
| TIME BET. BALKS 0.1545E-01 | 0.3201E-01 | 0.2103E-01 | 0.1280E-01 | 0.2757E-02 | 179 |

FILE STATISTICS

| FILE | ASSOC. NODE | AVERAGE LENGTH | STANDARD DEVIATION | MAXIMUM LENGTH | CURRENT LENGTH | AVERAGE WAITING TIME |
|------|-------------|----------------|--------------------|----------------|----------------|----------------------|
| 1 | QUEUE | 2.0423 | 1.5591 | 4 | 0 | 1.0280 |
| 2 | QUEUE | 1.5559 | 0.7283 | 4 | 2 | 0.7694 |
| 3 | CALCUL | 2.4175 | 0.5124 | 4 | 3 | 0.2358 |

SERVICE ACTIVITY STATISTICS

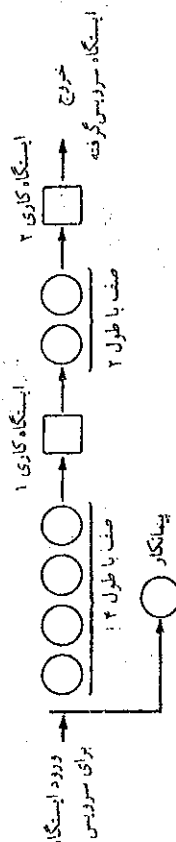
| ACTIVITY INDEX | START NODE OR ACTIVITY LABEL | SERVED CAPACITY | AVERAGE UTILIZATION | STANDARD DEVIATION | CURRENT UTILIZATION | AVERAGE BLOCKAGE | MAXIMUM FILE TIME/SERVERS | MAXIMUM BUSY TIME/SERVERS | ENTITY COUNT |
|----------------|------------------------------|-----------------|---------------------|--------------------|---------------------|------------------|---------------------------|---------------------------|--------------|
| 1 | SIN1 S/C TIM | 1 | 0.4705 | 0.4891 | 1 | 0.604 | 2.2051 | 4.4780 | 589 |
| 2 | SIN2 S/C TIM | 1 | 0.9421 | 0.2326 | 1 | 0.000 | 1.5795 | 81.425 | 58 |

بخش اول گزارش نتایج آمارگیری گره های COLET است که نامهای مختلف دارند مانند: $TIME IN$ ، $TIME BET$ BALKS، $SYSTEM$ در دسته اول میانگین، انحراف معیار، ضریب تغییرات، می نیم، ماکزیم و تعداد آمارهای قابل مشاهده (دفعات ورود به گره COLET) است. در بخش دوم نتایج آماری فایده آمده است. دو صف در فایده ای او ۲ و سوم را برای قورم شبه سازی در نظر گرفته است.

توجه:

- ۱- استفاده از یک شاخه فعالیت و صف به جای ۳ شاخه و ۳ صف (که داخل آن تقسیم می شوند).
- ۲- سادگی ظاهری مدل.
- ۳- شماره گذاری فعالیتها با استفاده از مراحل مشخص شده در $ATRIB(2)$.
- ۴- نگهداری مراحل هر مشتری در $ATRIB(2)$ بردار مشخصه اش.
- ۵- استفاده از $ATRIB(2)$ به عنوان شماره فایل (۳) فایل با یک نماد.
- ۶- $ATRIB(3)$ زمان سرویس را دارد که در مرحله ۱ برابر $XX(1)$ در مرحله ۲ برابر $XX(2)$ و در مرحله ۳ برابر $XX(3)$ است.
- ۷- مقادیر $XX(i)=123$ در خارج شبکه تعریف می گردد.

مثال ۱۵-۳۲ اگر مدل انتزاعی یک شرکت جهت تعمیر و نگهداری بصورت شکل ۱۵.۱۷ باشد:



شکل ۱۵-۱۷ دیاگرام پیمانکار یک شرکت تعمیراتی

شبه سازی با ۳۰۰ واحد زمانی انجام می پذیرد. در این مدل مشخص است که تعمیرات در ۲ بخش انجام می گیرند و دو سرویس دهنده یکی با صف بطول ۲ قطعه و دومی با یک صف بطول ۲ قطعه دارد. زمان بین ورود قطعات جهت تعمیر به سیستم دارای توزیع نمائی با میانگین $0/4$ و زمان تعمیر بخش اول و زمان تعمیر بخش دوم با توزیعهای نمائی با میانگینهای $0/25$ و $0/5$ می باشد. اگر قطعات نتوانند به شرکت وارد شوند به خارج کارخانه یعنی به یک پیمانکار (BALKING)

موجودیت در کل سیستم جریان دارند. در صورت اشغال بازرسین تلوزیونها به صف انتظار می‌روند. پس از بازرسی تلوزیونها ۸۵٪ آنها برای بسته‌بندی و ۱۵٪ جهت تعمیر مجدداً ارسال می‌شوند. ایستگاه تعمیر نیز یک سرویس دهنده دارد و در صورت اشغال وی تلوزیونها به یک صف انتظار وارد می‌شوند. تلوزیونهای تعمیری مجدداً جهت بازرسی به صف انتظار ارسال می‌گردند.

در مدل سیستم تلوزیونها با فواصل زمانی UNFRM(3,5,7,5) تولید و به سیستم وارد می‌شوند. زمان تولید آنها در مشخصه بردار مشخصات آنها ذخیره می‌گردد. پس از ورود به صف INSP می‌روند. فعالیت ۱ سرویس بازرسی را انجام می‌دهد که زمان بازرسی UNFRM(6,12) است و توسط دویارز برای صورت می‌گردد. تلوزیونها پس از بازرسی به گره GOON وارد می‌شود تا ۱۵٪ درصد آنها برای تعمیر مجدداً به ADT برود. ۸۵٪ آنها برای بسته‌بندی ارسال و به گره DPRT وارد می‌شوند. ۱۵٪ تلوزیونها پس از بازرسی جهت تعمیر مجدداً به گره ADT می‌روند که زمان تعمیر بین ۲۰ و ۴۰ در فعالیت شماره ۲ و توسط یک تعمیرکار صورت می‌گیرد.

دستورات اصلی برنامه در خط ۲ تا ۱۳ و بین دستورات NETWORK و END آمده است. دستور

دارای بخشهای OREILLY (نام مدلسازی)، INSP AND ADJUST (نام برنامه)، دستور 6/25/83 و تعداد ۵ موجودیت برای سیستم است. دستور شماره ۱۵ دارای ۲ دارای ۲ فایل، ۲ صفت، مشخصه برای هر موجودیت می‌کند. دستور شماره ۱۹ (MONTR) گزارش ردیابی از زمان تا ۶۰ را تولید می‌کند. گزارش ردیابی از ATRIB، NNQ(1)، NNQ(2) است. انتهای برنامه با FIN تعیین می‌گردد.

گزارش ردیابی:

| LINE | TIME | NAME | VALUE | UNIT | TYPE | STATUS |
|------|-------|--------|-------|------|------|--------|
| 1 | 0.000 | GEN | 1.000 | 1 | 1 | 1 |
| 2 | 0.000 | INSP | 1.000 | 1 | 1 | 1 |
| 3 | 0.000 | ADJUST | 1.000 | 1 | 1 | 1 |
| 4 | 0.000 | DPRT | 1.000 | 1 | 1 | 1 |
| 5 | 0.000 | GOON | 1.000 | 1 | 1 | 1 |
| 6 | 0.000 | ADT | 1.000 | 1 | 1 | 1 |
| 7 | 0.000 | MONTR | 1.000 | 1 | 1 | 1 |
| 8 | 0.000 | NNQ(1) | 1.000 | 1 | 1 | 1 |
| 9 | 0.000 | NNQ(2) | 1.000 | 1 | 1 | 1 |
| 10 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 11 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 12 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 13 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 14 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 15 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 16 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 17 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 18 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 19 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 20 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 21 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 22 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 23 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 24 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 25 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 26 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 27 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 28 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 29 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 30 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 31 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 32 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 33 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 34 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 35 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 36 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 37 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 38 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 39 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 40 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 41 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 42 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 43 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 44 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 45 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 46 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 47 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 48 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 49 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 50 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 51 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 52 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 53 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 54 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 55 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 56 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 57 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 58 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 59 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 60 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 61 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 62 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 63 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 64 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 65 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 66 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 67 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 68 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 69 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 70 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 71 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 72 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 73 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 74 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 75 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 76 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 77 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 78 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 79 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 80 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 81 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 82 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 83 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 84 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 85 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 86 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 87 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 88 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 89 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 90 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 91 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 92 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 93 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 94 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 95 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 96 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 97 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 98 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 99 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |
| 100 | 0.000 | FIN | 1.000 | 1 | 1 | 1 |

در طول ۴۸۰ دقیقه شبیه‌سازی ۸۴ تلوزیون با زمان میانگین ۲۶/۶۳ دقیقه گذرانده در سیستم با بخش خروجی ارسال شده است. پس از آن انحراف معیار ۳۵/۹۱، حداقل مقدار زمانی ۶/۳۸۱

برای هر فایل با شماره‌های مربوطه نام، متوسط طول فایل (مشتربان)، انحراف معیار، ماکزیمم، منبهم آن و متوسط زمان انتظار مشتربان آمده است. در بخش سوم آمارهای فعالیت موجود سیستم می‌باشد. در شکل ترسیمی توزیع زمانی که مشتری در سیستم بسر برده است بر سبیل هستوگرام نشان داده شده است.

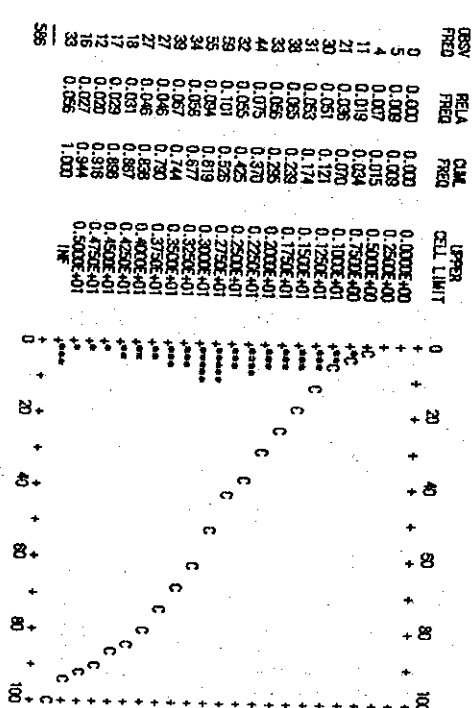
برنامه سیستم:

```

1 GEN, PRITSKE, SERIAL WORK STATIONS, 7/14/83, 1;
2 LIMITS 2, 1, 50;
3 NETWORK;
4 CREATE, EXPON(.4), 1;
5 QUEUE(1), 0, 4, BALK(SUB);
6 ACT/1, EXPON(.25);
7 QUEUE(2), 0, 2, BLOCK;
8 ACT/2, EXPON(.50);
9 COLCT, INT(1), TIME IN SYSTEM, 20/0, .25;
10 TERM;
11 SUB;
12 COLCT, BET, TIME, BET, BALKS;
13 TERM;
14 -INIT, 0, 300;
15 FIN;

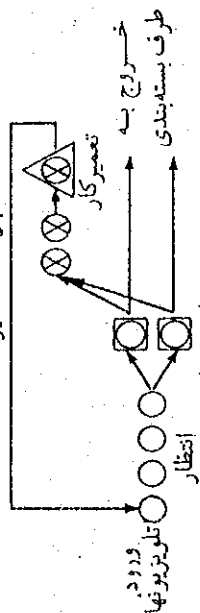
```

هستوگرام



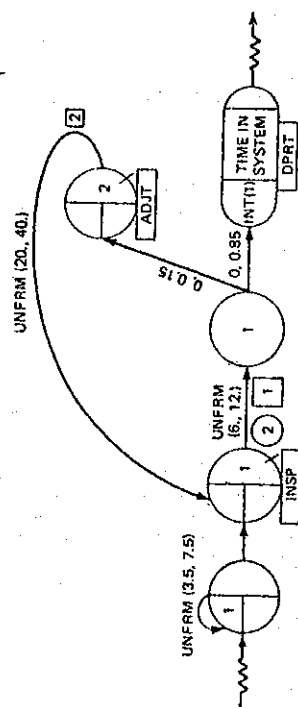
مثال ۲۳-۱۵ بازرسی و کنترل یک خط تولید تلوزیون در نظر گرفته می‌شود. تلوزیونها پس از بازرسی اگر سالم و بی عیب باشند برای بسته‌بندی منتقل می‌شوند. در غیر این صورت به ایستگاه مربوطه می‌روند و بعد از رفع اشکال مجدداً جهت بازرسی ارسال می‌گردند. تولید تلوزیونها زمانی بین ۳/۵ و تا ۷/۵ واحد زمانی می‌برد. مدت بازرسی بین ۶ تا ۱۲ واحد زمانی طول می‌کشد. تجربه نشان داده است که ۱۵٪ تلوزیونها دارای اشکال می‌باشند و زمان رفع اشکال بین ۲۰ تا ۴۰ واحد زمانی است و ۸۵٪ محصولات برای بسته‌بندی ارسال شوند. عمل بازرسی بر سبیل یک فعالیت سرویس با دویارس مدل می‌شود. تلوزیونها به عنوان یک

بارگشت تعمیر شده ها



شکل ۱۹-۱۵ دیاگرام سیستم بازرسی تلویزیونها

مدل سیستم:



شکل ۲۰-۱۵ مدل SLAM بازرسی و کنترل تلویزیونها

برنامه

```

1 GEN OREILLY, TV INSP, AND ADJUST, 6/25/83, 1;
2 LIMITS, 2, 2, 50;
3 NETWORK;
4 CREATE UNFRM(3, 5, 7, 5), 1;
5 INSP QUEUE(1);
6 ACT(2)/1, UNFRM(6, 12);
7 GOON;
8 ACT, 85, DPRT;
9 ACT, 15, ADJT;
10 ADJT QUEUE(2);
11 ACT/2, UNFRM(20, 40, 1), INSP;
12 DPRT COLCT, INT(1), TIME IN SYSTEM;
13 TERM;
14 END;
15 INIT, 0, 450;
16 ;
17 ; WRITE A TEXT TRACE FROM TIME 0 TO TIME 50;
18 ; PRINT THE VALUE OF ATTRIB(1), ANQ(1), AND NMQ(2)
19 MONTR, TRACE, 0, 50, ATTRIB(1), ANQ(1), NMQ(2);
20 FIN;

```

مثال ۳۴-۱۵ مدل سازی عملیات استخراج یک معدن مدنظر است. از سه معدن عملیات استخراج صورت می گیرد و کامیونهای بارگیری شده برای تخلیه به محل یک دستگاه سنگ خردکن می روند.

شبیه سازی

۴۵۰

ماکریم زمانی ۱۶۲/۲ خواهد بود. بخش بعدی گزارش خروجی از فایلها (صفها) گزارش می دهد. از فایلها نام آنها، میانگین طول فایل (صف آنها)، انحراف معیار، حداکثر طول فایل (صف)، طول فعلی آن و میانگین زمان انتظار آمده است. مثلاً برای فایل اول به ترتیب، INSP QVEVE (نام صف)، ۸۵۱۵/ (میانگین طول صف)، ۷۷۵۶/ (انحراف معیار آن)، ۳ (حداکثر طول صف)، ۰ (موجودی فعلی صف) و ۴۶۵/ (میانگین زمان انتظار این صف) است.

بخش بعدی گزارش ارائه اطلاعات فعالیتها است. از فعالیتها اطلاعات: شماره فعالیت (مانند ۱)، گروه شروع فعالیت (INSPECTION)، تعداد سرویس دهنده (۲)، میانگین استفاده (۱/۹۰۵۹)، انحراف معیار (۲۹۲۰/۰)، ظرفیت فعلی (۲)، میانگین زمان بلوک کردن فعالیت (۵)، ماکزیمم زمان بیکاری سرویس دهنده (۲)، ماکزیمم زمان اشتغال سرویس دهنده (۲) و تعداد موجودیهای سرویس گرفته (۹۹) است. برای فعالیت شماره ۲ نیز به همین ترتیب و با مقادیر مربوطه آمده است. درانتها تعیین می کند که یک تلویزیون منتظر تعمیرات هر دو بازرسی و تعمیر در انتهای شبیه سازی مشغولند. با افزودن چهار تلویزیون فوق به ۸۴ تلویزیون نشان می دهد که ۸۸ تلویزیون تولید شده است که با ۴۸۰-۵/۵ یا تقریباً ۸۷ ورود مطابق دارد.

این اطلاعات برای بررسی هایی از قبیل: آیا فضای کافی بازرسی تعمیرات موجود است؟ تخصیص نفرات برای بازرسی و تعمیر تلویزیونها مناسب است؟ زمان صرف شده برای یک تلویزیون زیاد نیست؟ مناسب است.

گزارش خروجی

SLAM II SUMMARY REPORT

SIMULATION PROJECT TV INSP, AND ADJUST. BY OREILLY
DATE 6/25/1983 RUN NUMBER 1 OF 1

CURRENT TIME 0.4000E+03
STATISTICAL ARRAYS CLEARED AT TIME 0.0000E+00

STATISTICS FOR VARIABLES BASED ON OBSERVATION

| | MEAN VALUE | STANDARD DEVIATION | COEFF OF VARIATION | MINIMUM VALUE | MAXIMUM VALUE | NUMBER OF OBSERVATIONS |
|----------------|------------|--------------------|--------------------|---------------|---------------|------------------------|
| TIME IN SYSTEM | 0.2863E+02 | 0.3591E+02 | 0.1348E+01 | 0.6381E+01 | 0.1622E+03 | 84 |

FILE STATISTICS

| FILE NUMBER | ASSOC NODE LABEL/TYPE | AVERAGE LENGTH | STANDARD DEVIATION | MAXIMUM LENGTH | CURRENT LENGTH | AVERAGE WAITING TIME |
|-------------|-----------------------|----------------|--------------------|----------------|----------------|----------------------|
| 1 | INSP QUEUE | 0.9515 | 0.7758 | 3 | 0 | 4.0485 |
| 2 | ADJT QUEUE | 1.6531 | 1.1465 | 4 | 1 | 4.8822 |
| 3 | ADJT DEQUEUE | 3.3019 | 0.1972 | 6 | 4 | 4.7177 |

SERVICE ACTIVITY STATISTICS

| ACTIVITY INDEX | START MODE OR ACTIVITY LABEL | SERVICES CAPACITY | AVERAGE UTILIZATION | STANDARD DEVIATION | CURRENT UTILIZATION | AVERAGE SLOTTAGE | MAXIMUM IDLE TIME/SERVICES | MAXIMUM BUSY TIME/SERVICES | ENTITY COUNT |
|----------------|------------------------------|-------------------|---------------------|--------------------|---------------------|------------------|----------------------------|----------------------------|--------------|
| 1 | INSPECTION ADJUSTMENT | 2 | 1.9259 | 0.7523 | 2 | 0.0000 | 2.0000 | 2.0000 | 89 |
| 2 | | 1 | 0.6719 | 0.3352 | 1 | 0.0000 | 46.3651 | 245.4028 | 13 |

مدلسازی شبکه های شامل عملیات مسیریابی و توزیع موجودیت جدای کامیونها ۲۰ تنی و ۵۰ تنی است. هر موجودیت پنج صفت مشخصه دارد: (۱) تناژ کامیون (۲۰ یا ۵۰)، (۲) شماره معدنی که کامیون در آن قرار دارد، (۳) میانگین بارگیری، (۴) میانگین زمان تخلیه بار (۵) زمان بارگشت، (۶) کامیون سیستم (۶ کامیون ۲۰ تنی و ۳ کامیون ۵۰ تنی) در انتهای برنامه قرار داده می شود (موجودیهای ثابتی هستند). موجودیها به صف SVLS حاوی ۳ صف برای بارگیری وارد می شوند. کامیونها پس از تخلیه بار مجدداً به این صف برمی گردند. SHOVEI یک صفت از موجودیت کامیون است و شامل شماره معدنی است که کامیون برای آن مشخص شده است. هر صف معدن بر سبیل گره SVLS با عملیات بارگیری از طریق فعالیت ACTIVITY ارائه می شود. متغیر SHOVEI با صفت مشخصه درم موجودیها برابر است. لذا موجودیها ورودی حسب این صفتشان به صف مربوطه می روند. فعالیت سرویس بارگیری با توزیع نمایی صورت می گیرد که میانگین آن در متغیر LOAD قرار دارد (معادل صفت مشخصه ۳ موجودیها ATTRIB کامیون پس از بارگیری بطرف سنگ خردکن می رود و وارد صف سرویس می شود. پس از آن فعالیت تخلیه در آسباب با توزیع نمایی و میانگین UNLOAD صورت می گیرد (مطابق (ATTRIB(4) پس از تخلیه موجودیت با زمان TRAVEL مطابق (ATTRIB(5) به RSVLS باز می گردد. توجه کنید که در دستور بالای مدل کلمات معادل صفات موجودیت آمده است و ضمناً در فایل شماره ۴ (CRSR) انتخاب کامیونها برای فعالیت تخلیه بر اساس امتیاز صفت ATTRIB(1) یعنی HVF(1) تنظیم می شود یعنی اولویت با کامیونهای ۵۰ تنی است.

برنامه سیستم:

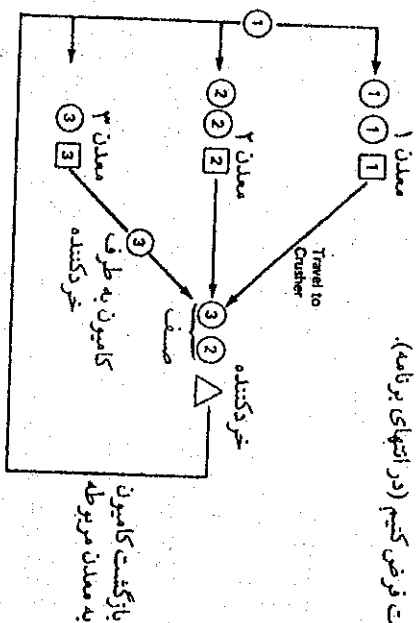
```

1 GEN, PARTS, QUANT, CREATIONS, 12/19/85, 1;
2 LIMITS 4, 5, 75;
3 PRIORITY/4, HVF(1);
4 EQUIVALENCE/ATTRIB(2), SHOVEI/
5 ATTRIB(3), LOAD/
6 ATTRIB(4), UNLOAD/
7 ATTRIB(5), TRAVEL;
8 NETWORK;
9
10 ATTRIB(1)=CONV(ATTRIB(2)*SVL NO., ATTRIB(3)*SVL TIME
11 ATTRIB(4)=CONV TIME, ATTRIB(5)*RETURN 181P TIME
12 SVLS QUEUE(SHOVEI, 1, 3);
13 ENCL GOON;
14 ACT/SHOVEI, 1, 3, EXPON(LOAD); LOAD SHOVEI
15 ACT/SHOVEI, 1;
16 CSRR ACT/SHOVEI, 1;
17 CSRR ACT/SHOVEI, 1;
18 CSRR ACT/SHOVEI, 1;
19 CSRR ACT/SHOVEI, 1;
20 ACT/4, TRAVEL, SVLS;
21 ENCL;
22 INIT 0, 480;
23
24 PLACE TWO 20 TON AND ONE 50 TON TRUCK IN EACH SHOVEI QUEUE
25
26 FILE STORAGE SHOVEI LOAD UNLOAD TRAVEL
27
28 ENTRY/ 1 20 1 5 2 1.5 /
29 2 20 1 5 2 1.5 /
30 3 20 1 5 2 1.5 /
31 4 20 1 5 2 1.5 /
32 5 20 1 5 2 1.5 /
33 6 20 1 5 2 1.5 /
34 7 20 1 5 2 1.5 /
35 8 20 1 5 2 1.5 /
36 9 20 1 5 2 1.5 /
37 10 20 1 5 2 1.5 /
38 11 20 1 5 2 1.5 /
39 12 20 1 5 2 1.5 /
40 13 20 1 5 2 1.5 /
41 14 20 1 5 2 1.5 /
42 15 20 1 5 2 1.5 /
43 16 20 1 5 2 1.5 /
44 17 20 1 5 2 1.5 /
45 18 20 1 5 2 1.5 /
46 19 20 1 5 2 1.5 /
47 20 20 1 5 2 1.5 /
48 21 20 1 5 2 1.5 /
49 22 20 1 5 2 1.5 /
50 23 20 1 5 2 1.5 /
51 24 20 1 5 2 1.5 /
52 25 20 1 5 2 1.5 /
53 26 20 1 5 2 1.5 /
54 27 20 1 5 2 1.5 /
55 28 20 1 5 2 1.5 /
56 29 20 1 5 2 1.5 /
57 30 20 1 5 2 1.5 /
58 31 20 1 5 2 1.5 /
59 32 20 1 5 2 1.5 /
60 33 20 1 5 2 1.5 /
61 34 20 1 5 2 1.5 /
62 35 20 1 5 2 1.5 /
63 36 20 1 5 2 1.5 /
64 37 20 1 5 2 1.5 /
65 38 20 1 5 2 1.5 /
66 39 20 1 5 2 1.5 /
67 40 20 1 5 2 1.5 /
68 41 20 1 5 2 1.5 /
69 42 20 1 5 2 1.5 /
70 43 20 1 5 2 1.5 /
71 44 20 1 5 2 1.5 /
72 45 20 1 5 2 1.5 /
73 46 20 1 5 2 1.5 /
74 47 20 1 5 2 1.5 /
75 48 20 1 5 2 1.5 /
76 49 20 1 5 2 1.5 /
77 50 20 1 5 2 1.5 /
78 51 20 1 5 2 1.5 /
79 52 20 1 5 2 1.5 /
80 53 20 1 5 2 1.5 /
81 54 20 1 5 2 1.5 /
82 55 20 1 5 2 1.5 /
83 56 20 1 5 2 1.5 /
84 57 20 1 5 2 1.5 /
85 58 20 1 5 2 1.5 /
86 59 20 1 5 2 1.5 /
87 60 20 1 5 2 1.5 /
88 61 20 1 5 2 1.5 /
89 62 20 1 5 2 1.5 /
90 63 20 1 5 2 1.5 /
91 64 20 1 5 2 1.5 /
92 65 20 1 5 2 1.5 /
93 66 20 1 5 2 1.5 /
94 67 20 1 5 2 1.5 /
95 68 20 1 5 2 1.5 /
96 69 20 1 5 2 1.5 /
97 70 20 1 5 2 1.5 /
98 71 20 1 5 2 1.5 /
99 72 20 1 5 2 1.5 /
100 73 20 1 5 2 1.5 /

```

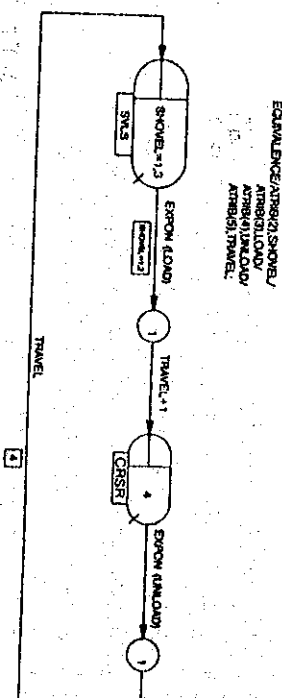
ظرفیت کامیونها ۲۰ و ۵۰ تن هستند. زمان بارگیری، حمل، تخلیه و بارگشت مجدد برای کامیونهای ۲۰ تن تابع توزیع نمایی با میانگین ۵ و عدد ثابت ۲/۵ و متغیر تصادفی با میانگین ۲ و عدد ثابت ۱/۵ است. برای کامیونهای ۵۰ تن نیز توزیع نمایی با میانگین ۱۰، عدد ثابت ۳، متغیر تصادفی با میانگین ۴ و عدد ثابت ۲ است.

برای هر معدن ۲ کامیون ۲۰ تنی و یک کامیون ۵۰ تنی در نظر گرفته اند در صفهای بارگیری اولویت براساس زمان ورود و در صف تخلیه براساس ظرفیت کامیون است. (مدت شبیه سازی ۴۸۰ واحد زمانی). ۹ کامیون داریم که در سیستم تولید نمی شوند و از بین نمی روند و لذا باید آنها را به عنوان داده های ثابت فرض کنیم (در انتهای برنامه).



شکل ۲۱-۱۵ دیگرام استخراج معدن

مدل سیستم:



شکل ۲۲-۱۵ مدل SLAM سیستم استخراج معدن

فصل شانزدهم

گره‌های منابع و دروازه‌ها

در مدیریت منابع یک سیستم عامل، متقاضیان جهت انجام امور مربوطه نیاز به منابع بسیاری دارند و پس از استفاده نیز منابع را به سیستم باز می‌گردانند. در SLAM نیز موجودیتها می‌توانند جهت عبور از گره فعالیت نیاز به منابع داشته باشند. اگر سرورس‌دهنده آزاد باشد و منابع مورد نیاز موجودیت فراهم باشد از فعالیت سرورس عبور می‌کنند.

انواع منابع و ظرفیت هر منبع (یعنی کل موجودی آن منبع) قابل تعریف می‌باشد. موجودیتهایی که نیاز به منابع دارند به گره Awaiting وارد و در صورت وجود منابع مورد نیاز از آن خارج می‌شوند. تعداد منابع و انواع آن در این گره تعریف می‌گردند. کلیه موجودیتهای منتظر در گره Awaiting در فایلی که در گره تعریف شده، قرار می‌گیرند.

برای آنکه امکان استفاده از منابع در اختیار موجودیتهای دیگر برای موجودیت جاری فراهم شود به گره PREEMPT وارد می‌شود. موجودیتهایی که در گره‌های Awaiting و PREEMPT منتظر هستند، در استفاده از منابعی که آزاد خواهند شد دارای تقدم می‌باشند. تقدم مورد نظر و ظرفیت منابع مورد نیاز در بلاک RESOURCE تعیین می‌شود.

موجودیتهایی که نیاز به منابع خویش ندارند. جهت آزاد سازی منابع در اختیارشان به گره FREE می‌روند و در واقع به این طریق به دیگران آزادسازی منابع را اطلاع می‌دهند. ظرفیت منابع موجود می‌تواند با ورود موجودیت به گره ALTER تغییر کند و ظرفیتها افزایش یا کاهش یابد.

برای عبور موجودیتها و انتظار آنها جهت اعلانات خاصی و علامت دادن جهت خروج آنها می‌توان گره GATE تعریف کرد. دروازه‌هایی که باز و بسته بودن آنها علامتی است که برای موجودیت معنی خاصی دارد. دروازه‌ها باز و بسته می‌شوند و در گره Awaiting نام دروازه‌های مربوطه ذکر می‌گردد.

گره‌های OPEN و CLOSE جهت عبور موجودیت و باز و بسته کردن دروازه‌ها می‌باشند و هر زمانی که دروازه‌ای باز شود، کلیه منتظران آن با یکدیگر از گره Awaiting خارج می‌شوند. (مثال نمادین این

در برنامه پس از دستورهای GEN و LIMIT دستور PRURITY آمده است. این دستور برای تعیین اولویت صفها می‌باشد که در آن برای فایل ۴ (CRSR) اولویت HVF(1) تعریف شده است. این اولویت یعنی موجودیتهایی (کامیونها) که صفت اول آنها (ATTRIB(1)) بیشتر باشند اولویت بیشتری دارند. تقدم بقیه صفها FIFO است.

دستور EQVTVALENBCE برای نامگذاری دلخوای صفات کامیونها بکار می‌رود. مدل شبکه شامل دستورهای ۹ تا ۱۲ است. کامیونها پس از بارگیری برای رفتن به دستگاه خردکن از یک فعالیت عبور می‌کنند (خط ۱۴). به دنبال آن گره صف CRSR با سرورس دیگری (خط ۱۸) و پس از آن گره GOON (خط ۱۹) آمده است. آخرین فعالیت (خط ۲۰) فعالیت شماره ۴ است که زمان TRAVEL دارد. خط ۲۸ تا ۳۶ مقداردهی ۹ موجودیت ثابت (کامیونها) را انجام می‌دهد. که هر خط آن به ترتیب برای یک کامیون صفاتش را تعریف می‌کند. اولین دستور ENTRY برای فایل شماره ۱ صفات: (20,1,5,2,1,5), (20,1,5,2,1,5) و (50,1,10,4,2,0) است. که اولین عدد سمت چپ اولین صفت یعنی -- (۲۰ یا ۵۰ تن) و صفت دوم و مانند آن می‌آید. برای فایل ۲ و فایل ۳ نیز مانند آن تکرار شده است (مجموعاً ۹ موجودیت).

گزارش خروجی

SLAM II SUMMARY REPORT

SIMULATION PROJECT QUARRY OPERATIONS BY PRITSCHER
DATE 12/19/1985 RUN NUMBER 1 OF 1

CURRENT TIME 0.4800E+03
STATISTICS APPRIS CLEARED AT TIME 0.0000E+00

FILE STATISTICS

| FILE NUMBER | ASSOC. NODE LABEL/TYPE | AVERAGE LENGTH | STANDARD DEVIATION | MAXIMUM LENGTH | CURRENT LENGTH | AVERAGE WAITING TIME |
|-------------|------------------------|----------------|--------------------|----------------|----------------|----------------------|
| 1 | SLS QUEUE | 0.5159 | 0.7567 | 2 | 1 | 4.5025 |
| 2 | SLS QUEUE | 0.5583 | 0.7781 | 2 | 4 | 4.9451 |
| 3 | SLS QUEUE | 0.5072 | 0.7457 | 2 | 2 | 4.5821 |
| 4 | CRSR QUEUE | 3.0464 | 2.2925 | 8 | 0 | 9.4652 |
| 5 | CRSR CALENDAR | 4.3741 | 1.3701 | 9 | 5 | 2.4618 |

REGULAR ACTIVITY STATISTICS

| ACTIVITY INDEX/LABEL | AVERAGE UTILIZATION | STANDARD DEVIATION | MAXIMUM CURRENT UTIL. | ENTITY COUNT |
|----------------------|---------------------|--------------------|-----------------------|--------------|
| 4 TRUCK RETURN | 0.5400 | 0.7431 | 5 | 1 |
| | | | | 152 |

SERVICE ACTIVITY STATISTICS

| ACTIVITY INDEX | START MODE OR ACTIVITY LABEL | SERVICES CAPACITY | AVERAGE UTILIZATION | STANDARD DEVIATION | CURRENT UTILIZATION | AVERAGE BLOCKAGE | MAXIMUM IDLE TIME/SERVERS | MAXIMUM BUSY TIME/SERVERS | ENTITY COUNT |
|----------------|------------------------------|-------------------|---------------------|--------------------|---------------------|------------------|---------------------------|---------------------------|--------------|
| 1 | LOAD SQUEL | 1 | 0.2073 | 0.4550 | 1 | 0.0000 | 20.3895 | 62.8828 | 59 |
| 2 | LOAD SQUEL | 1 | 0.0423 | 0.0793 | 1 | 0.0000 | 18.4206 | 47.8185 | 51 |
| 3 | LOAD SQUEL | 1 | 0.7055 | 0.4654 | 1 | 0.0000 | 16.5853 | 57.1151 | 50 |
| 0 | CRSR QUEUE | 1 | 0.5813 | 0.2843 | 1 | 0.0000 | 9.2714 | 84.1552 | |

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |

FREE.ATTRIB(3)/1;

مثال ۱۶-۸

با عبور موجودیت از گره FREE به تعداد یک واحد از منبع مشخص شده از خانه سوم بردار مشخصه موجودیت آزاد می‌شود.

۱۶.۴- گره الویت PREEMPT

| | | | |
|-----|----|-----|------|
| IFL | PR | RES | NATR |
| 1 | 1 | 1 | 1 |

PREEMPT(IFL,PR,RES,SNLBL,NATR,M);

SNLBL

شکل ۱۱-۴ سبیل گرافیکی گره PREEMPT و دستور SLAM

الویت در دسترسی به منابع وجود دارد و در واقعی حالت خاصی از گره AWAIT است که حق تقدم در آن مطرح است.

پارامترها:

IFL: شماره فایلی که موجودیتها در آن منتظرند.

RES: نام منبع.

PR: الویت موجودیت نسبت به دریافت منبع اولویت (HIGH(K) و LOW(K) خانه Kم بردار

مشخصه (پیش فرض FIFO).

SNLBL: برچسب گره‌ای که خروجی به آن می‌رود.

NATR: شماره خانه بردار مشخصه مشتری که زمان باقیمانده پردازش در آن ذخیره شود.

چنانچه موجودیت a در گره AWAIT، منبعی (RES) را اشغال کند و از گره AWAIT خارج شود اگر همان زمان در بخشی دیگر از سیستم موجودیت b جهت تخصیص همین منبع به گره PREEMPT وارد شود مقدار است منبع اشغالی را از دست موجودیت a خارج سازد (الویت)، یعنی منابع دریافتی خوش را به موجودیت a که در گره PREEMPT قرار دارد می‌سپارد و سپس به گره a برچسب SNLBL می‌رود.

مدت زمان باقیمانده فرآیند موجودیت a در متغیر بردار مشخصه ATRIB(NATR) قرار می‌گیرد و چنانچه این برچسب برای گره (SNLBL) تعریف نشده باشد، موجودیت a مجدداً به گره AWAIT باز

می‌گردد (در واقع با تعریف محل هدایت موجودیتها این چنتی که نوبت خویش را از دست داده‌اند

با الویت می‌کنیم) و منتظر تخصیص منابع مورد تقاضایش می‌شود.

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |

AWAIT.ATTRIB(2)=3.5;TELEX/1;

مثال ۱۶-۵

شماره فایلی در خانه دوم بردار مشخصه موجودیت تعیین می‌گردد و یکی از ۳ فایلی ۳، ۴ یا ۵ می‌باشد. هر موجودیت از منبع TELEX یک واحد نیاز دارند، که حداکثر ۳ واحد موجود دارد. تقسیم برای قرار گرفتن مشتریان در فایلهای در دستور PRIORITY تعیین می‌شود. در مثال فوق در صورت دسترسی به منبع آزاد، کلیه موجودیتهای فایلی ۴ از گره خارج می‌شود و سپس فایلی ۵ و در انتها فایلی شماره ۳ می‌باشد.

۱۶.۳- گره آزادسازی FREE

FREE.RES,UF,M;

| | | |
|-----|----|---|
| RES | UF | M |
| 1 | 1 | 1 |

شکل ۱۱-۳ سبیل گرافیکی گره FREE و دستور SLAM

برای آزادسازی منابع تخصیص یافته در گره AWAIT بکار می‌رود (به تعداد NUF از منبع RES):

پارامترها:

RES: نام منبع.

UF: تعداد منبع آزاد شده.

M: شاخه‌های خروجی (تعداد).

مثال ۱۶-۶

FREE.BOOKS/2;

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |

هر دفعه که موجودیتی از این گره عبور کند ۲ واحد از منبع BOOKS آزاد می‌شود.

FREE.BOOKS.ATTRIB(4);

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |

مثال ۱۶-۷

با عبور موجودیت از این گره به تعداد مشخص در خانه چهارم بردار مشخصه اش از منبع BOOKS آزاد می‌شود.

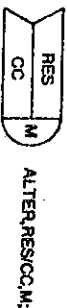


PREEMPT(1), ATTRIB(4), 2

مثال ۱۶-۱۲

همان مثال فوق است و فقط منبع مورد در خواست در خانه چهارم بردار مشخصه موجودیت (ATTRIB(4) ذخیره شده، الیت به روش FIFO (پیش فرض) است.

۱۶.۵- گروه جایگزین ALTER



شکل ۵-۱۶ سبیل گرافیکی گروه (جایگزین) ALTER و دستور SLAM

در مواقعی که کاهش داریم و کاهش بیشتر از منابع آزاد است، منابع موجود آزاد حذف می‌شوند و مابقی از منابعی که آزاد می‌شوند حذف می‌گردند.

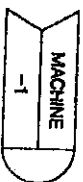
گروه جایگزین (ALTER) برای تغییر ظرفیت منابع بکار می‌رود. این امر یعنی افزایش منبع (در مواقعی مانند افزایش منبع جدید) و کاهش منبع (در مواقع خرابی منبع) توسط این گروه امکان‌پذیر است. موجودیتی که وارد این گروه می‌شود به تعداد CC به ظرفیت (موجودی کل) منبع RES اضافه می‌شود. (CC > 0) و یا به این تعداد از منبع RES کم می‌کند (CC < 0).

پارامترها:

نام منبع: RES

تعداد منبع: CC

مثال ۱۶-۱۳



ALTER, MACHINE, 1;

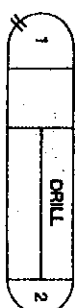
از منبعی بنام MACHINE یک واحد کسر می‌گردد.

عدم الیت در تخصیص منابع گروه PREEMPT:

۱- ظرفیت منبع RES بیش از یک واحد است.

۲- پیش از اعمال الیت موجودیت در فعالیت سرونش باشد.

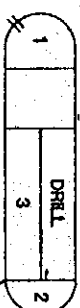
۳- پیش از اعمال الیت موجودیت وارد فایلهای (گروه‌های) AWAIT یا QUEUE شده باشد.



PREEMPT(1), DRILL, 2

مثال ۱۶-۹

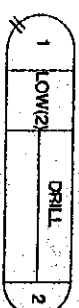
تخصیص منبع به روش FIFO (پیش فرض) صورت می‌گیرد. موجودیت‌هایی که منابع خرویش را واگذار می‌کنند مجدداً به گروه PREEMPT یا AWAIT دیگری که از آن گروه، منابع در اختیارش قرار داده شده بود باز می‌گردند. زمان باقیمانده این موجودیت ذخیره نمی‌شود و از این نظر مورد توجه نمی‌باشد. هر موجودیت پس از خروج از این گروه حداکثر به دو فعالیت پس از آن می‌رود.



PREEMPT(1), DRILL, AWIT2, 2

مثال ۱۶-۱۰

موجودیتی که به ازای فعالیت این گروه منابع خود را از دست می‌دهد به گروه با برچسب AWIT2 هدایت می‌شود و زمان باقیمانده پروسه (فرآیند) در خانه سوم بردار مشخصه موجودیت (ATTRIB(3) ذخیره می‌شود.




PREEMPT(1), LOW(2), DRILL, 2

مثال ۱۶-۱۱

الیت با کمترین مقدار خانه دوم بردار مشخصه موجودیت منتظر در گروه است (LOW(2)). چنانچه موجودیت (۵) وارد این گروه شود که مقدار مشخصه دوم آن کمتر از موجودیت (۵) است و قبلاً در گروه دیگر PREEMPT منبع DRILL را در یافته است، منبع فوق به موجودیت (۵) فوق‌الذکر تخصیص می‌یابد و موجودیت (۵) به گروه PREEMPT (محال خرویش) بازگردانده می‌شود.

۱۶.۷- گره باز OPEN

OPEN, GATE, M;


شکل ۱۶-۷: سمبل گرافیکی گره OPEN و دستور SLAM

پارامترها:


GATE: نام برچسب گره GATE.

M: شاخه خروجی (همزمان).

جهت باز کردن گره بسته (GLBL) استفاده می‌شود و اگر موجودیتی از این گره عبور کند (وارد شود) دروازه بسته (GLBL) باز می‌شود.

۱۶.۸- گره بستن CLOSE

پارامترها مانند گره OPEN است و عکس عمل OPEN صورت می‌گیرد.

CLOSE, GATE, M;



شکل ۱۶-۸: سمبل گرافیکی گره CLOSE و دستور SLAM

مثال ۱۶-۱۲



از منبع مذکور در بردار مشخصه ATRIB(2) به تعداد مذکور در بردار مشخصه ALTER(3) کسر می‌شود.

۱۶.۶- بلاک دروازه GATE

GATE, GLBL, OPEN or CLOSE, IFL1, IFL2;


شکل ۱۶-۶: سمبل گرافیکی بلاک دروازه GATE و دستور SLAM

از این گره برای تعریف دروازه‌های موردنظر (GATE) استفاده می‌گردد و در واقع گره‌های تأخیر دهنده هستند.

پارامترها:

GLBL: نام برچسب دروازه.

OPEN OR CLOSE: باز یا بسته بودن دروازه (وضعیت اولیه).

IFLS: شماره فعالیت‌هایی که موجودیهای گره AWAIT جهت باز شدن دروازه استفاده می‌کنند.

همانطور که منابع با بلاک RESOURCE تعریف می‌گردید، دروازه‌های تأخیری نیز با بلاک

RESOURCE تعریف می‌گردند. موجودیها در گره‌های AWAIT منتظر باز شدن دروازه‌های بسته

می‌باشند و باز (OPEN) یا بسته بودن (CLOSE) دروازه‌ها به وسیله دستورات OPEN و CLOSE تعیین

می‌شوند.

INV_POS : XX(4) موجودی با نام
ORDER-PT : XX(2) قطعه سفارش با نام
SCL : XX(3) حداکثر موجودی با نام
ORDER-QTY : ATR1B(1) رسیدن سفارش (مدت) با نام

```

SIMULATION PROJECT INVENTORY PROBLEM
DATE 8/7/7663
RUN NUMBER 1 OF 1
CURRENT TIME 0.372E+03
STATISTICAL REPORTS CLEARED AT TIME 0.500E+02

```

| ITEM
VALUE | STANDARD
DEVIATION | Coeff. of
VARIATION | MINIMUM
VALUE | MAXIMUM
VALUE | NUMBER OF
OBSERVATIONS |
|----------------|-----------------------|------------------------|------------------|------------------|---------------------------|
| 18. LIST SALES | 0.2096E+01 | 0.6112E+01 | 0.1923E+02 | 0.4020E+02 | 134 |
| SAFETY STOCK | 0.3003E+00 | 0.8234E+00 | 0.0003E+00 | 0.4003E+01 | 20 |

| MEAN
VALUE | STANDARD
DEVIATION | MINIMUM
VALUE | MAXIMUM
VALUE | TIME
INTERVAL | CURRENT
WAVE |
|---------------|-----------------------|------------------|------------------|------------------|-----------------|
| 0.4531E+02 | 0.1828E+02 | -0.300E+01 | 0.723E+02 | 0.200E+03 | 0.420E+02 |

| FILE NUMBER | ASSOC CODE LABEL/TYPE | AVERAGE LENGTH | STANDARD DEVIATION | MAXIMUM LENGTH | CURRENT LENGTH | AVERAGE WAITING TIME |
|-------------|-----------------------|----------------|--------------------|----------------|----------------|----------------------|
| 1 | MAIL | 0.1014 | 0.5293 | 5 | 0 | 0.0215 |
| 2 | CHILDREN | 2.2208 | 0.4213 | 8 | 2 | 0.1425 |

| RESOURCE
NUMBER | RESOURCE
LABEL | CURRENT
CAPACITY | AVERAGE
UTILIZATION | STANDARD
DEVIATION | MAXIMAL
UTILIZATION | CURRENT
UTILIZATION |
|--------------------|-------------------|---------------------|------------------------|-----------------------|------------------------|------------------------|
| 1 | PROD10 | 72 | 43.3572 | 19.9391 | 72 | 30 |

| RESOURCE
NUMBER | RESOURCE
LABEL | CURRENT
AVAILABLE | AVERAGE
AVAILABLE | MINIMUM
AVAILABLE | MAXIMUM
AVAILABLE |
|--------------------|-------------------|----------------------|----------------------|----------------------|----------------------|
| 1 | RA010 | 42 | 29.8429 | 0 | 70 |

هتال ۱۶-۱۹ عملیات یک سیستم بسته‌بندی بررسی می‌شود. بعد از وزن کردن عمل بسته‌بندی پس از کار افتادن دستگاه، تعمیر و عملیات مجدد آن با استفاده از گره PREEMPT انجام می‌دهد.

EXPON (0,2)

```
graph TD
    Start(( )) --> Node1((1))
    Node1 -- "NNRSC (RADIO) GTD  
OR BRAND L.E. 0.2" --> Node2[INV_POS = 1  
INV_POS = SCL]
    Node2 --> Node3((1))
    Node3 --> End1(( ))
    Node1 -- "INV_POS.L.E.  
REORDER PT" --> Node4[ORDER_QTY = SCL - INV_POS  
INV_POS = SCL]
    Node4 --> Node5((1))
    Node5 -- "LEAD TIME" --> Node6[NNRSC (RADIO) SAFETY STOCK]
    Node6 --> Node7[1]
    Node7 -- "RADIO ORDER_QTY" --> End2(( ))
    Node7 --> Node1
    Node2 --> LOST[LOST]
```

RESOURCE

| | | |
|-------|----|---|
| RADIO | 72 | 1 |
|-------|----|---|

برنامه

```

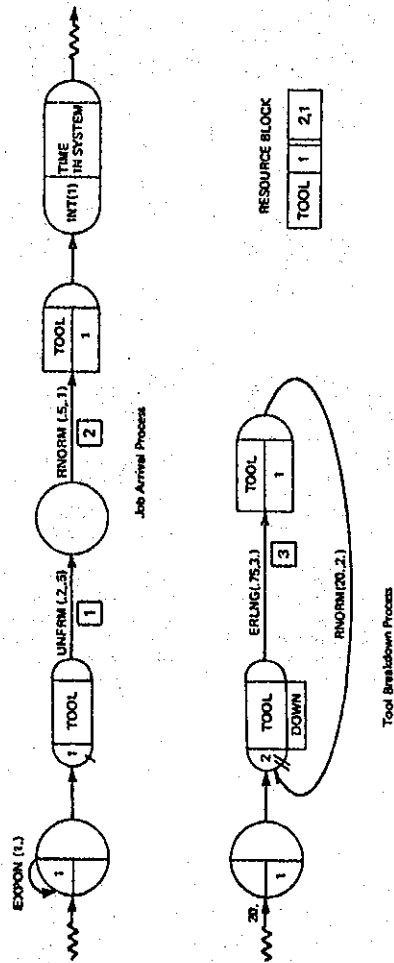
1 GEN AREPLY INVENTORY PROBLEM,9/12/83.1,Y,Y,V/N:
2 LIMITS.1,2,30:
3 EQUIVALENCE/XX(1),INV POS/
4 XX(2),REORDER_PT/
5 XX(3),SCL/
6 ATR18(1),ORDER_QTY/
7 3 LEAD TIME:
8 INTLC,INV POS=72,REORDER_PT=18,SCL=72:
9 TIMST,INV_POS,INV. POSITION:
10 NETWORK:
11 RESOURCE/RADIO(72),1:
12 :
13 :
14 :
15 :
16 :
17 :
18 :
19 :
20 :
21 :
22 :
23 :
24 :
25 :
26 :
27 :
28 :
29 :
30 :
31 :
32 :
33 :
34 :
35 :
36 :
37 :
38 :
39 :
40 :
41 :
42 :
43 :
44 :
45 :
46 :
47 :
48 :
49 :
50 :
51 :
52 :
53 :
54 :
55 :
56 :
57 :
58 :
59 :
60 :
61 :
62 :
63 :
64 :
65 :
66 :
67 :
68 :
69 :
70 :
71 :
72 :
73 :
74 :
75 :
76 :
77 :
78 :
79 :
80 :
81 :
82 :
83 :
84 :
85 :
86 :
87 :
88 :
89 :
90 :
91 :
92 :
93 :
94 :
95 :
96 :
97 :
98 :
99 :
100 :
101 :
102 :
103 :
104 :
105 :
106 :
107 :
108 :
109 :
110 :
111 :
112 :
113 :
114 :
115 :
116 :
117 :
118 :
119 :
120 :
121 :
122 :
123 :
124 :
125 :
126 :
127 :
128 :
129 :
130 :
131 :
132 :
133 :
134 :
135 :
136 :
137 :
138 :
139 :
140 :
141 :
142 :
143 :
144 :
145 :
146 :
147 :
148 :
149 :
150 :
151 :
152 :
153 :
154 :
155 :
156 :
157 :
158 :
159 :
160 :
161 :
162 :
163 :
164 :
165 :
166 :
167 :
168 :
169 :
170 :
171 :
172 :
173 :
174 :
175 :
176 :
177 :
178 :
179 :
180 :
181 :
182 :
183 :
184 :
185 :
186 :
187 :
188 :
189 :
190 :
191 :
192 :
193 :
194 :
195 :
196 :
197 :
198 :
199 :
200 :
201 :
202 :
203 :
204 :
205 :
206 :
207 :
208 :
209 :
210 :
211 :
212 :
213 :
214 :
215 :
216 :
217 :
218 :
219 :
220 :
221 :
222 :
223 :
224 :
225 :
226 :
227 :
228 :
229 :
230 :
231 :
232 :
233 :
234 :
235 :
236 :
237 :
238 :
239 :
240 :
241 :
242 :
243 :
244 :
245 :
246 :
247 :
248 :
249 :
250 :
251 :
252 :
253 :
254 :
255 :
256 :
257 :
258 :
259 :
260 :
261 :
262 :
263 :
264 :
265 :
266 :
267 :
268 :
269 :
270 :
271 :
272 :
273 :
274 :
275 :
276 :
277 :
278 :
279 :
280 :
281 :
282 :
283 :
284 :
285 :
286 :
287 :
288 :
289 :
290 :
291 :
292 :
293 :
294 :
295 :
296 :
297 :
298 :
299 :
300 :
301 :
302 :
303 :
304 :
305 :
306 :
307 :
308 :
309 :
310 :
311 :
312 :
313 :
314 :
315 :
316 :
317 :
318 :
319 :
320 :
321 :
322 :
323 :
324 :
325 :
326 :
327 :
328 :
329 :
330 :
331 :
332 :
333 :
334 :
335 :
336 :
337 :
338 :
339 :
340 :
341 :
342 :
343 :
344 :
345 :
346 :
347 :
348 :
349 :
350 :
351 :
352 :
353 :
354 :
355 :
356 :
357 :
358 :
359 :
360 :
361 :
362 :
363 :
364 :
365 :
366 :
367 :
368 :
369 :
370 :
371 :
372 :
373 :
374 :
375 :
376 :
377 :
378 :
379 :
380 :
381 :
382 :
383 :
384 :
385 :
386 :
387 :
388 :
389 :
390 :
391 :
392 :
393 :
394 :
395 :
396 :
397 :
398 :
399 :
400 :
401 :
402 :
403 :
404 :
405 :
406 :
407 :
408 :
409 :
410 :
411 :
412 :
413 :
414 :
415 :
416 :
417 :
418 :
419 :
420 :
421 :
422 :
423 :
424 :
425 :
426 :
427 :
428 :
429 :
430 :
431 :
432 :
433 :
434 :
435 :
436 :
437 :
438 :
439 :
440 :
441 :
442 :
443 :
444 :
445 :
446 :
447 :
448 :
449 :
450 :
451 :
452 :
453 :
454 :
455 :
456 :
457 :
458 :
459 :
460 :
461 :
462 :
463 :
464 :
465 :
466 :
467 :
468 :
469 :
470 :
471 :
472 :
473 :
474 :
475 :
476 :
477 :
478 :
479 :
480 :
481 :
482 :
483 :
484 :
485 :
486 :
487 :
488 :
489 :
490 :
491 :
492 :
493 :
494 :
495 :
496 :
497 :
498 :
499 :
500 :
501 :
502 :
503 :
504 :
505 :
506 :
507 :
508 :
509 :
510 :
511 :
512 :
513 :
514 :
515 :
516 :
517 :
518 :
519 :
520 :
521 :
522 :
523 :
524 :
525 :
526 :
527 :
528 :
529 :
530 :
531 :
532 :
533 :
534 :
535 :
536 :
537 :
538 :
539 :
540 :
541 :
542 :
543 :
544 :
545 :
546 :
547 :
548 :
549 :
550 :
551 :
552 :
553 :
554 :
555 :
556 :
557 :
558 :
559 :
560 :
561 :
562 :
563 :
564 :
565 :
566 :
567 :
568 :
569 :
570 :
571 :
572 :
573 :
574 :
575 :
576 :
577 :
578 :
579 :
580 :
581 :
582 :
583 :
584 :
585 :
586 :
587 :
588 :
589 :
590 :
591 :
592 :
593 :
594 :
595 :
596 :
597 :
598 :
599 :
600 :
601 :
602 :
603 :
604 :
605 :
606 :
607 :
608 :
609 :
610 :
611 :
612 :
613 :
614 :
615 :
616 :
617 :
618 :
619 :
620 :
621 :
622 :
623 :
624 :
625 :
626 :
627 :
628 :
629 :
630 :
631 :
632 :
633 :
634 :
635 :
636 :
637 :
638 :
639 :
640 :
641 :
642 :
643 :
644 :
645 :
646 :
647 :
648 :
649 :
650 :
651 :
652 :
653 :
654 :
655 :
656 :
657 :
658 :
659 :
660 :
661 :
662 :
663 :
664 :
665 :
666 :
667 :
668 :
669 :
670 :
671 :
672 :
673 :
674 :
675 :
676 :
677 :
678 :
679 :
680 :
681 :
682 :
683 :
684 :
685 :
686 :
687 :
688 :
689 :
690 :
691 :
692 :
693 :
694 :
695 :
696 :
697 :
698 :
699 :
700 :
701 :
702 :
703 :
704 :
705 :
706 :
707 :
708 :
709 :
710 :
711 :
712 :
713 :
714 :
715 :
716 :
717 :
718 :
719 :
720 :
721 :
722 :
723 :
724 :
725 :
726 :
727 :
728 :
729 :
730 :
731 :
732 :
733 :
734 :
735 :
736 :
737 :
738 :
739 :
740 :
741 :
742 :
743 :
744 :
745 :
746 :
747 :
748 :
749 :
750 :
751 :
752 :
753 :
754 :
755 :
756 :
757 :
758 :
759 :
760 :
761 :
762 :
763 :
764 :
765 :
766 :
767 :
768 :
769 :
770 :
771 :
772 :
773 :
774 :
775 :
776 :
777 :
778 :
779 :
780 :
781 :
782 :
783 :
784 :
785 :
786 :
787 :
788 :
789 :
790 :
791 :
792 :
793 :
794 :
795 :
796 :
797 :
798 :
799 :
800 :
801 :
802 :
803 :
804 :
805 :
806 :
807 :
808 :
809 :
810 :
811 :
812 :
813 :
814 :
815 :
816 :
817 :
818 :
819 :
```

مثال ۲۰-۱۶ کارگاهی بطور متوسط سالی یک سفارش می‌گیرد. کارها را با تقدم متداول انجام می‌دهد. زمان کار متغیر تصادفی نرمال با میانگین ۵/۰ و انحراف معیار ۱ است در عین حال باید دستگاهها تنظیم و آماده باشد. زمان آماده سازی دستگاهها زمانی بین ۲/۰ تا ۵/۰ است. احتمال خرابی دستگاههای کاری نیز وجود دارد و دارای توزیع نرمال با میانگین و انحراف معیار ۲۰ و ۲ است. اگر کاری ناتمام بماند و دستگاه کاری خراب شود، تقدم می‌یابد و زمان باقیمانده آن در نظر گرفته می‌شود. عمل تعمیر در سه فاز که فاز هر فاز زمانی با میانگین ۳/۴، ۲/۳، ۱/۴ می‌شود. جمع این سه فاز نمائی مستقل با توزیع اولنگ می‌باشند و ۵۰۰ ساعت عمل شبیه‌سازی باید انجام می‌گیرد.

بخش اول مدل، ماشین کار منبع با نام TOOL تعریف شده است. پس از انتظار هر کار سرویسهای ۱ و ۲ روی آنها انجام می‌گیرد. در گره AWAIT منبع را اشغال و در گره FREE رها می‌کند. در گره COLCT آمارها جمع‌آوری می‌شوند. آمارها از زمان ورود مشتری (ATTRIB(1)) تا پایان پروسه انجام می‌شود که INT(1) زمان (TNOW-ATTRIB(1)) را خواهند داد.

بخش دوم مدل، خرابی دستگاه کار را مدل می‌کند و کارهای مربوطه که منبع را از دست داده‌اند به گره AWAIT باز گردانده می‌شوند. تعمیر در فعالیت ۳ انجام می‌گیرد.

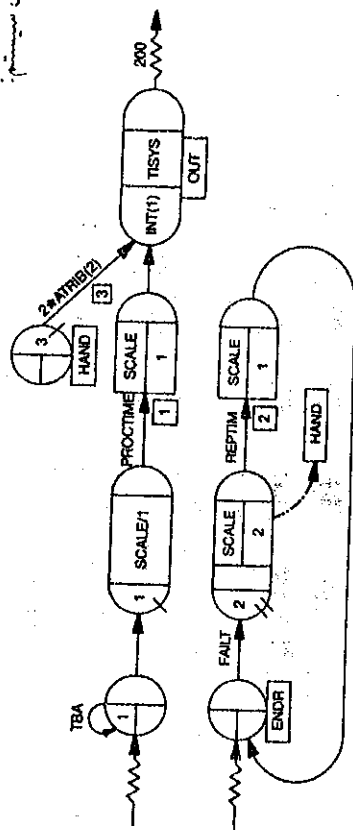
مدل سیستم:



صورت می‌گیرد. این دستگاه به وسیله منبع با نام SCALE و ظرفیت ۱ تعیین می‌شود. عملیات تعمیر این دستگاه در بخش دوم شبکه انجام می‌پذیرد (ENDR شروع آن است).

ورود بسته‌ها برای عملیات از گره CTEATE و فواصل زمانی TBA تولید می‌گردد. بسته‌هایی که منبع خود را از دست می‌دهند به گره HAND و پس از اتمام عملیات برای آماریگری به گره OUT می‌روند.

مدل سیستم:



برنامه سیستم:

```

NETWORK:
:
: NORMAL PROCESSING
:
: RESOURCE/SCALE(1),2,1;
: CREATE,TBA,,1;
: AWAIT(1),SCALE/1;
: ACT/1,PROCTIME;
: FREE,SCALE/1;
: COLCT,INT(1),TISYS;
: TERM,200;

OUT
:
: PREEMPTION OF SCALE
:
: ENDR
:
: CREATE:
: ACT,FAIT;
: PREEMPT SCALE, SEND ENTITY IN PROCESS TO 'HAND'
: PREEMPT(2),SCALE,HAND,2;
: ACT/2,REPTIME;
: FREE,SCALE/1;
: ACT,...ENDR;
:
: CAUSE SUBSEQUENT FAILURES
:
: PROCESSING OF PREEMPTED ENTITY
:
: HAND: QUEUE(3);
: ACT/3,2*ATTRIB(2),OUT;
: ENDNETWORK;

```

| نوع تانکر | درصد مراجعه | زمان بارگیری |
|-----------|-------------|--------------|
| ۱ | ۰/۲۵ | ۱۸±۲ |
| ۲ | ۰/۵۵ | ۲۴±۳ |
| ۳ | ۰/۲۰ | ۳۶±۴ |

یکی بدکشی برای ورود تانکرهای نفتکش به بندر و خروج از بندر تا بخش عمیق آب وجود دارد. زمان انتقال از اسکله به بخش عمیق آب و بالعکس ۱ ساعت است و روش بدکشی آزاد می شود انتقال کشتیها به اسکله نسبت به انتقال از اسکله قدم دارد. برای حمل نفت از بندر ۵ کشتی در نظر می گیریم که زمان بارگیری هر کشتی ۲۱±۳ ساعت است. پس از بارگیری زمان مسافرت (شامل تخلیه نفت) ۲۴±۳ ساعت می باشد. طوفانهای دریایی عملیات را مختل می کنند. زمان بین وقوع دو طوفان ۴±۲ است و در طوفان عملیات بدکشی متوقف می گردد. شبیه سازی به مدت ۸۴۶۰ ساعت صورت می پذیرد تا اثر افزودن کشتیها بر ترافیک بندر ارزیابی شود.

تانکرهای نفتی موجودیتهای سیستم محسوب و بدکشی و اسکله به صورت منابع با نامهای TUG با ظرفیت ۳ و BERTH با ظرفیت ۱ تعریف می شود. مدل از سه بخش تشکیل شده است. بخش اول نمایش عملکرد جاری می باشد. تانکرها در گره Create یا فواصل زمانی یکپارخت (4,18) ایجاد و به یکی از سه گره ASSIGN یا نامهای ARV1، ARV2 و ARV3 می روند. زمان بارگیری هر کشتی و نوع آن در مشخصه اول و دوم موجودیت تعیین می گردد. عملیات بارگیری از گره PORT آغاز می شود. قسمت دوم ۵ کشتی به ترافیک بندر اضافه می کند. مشخصه سوم برای زمان صرف شده برای هر کشتی در سیستم با متغیر TNOW تعیین می گردد. دو گره بعدی AWAIT برای آمادگی اسکله و بندر است. در صورت وجود بدکشی فعالیت انتقال به اسکله (به مدت یک ساعت) انجام شده و منابع TUG در گره های FREE آزاد می گردند. اگر بدکشی آزاد باشد انتقال عمیق آب پس از یک ساعت انجام و منبع BERTH و بعد از آن منبع TUG در گره های FREE آزاد می شوند. کشتیها برای جمع آوری آمار به یکی از گره های DT1,DT5 وارد می شوند.

بخش سوم طوفان و توقف حرکت بدکشی را مدل سازی می کند که با زمان نامانی و میانگین ۴۸ است. با ایجاد اولین موجودیت و ورود آن به گره ALTER از منبع TUG یک واحد کم می شود (تا به صفر برسد) و تا تداوم طوفان فعالیت بعد از گره ALTER تا یکپارخت، ظرفیت منبع بدکشی صفر

برنامه سیستم:

```

1 GEN,PRITSKER,MACHINE,BREAKDOWN,7/5/83,5,....,NO;
2 LIMITS,2,1,50;
3 NETWORK;
4 RESOURCE/TOOL(1)2,1;
5 CREATE,EXPON(1,1),0,1;
6 MAINT(1),TOOL/1;
7 ACT/1,UNFIRM(2,5);
8 GOON,1;
9 ACT/2,RNOBM(.5,1);
10 FREE,TOOL/1;
11 COLCT,INT(1),TIME,IN,SYSTEM;
12 TERM;
13 ;
14 CREATE,20,1;
15 DOWN,PREEMPT(2),TOOL;
16 ACT/3,ERLNG(.75,3,0);
17 FREE,TOOL/1;
18 ACT,RNOBM(20,2),DOWN;
19 END;
20 INIT,0,500;
21 FIN;

```

CREATE ARRIVALS
AWAIT THE TOOL
SET UP
PROCESSING
FREE THE TOOL
COLLECT STATISTICS
CREATE 1ST BREAKDOWN
PREEMPT THE TOOL
DOWN TIME
FREE THE TOOL
TIME BETWEEN FAILURES

نتایج خروجی: حاصل از ۵ بار اجرا می باشد.

| Run Number | Average Time in System | Average Number of Jobs Waiting | In-Use | Tool Status Percentages | In-Repair |
|------------|------------------------|--------------------------------|--------|-------------------------|-----------|
| 1 | 12.06 | 12.89 | 90 | 1 | 9 |
| 2 | 7.46 | 6.78 | 86 | 4 | 10 |
| 3 | 6.39 | 5.63 | 86 | 4 | 10 |
| 4 | 5.33 | 4.35 | 82 | 7 | 11 |
| 5 | 9.48 | 9.68 | 82 | 7 | 11 |

مثال ۲۱-۱۶

عملیات بارگیری نفت خام یکی بندر توسط تانکرها انجام می پذیرد. بطور همزمان عملیات بارگیری ۳ تانکر می تواند صورت گیرد. زمان رسیدن تانکرها به بندر با فواصل زمانی ۱۱±۷ ساعت است و مشخصات آنها به شرح زیر می باشد:

برنامه سیستم:

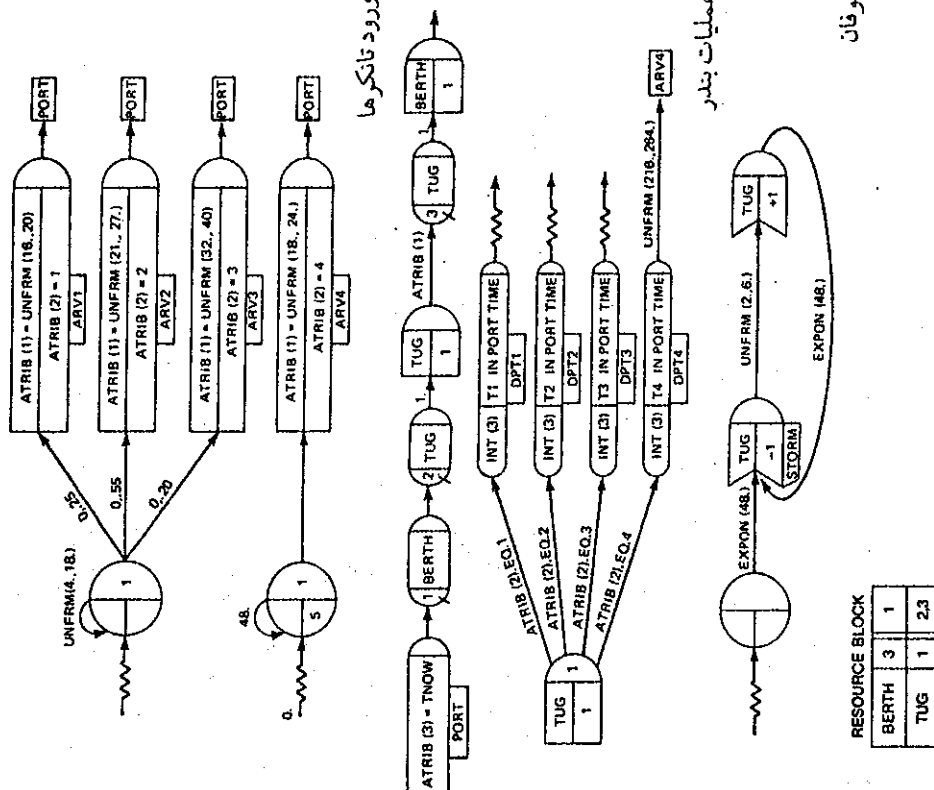
```

1 GEN PRITSKEV, AFRICA PORT, 5/21/83, 1;
2 LIMITS, 3, 3, 50;
3 NETWORK;
4 RESOURCE/BERTH(3), 1;
5 RESOURCE/TUG(1), 2, 3;
6 ;
7 ; TANKER ARRIVAL SEGMENT
8 ;
9 CREATE, UNFRM(4, 18, .);
10 ACT, ., 25, ARV1;
11 ACT, ., 55, ARV2;
12 ACT, ., 20, ARV3;
13 ARV1 ASSIGN, ATRIB(1) = UNFRM(16, 20, .), ATRIB(2) = 1;
14 ACT, ., PORT;
15 ARV2 ASSIGN, ATRIB(1) = UNFRM(21, 27, .), ATRIB(2) = 2;
16 ACT, ., PORT;
17 ARV3 ASSIGN, ATRIB(1) = UNFRM(32, 40, .), ATRIB(2) = 3;
18 ACT, ., PORT;
19 CREATE, 48, 0, 5;
20 ARV4 ASSIGN, ATRIB(1) = UNFRM(18, 24, .), ATRIB(2) = 4;
21 ;
22 ; PORT OPERATION SEGMENT
23 ;
24 PORT
25 ASSIGN, ATRIB(3) = TNOW;
26 ATRIB(1), BERTH/1;
27 ATRIB(2), TUG/1;
28 ACT, 1;
29 FREE, TUG/1;
30 ACT, ATRIB(1);
31 ATRIB(3), TUG/1;
32 ACT, 1;
33 FREE, BERTH/1;
34 FREE, TUG/1;
35 ACT, ATRIB(2), EQ, 1, DPT1;
36 ACT, ATRIB(2), EQ, 2, DPT2;
37 ACT, ATRIB(2), EQ, 3, DPT3;
38 ACT, ATRIB(2), EQ, 4, DPT4;
39 DPT1 COLCT, INT(3), T1 IN PORT TIME;
40 DPT2 COLCT, INT(3), T2 IN PORT TIME;
41 DPT3 COLCT, INT(3), T3 IN PORT TIME;
42 DPT4 COLCT, INT(3), T4 IN PORT TIME;
43 TERM;
44 ACT, UNFRM(216, 264, .), ARV4;
45 ;
46 ; STORM SEGMENT
47 ;
48 CREATE;
49 ACT, EXPON(48, .);
50 STORM ALTER, TUG/-1, 1;
51 ACT, UNFRM(2, 6, .);
52 ALTER, TUG/+1;
53 ACT, EXPON(48, .), STORM;
54 END;
55 INIT, 0, 8640;
56 FIN;

```

REQUEST TUG CAPACITY DECREASE
DURATION OF THE STORM
REQUEST TUG CAPACITY INCREASE
SCHEDULE THE NEXT STORM

است و با ورود موجودیت به گره ALTER بدوی ظرفیت افزایش می‌یابد. موجودیتهای مربوطه به گره STORM باز می‌گردند و طوفان بعدی را باعث می‌شوند.
مدل سیستم:



(ج) بخش طوفان

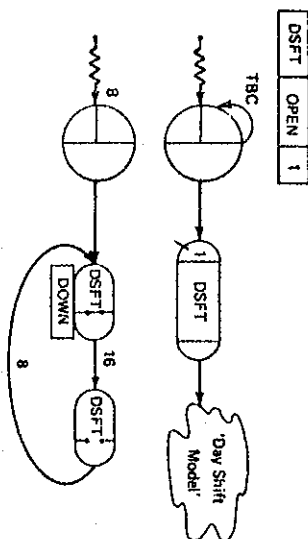
(ب) بخش عملیات بندر

(الف) بخش ورود فانکرها

مثال ۱۶-۲۲

عملیات پستی روی بسته‌های اداره پست در طول ۲۴ ساعت انجام می‌شود. عملیات بررسی، چسباندن تمبر و بارگیری بسته‌ها صورت می‌گیرد. بسته‌های پستی به عنوان مشتری تلقی می‌شوند و با فواصل زمانی به سیستم وارد می‌گردند. چنانچه مسئول شیفت نباشد و اجازه ندهد (دروازه DSFT) در گره AWAIT منتظر می‌مانند و پس از باز شدن این دروازه به مدل شیفت دروازه (مدل SLAM این بخش ترسیم نشده است) سپرده می‌شوند. پس از ۱۶ ساعت فعالیت دروازه بسته DSFT باز می‌گردد و پس از ۸ ساعت کار کردن (شیفت کاری) بسته می‌شود.

مدل سیستم:



برنامه سیستم:

NETWORK:
GATE/DSFT,OPEN,1;
CREATE,TBC;
AWAIT(1),DSFT;
SHIFT MODEL;
DAY
CREATE,8;
CLOSE,DSFT;
DOWN
ACTIVITY,16;
OPEN,DSFT;
ACTIVITY,8,DOWN;
ENDNETWORK;

مثال ۲۳-۱۶ بررسی و مدل‌سازی یک خیابان دو طرفه که در بخشی از آن به علت تعمیرات عبور و مرور به یک طرفه با تقسیم زمانی تبدیل شده را داریم (یک جهت در هر بخشی زمانی). برای کنترل عملیات دو چراغ راهنمایی (۱ و ۲) نصب شده است. فاصله زمانی ورود ماشینها از جهت شماره ۱ و

نتیجه خروجی:

| SLAM II SUMMARY REPORT | | | | | | | | | |
|---|------------------------|--------------------|---------------------|----------------|----------------|--|---------|------------------------|--|
| SIMULATION PROJECT AFRICA PORT | | | | | | BY PRATISHER | | | |
| DATE 5/21/1983 | | | | | | RUN NUMBER 1 OF 1 | | | |
| CURRENT TIME 0.000000 | | | | | | STATISTICAL ABBREVS CLEARED AT TIME 0.000000 | | | |
| ***STATISTICS FOR VARIABLES BASED ON OBSERVATION*** | | | | | | | | | |
| NAME | VALUE | STANDARD DEVIATION | COEFF. OF VARIATION | HIGH | MIN | MAX | AVERAGE | NUMBER OF OBSERVATIONS | |
| IN PORT TIME | 0.3007E+02 | 0.1462E+02 | 0.4828E+00 | 0.1808E+02 | 0.0000E+00 | 0.6534E+02 | 14.8211 | 183 | |
| IN PORT TIME | 0.4132E+02 | 0.1594E+02 | 0.3702E+00 | 0.2340E+02 | 0.0000E+00 | 0.5989E+02 | 0.2831 | 449 | |
| IN PORT TIME | 0.5258E+02 | 0.1588E+02 | 0.2831E+00 | 0.3407E+02 | 0.1019E+03 | 0.1019E+03 | 0.2844 | 142 | |
| IN PORT TIME | 0.5802E+02 | 0.1405E+02 | 0.3042E+00 | 0.2005E+02 | 0.1008E+03 | 0.1008E+03 | 8.3102 | 153 | |
| ***FILE STATISTICS*** | | | | | | | | | |
| FILE NUMBER | ASSOC. NODE LABEL/TYPE | AVERAGE LENGTH | STANDARD DEVIATION | MAXIMUM LENGTH | CURRENT LENGTH | AVERAGE WAITING TIME | | | |
| 1 | ANALYT | 1.8055 | 1.8801 | 8 | 2 | 14.8211 | | | |
| 2 | ANALYT | 0.0215 | 0.1818 | 2 | 0 | 0.2831 | | | |
| 3 | ANALYT | 0.0035 | 0.1834 | 3 | 0 | 0.2844 | | | |
| 4 | CALCULOR | 8.0028 | 0.7388 | 11 | 8 | 8.3102 | | | |

FILE STATISTICS

| FILE NUMBER | ASSOC. NODE LABEL/TYPE | AVERAGE LENGTH | STANDARD DEVIATION | MAXIMUM LENGTH | CURRENT LENGTH | AVERAGE WAITING TIME |
|-------------|------------------------|----------------|--------------------|----------------|----------------|----------------------|
| 1 | AWAIT | 1.8835 | 1.8831 | 8 | 2 | 14.8211 |
| 2 | AWAIT | 0.1015 | 0.1818 | 2 | 0 | 0.2831 |
| 3 | AWAIT | 0.1035 | 0.1834 | 3 | 0 | 0.2844 |
| 4 | CUSTOM | 8.0828 | 0.7288 | 11 | 8 | 8.3102 |

RESOURCE STATISTICS

| RESOURCE NUMBER | RESOURCE LABEL | CURRENT CAPACITY | AVERAGE UTILIZATION | STANDARD DEVIATION | MAXIMUM UTILIZATION | CURRENT UTILIZATION |
|-----------------|----------------|------------------|---------------------|--------------------|---------------------|---------------------|
| 1 | BERTH | 3 | 2.8745 | 0.3731 | 3 | 3 |
| 2 | TUG | 1 | 0.2148 | 0.4107 | 1 | 1 |

| RESOURCE NUMBER | RESOURCE LABEL | CURRENT AVAILABLE | AVERAGE AVAILABLE | MINIMUM AVAILABLE | MAXIMUM AVAILABLE |
|-----------------|----------------|-------------------|-------------------|-------------------|-------------------|
| 1 | BERTH | 0 | 0.1255 | 0 | 3 |
| 2 | TUG | 0 | 0.7073 | -1 | 1 |

بخش اول نتایج از گره‌های COLLECT است و مدت زمانی را نشان می‌دهد که هر یک از کشتی‌ها در سیستم سیری می‌کند. نتایج فایلهای ۲۰۱ و ۲ در بخش دوم ارائه و بخش آخر آمار منابع BERTH و TUG است. آمار نشان می‌دهد که بطور متوسط ۲/۸۷۴۵ از سه بندر در حال استفاده می‌باشند در حالی که فقط ۲۱/۴۸ درصد زمان یک‌کش مشغول عملیات است. بنابراین زمان انتظار کشتی‌ها در بندر خیلی زیاد می‌باشد.

برنامه سیستم:

```

1 GEN, OREILLY, TRAFFIC LIGHTS, 7/12/83, 3, NO., NO.;
2 LIMITS, 4, 3, 100;
3 INTLC, XX(1)=60, XX(2)=45;
4 NETWORK;
5 RESOURCE/START1, 1/START2, 2;
6 GATES/LIGHT1, CLOSE, 3/LIGHT2, CLOSE, 4;
7 :TRAFFIC LANES
8 :
9 CREATE, EXPON(9), 1;
10 ASSIGN, ATRIB(2)=1, ATRIB(3)=3;
11 ACT, ...QUE;
12 CREATE, EXPON(12), 1;
13 ASSIGN, ATRIB(2)=2, ATRIB(3)=4;
14 QUEUE, ATRIB(2)=1, 2, ATRIB(2);
15 WAIT, ATRIB(3)=3, 4, ATRIB(2);
16 COLCP, INT(1), WAIT TIME, 1;
17 ACT, 2, TNOW, GT, ATRIB(1);
18 ACT;
19 FREE, ATRIB(2);
20 TERM;
21 :TRAFFIC LIGHTS
22 :
23 CREATE, ...1;
24 ACT, 55;
25 LOOP
26 OPEN, LIGHT1;
27 ACT, XX(1);
28 CLOSE, LIGHT1;
29 ACT, 55;
30 OPEN, LIGHT2;
31 ACT, XX(2);
32 CLOSE, LIGHT2;
33 ACT, 55, LOOP;
34 END;
35 INIT, 0, 3600;
36 SIMULATE;
37 INTLC, XX(1)=80, XX(2)=60;
38 SIMULATE;
39 INTLC, XX(1)=40, XX(2)=30;
40 FIN;

```

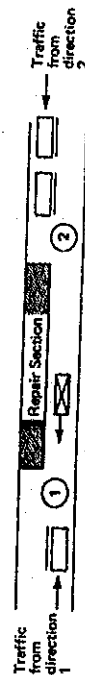
با ورودی‌های مشخص زیر به عنوان سه مقدار $xx(1)$ و $xx(2)$ برای سه بار اجرای استخراج شده است:

| Run Number | XX(1) | XX(2) |
|------------|-------|-------|
| 1 | 60 | 45 |
| 2 | 80 | 60 |
| 3 | 40 | 30 |

| Run Number | Green Times | | Average Waiting Time | |
|------------|-------------|-------------|----------------------|-------------|
| | Direction 1 | Direction 2 | Direction 1 | Direction 2 |
| 1 | 60 | 45 | 62.35 | 74.92 |
| 2 | 80 | 60 | 66.06 | 75.79 |
| 3 | 40 | 30 | 185.43 | 185.30 |

نتایج:

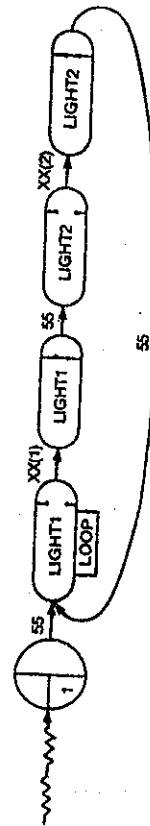
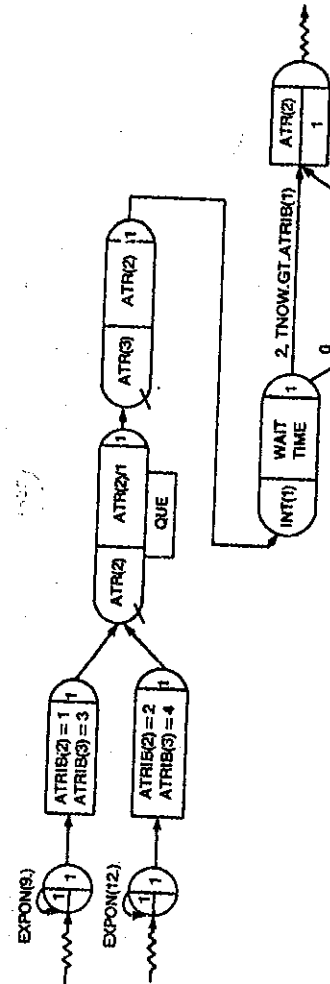
شماره ۲ تابع توزیع نمایی با میانگین ۹ و ۱۲ است. چراغ‌ها حالت‌های مختلف دو رنگ سبز و قرمز را تداعی می‌کنند (هر دو سبز، هر دو قرمز، یکی سبز و یکی قرمز). مدت زمان قرمز بودن هر دو چراغ ۵۵ ثانیه است. و ماشین‌ها برای شروع حرکت ۲ ثانیه معطلی ایجاد می‌کنند.



Legend:
 (1) traffic light
 [] waiting car
 [X] car in transit

دیگرام کنترل نور

مدل سیستم:



| RESOURCE BLOCK | | GATE BLOCK | |
|----------------|---|------------|---|
| START1 | 1 | LIGHT1 | C |
| START2 | 2 | LIGHT2 | C |

فصل هفدهم

گروه‌های تصمیم‌گیری و منطقی

بنچ گروه در زبان SLAM II وجود دارند که عملیات منطقی و تصمیم‌گیری را شکل می‌دهند. گروه ACCUMULATE برای جمع‌آوری تعداد خاصی از موجودیتها است. گروه BATCH تصمیم‌یافته گروه ACCUMULATE می‌باشد و اجازه می‌دهد که هر موجودیت در یک دسته نگهداری شود. گروه UNBATCH عکس گروه BATCH عمل می‌کند و با ورود یک موجودیت به آن چندین موجودیت از آن خارج می‌گردد. گروه MATCH جمع‌آوری و هدایت موجودیت‌های با مشخصات یکسان و هدایت به گروه دیگری را انجام می‌دهد. گروه SELECT برای مسیریابی هدفمند است و قوانینی برای انتخاب از یک گروه QUEUE از میان تعدادی گروه موازی QUEUE می‌باشد. یکی از این قوانین برای گروه SELECT، ASSEMBLY است که یک انتخاب براساس موجودیت‌های موجود از یک سری گروه‌های QUEUE مختلف و مرتبط آنهاست.

۱.۷ اگره تجمع ACCUMULATE

| | | |
|----|------|---|
| FR | SAVE | M |
| SR | | |

ACCUMULATE.FR,SR,SAVE,M;

شکل ۱-۱۷ سیمای گرافیکی گروه تجمع ACCUMULATE و دستور SLAM

باعث تجمع موجودیتها می‌گردد و وقتی تعداد مشخصی موجودیت وارد گروه شود یک موجودیت را از گروه رها می‌کند.

پارامترها:

FR: تعداد اولین تجمع که یک نفر به ازای آنها آزاد شود.

SR: تعداد تجمع‌های بعدی که یک نفر به ازای آنها آزاد شود.

SAVE: مشخصه موجودیت خروجی چه چیزی تعیین شود؟ و می‌توان:

اگر تعدادی موجودیت وارد گروه BATCH شوند که جمع مشخصه با شماره NATRS آنها بزرگتر یا مساوی مقدار معین TRESH باشد، آنگاه یک موجودیت به نمایندگی از موجودیتهای دسته از گروه خارج می‌شود. اگر مجموع ATTRIB(NATRB) مربوط به ۵ عضو یکدسته ۷ و مقدار TRESH=10 باشد، چنانچه موجودیت ورودی ششم دارای مشخصه ATTRIB (NATRB) با مقدار منفی است، موجودیت نماینده دسته از گروه خارج می‌شود (خروج موجودیت با عملکرد مقدار منفی). RETAIN باعث نگهداری موجودیتهای عضو هر دسته شده تا در مواقع ضروری با ارسال موجودیت دسته به گروه UNBATCH آن موجودیتهای آزاد شوند. مثلاً با تعیین ALL(NATR) برای پارامتر RETAIN باعث می‌شود که موجودیتهای عضو دسته که از گروه UNBATCH رها می‌شوند، همگی دارای مقدار مساوی ATTRIB(NATR) باشند و NONE برای RETAIN یعنی هیچکدام موجودیتهای نگهداری نمی‌شوند. دامنه عملکرد این گروه بسیار وسیع است.

مثال ۱۷-۲

| | |
|---------|---|
| 100 | 3 |
| FIRST/4 | 1 |
| NONE | |

BATCH, 5/2, 100, 3, FIRST/4, NONE, 1;

حداکثر ۵ دسته مختلف از موجودیتهای نگهداری می‌شوند و موجودیتهایی که ATTRIB(2) آنها برابر است در یک دسته قرار می‌گیرند. اگر مجموع ATTRIB(3) موجودیتهای عضو هر دسته بزرگتر یا مساوی ۱۰۰ باشد، موجودیت دسته از گروه خارج می‌شود و به استثنای ATTRIB(4) بقیه مشخصه‌های موجودیت خروجی برابر اولین موجودیت عضو هر دسته می‌باشد. مشخصه‌های موجودیتهای عضو دسته ذخیره نمی‌شود (NONE).

مثال ۱۷-۳

| | |
|--------------|--|
| 18 | |
| LOW(1)/2,3,4 | |
| ALL(5) | |

BATCH, 18, LOW(1)/2,3,4, ALL(5);

به علت عدم تعریف NBATCH فقط یک دسته از موجودیتهای تشکیل می‌شود و وقتی نماینده دسته خارج می‌شود که اعضای دسته به ۱۸ رسیده باشد. عدم تعریف NATR باعث شمارش موجودیتهای دسته‌ها می‌شود. مشخصه‌های دوم، سوم و چهارم (موجودیت) نماینده خروجی برابر همین مشخصه‌های کلیه موجودیتهای عضو دسته می‌باشد و مشخصه‌های اول و پنجم برابر کمترین مقدار از

FIRST: مشخصه اولین موجودیت ورودی هر دسته تجمع انتخاب شود.
LAST: مشخصه آخرین موجودیت ورودی هر دسته تجمع انتخاب شود.
HIGH(1): موجودیتی که بیشترین مقدار بردار مشخصه ATTRIB(1) را دارد انتخاب گردد.
LOW(1): موجودیتی که کمترین مقدار بردار مشخصه ATTRIB(1) را دارد انتخاب گردد.

مثال ۱۷-۱

| | | |
|---|------|----------------|
| 2 | LAST | ACC,2,1, LAST; |
|---|------|----------------|

در اولین تجمع به ازای ۲ موجودیت یک خروجی و در تجمع‌های بعدی هر ۱ موجودیت یک خروجی و مشخصه خروجی آن مشخصه آخرین ورودی به دسته باشد.

۱۷-۲- گروه دسته‌ای BATCH

| | |
|--------|-------|
| THRESH | NATRS |
| SAVE | |
| RETAIN | N |

BATCH, NBATCH/NATRB, THRESH, NATRS, SAVE, RETAIN, M;
شکل ۱۷-۲: سبیل گرافیکی گروه BATCH و دستور SLAM

در این گروه موجودیتهای در یک دسته تجمع می‌کنند و امکان رها شدن موجودیتهای به صورت دسته‌ای نیز وجود دارد که شکل تعمیم یافته گروه ACCUMULATE می‌باشد (سپس تجمع تعدادی خروجی دارد).

پارامترها:

NBATCH: تعداد کل دسته‌های ایجاد می‌گردد.
NATRB: شماره خانه بردار مشخصه معروف موجودیتهای عضو هر دسته یعنی ATTRIB(NATRB) موجودیتهایی که شماره مشخصه یکسانی دارند در یک دسته قرار می‌گیرند.
THRESH: آستانه، تعداد تجمع موجودیتهای جهت خروج از آن دسته.
NATRS: شماره مشخصه‌ای که جمع مقادیر را نگه دارد.
SAVE: مانند شرح گروه ACCUMULATE است.
RETAIN: تعیین موجودیتهای خاصی که در یک دسته باقی بماند (NONE) و (ALL, NATR).

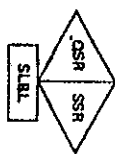


UNBATCH,3,1;

مثال ۱۷-۵

اگر مقدار ATTRIB(3) موجودیت ورودی برنامه ساز تعیین شود آنگاه به ازای مقدار آن، کپی از موجودیت‌های ورودی خارج می‌شوند ولی اگر تعیین نشود، موجودیت‌های دسته گره BATCH از UNBATCH خارج می‌گردد.

۱۷.۴- گره انتخاب SELECT



SLBL SELECT,OSR,SSR,BLOCK or BALK(NLBL),QLBLS;

شکل ۱۷-۴ سبیل گرافیکی گره SELECT

جهت تصمیم‌گیری برای انتخاب از میان چند فعالیت سروس با چند صف بکار می‌رود. برای انتخاب فعالیت سروس از (قانون انتخاب سروس دهند) SSR و برای انتخاب صف از (قانون انتخاب صف) OSR استفاده می‌کند.

موارد استفاده:

- ۱- اختصاص یک موجودیت به یکی از صف‌های موجود (چندین صف).
 - ۲- انتخاب موجودیت جهت سروس در یک سیستم با چندین سروس دهنده و چند صف موازی.
 - ۳- اختصاص یک موجودیت در یک سیستم با چندین سروس دهنده موازی.
- علامت گرافیکی شامل ۲ بخش است، زمانی که موارد استفاده ۱ و ۳ پیش‌آید از علامت گرافیکی استفاده می‌شود که معنی آن است که صف‌ها براساس قانون OSR موجودیت دریافت می‌کنند. یعنی اختصاص موجودیت به سروس دهنده براساس قانون SSR است. زمانی که انتخاب SELECT براساس گره‌های قبلی باشد از علامت گرافیکی OSR استفاده می‌کنیم، در این حالت ۲ (یعنی انتخاب موجودیت از چندین صف) است (با قانون OSR). گره هر دو حالتها را می‌تواند داشته باشد.

پارامترها:

OSR: قانون انتخاب صف.

SSR: قانون انتخاب سروس دهنده.

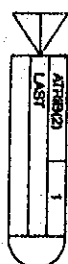
BLOCK: بلوک که شدن موجودیتها و سروس دهند.ها.

NLBL: هدایت به گره B: ALK(NLBL)

QLBLS: نام گره‌های صف مرتبط با گره SELECT

موجودیتها می‌گردند. در مشخصه پنجم مرتب از آدرس‌های موجودیت‌های هر دسته را تعیین می‌کند

ALL(5)



مثال ۱۷-۴

BATCH,ATTRIB(2),1,LAST;

اگر مجموع ATTRIB(1) موجودیت‌های دسته بزرگتر یا مساوی ATTRIB(2) اولین موجودیت ورودی به گره باشد، نماینده از گره خارج می‌شود و تمامی مشخصه‌های وی برابر آخرین موجودیت دسته می‌گردد.

۱۷.۳- گره غیردسته‌ای UNBATCH



UNBATCH,NATTR,M;

شکل ۱۷-۳ سبیل گرافیکی گره UNBATCH و دستور SLAM

با ورود یک مشتری به این گره چندین مشتری می‌تواند از آن خارج گردد و عکس عمل

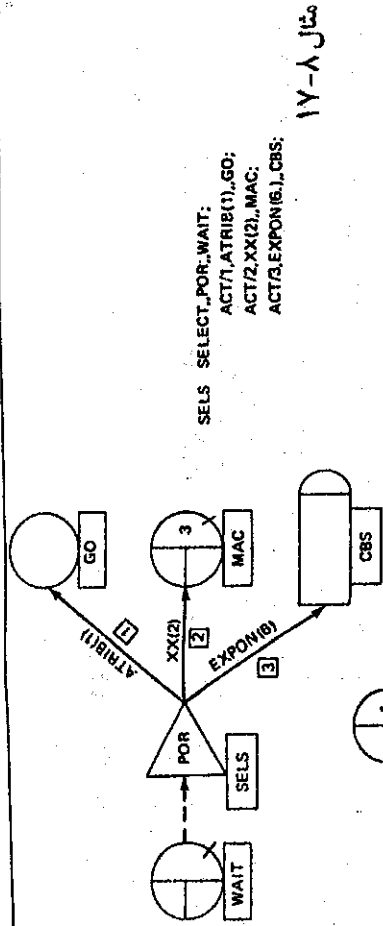
است. BATCH

پارامتر: NATTR: شماره مشخصه که تعیین می‌کند کلیه موجودیت‌های گره BATCH از UNBATCH آزاد شوند و با تعداد مشخصی مشابه موجودیت ورود به گره، خارج شود.

اگر برای ATTRIB(NATTR) مقدار تعیین کنیم (در گره ASSIGN مقداردهی شود)، لذا وقتی موجودیتی وارد گره UNBATCH شود به تعداد ATTRIB(NATTR) موجودیت مشابه خود تبدیل می‌گردد. ولی اگر برای ATTRIB(NATTR) مقدار تعیین نشود، کلیه موجودیت‌های گره BATCH به صورت یک دسته از گره UNBATCH خارج و موجودیت ورودی گره از بین می‌رود. اگر بخواهیم ترتیب خروج موجودیت عضو هر دسته عکس ورود آنها به گره BATCH نشود باید پس از گره یک فعالیت با زمان سروس صفر قرار دهیم.

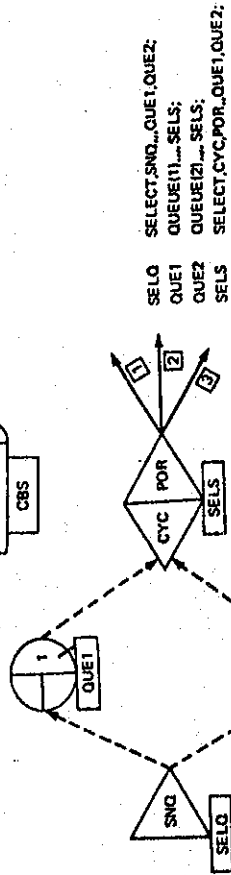
اگر $ATTRIB(NATTR) = 30$ باشد و مقدار $M = 5$ باشد امکان تقسیم یک موجودیت به تعداد دیگر فراهم

باشد، با ورود موجودیت به گره UNBATCH به تعداد ۱۵۰ موجودیت از گره خارج می‌شود.



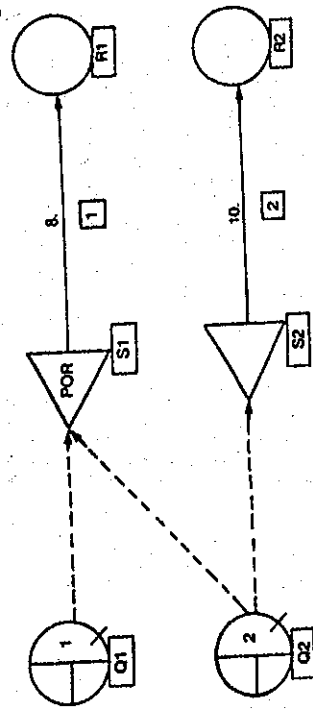
SELS SELECT_POR_WAIT;
ACT1/ATTRIB(1),GO;
ACT2/XX(2),MAC;
ACT3/EXPON(6),CBS;

مثال ۱۷-۸



در ترکیب ۲ مثال قبل، ابتدا از بین دو صف و به ترتیب تقدم QUE1 و QUE2 انتخاب می‌کند. سپس به ترتیب به سرورس دهنده‌های ۱ و ۲ تخصیص می‌دهد و گره SELO ورودی را به ترتیب (براساس تمرین مقدار صف) به صفهای QUE1 و QUE2 ارسال می‌کند (در صورت تساوی صفها، صف QUE1 الویت دارد). این ترتیب انتخاب QUE1 و سپس QUE2 به صورت دورهای می‌چرخد و پس از انتخاب QUE2، QUE1 انتخاب می‌گردد.

مثال ۱۷-۹



Q1 QUE1(1)...S1;
Q2 QUE2(2)...S2;
S1 SEL_POR_Q1_Q2;
ACT1/S...R1;
S2 SEL...Q2;
ACT2/10...R2;

تذکرات:

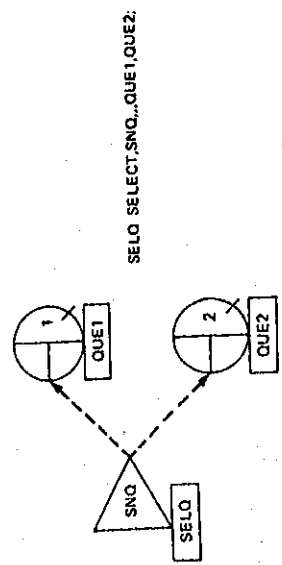
- ۱- در دو طرف یک گره SELECT نمی‌توان از گره‌های صف استفاده کرد.
- ۲- در صورت قرار گرفتن فعالیت‌های سرورس پس از گره SELECT، قبل از آن حتماً باید گره صف جهت انتظار موجودیتها باشد.
- ۳- عمل BLACK یا BLOCK زمان انجام می‌شود که تمامی گره‌های صف بعد از گره SELECT پُر شده باشند.



شکل ۱۷-۵: سبل گرافیکی BLACK یا BLOCK در گره SELECT

- ۴- چنانچه گره SELECT با گره‌های ماقبل خود ارتباط دارد، باید در گره ماقبل آن اسم گره SELECT ذکر شود (در تعاریف ابتدائی صف آمده است).
- ۵- باید برای گره SELECT نام تعریف کرد (SLBL).

مثال ۱۷-۶: تخصیص موجودیتها به صف کوتاهتر (QUE1 یا QUE2).

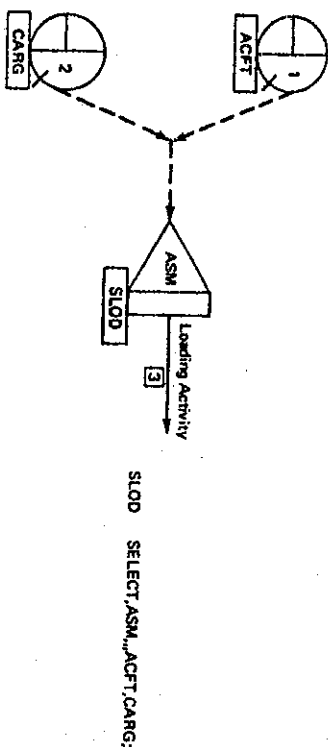


مثال ۱۷-۷: انتخاب یکی از سه سرورس دهنده به ترتیب تقدم (۲، ۱ و سپس ۳).

قانون ASSEMBLY در انتخاب موجودیت از صف

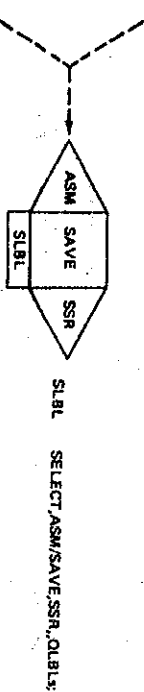
از قوانین انتخاب صف قانون ASM (موتاز) است (از جدول ۱-۱۲). در واقع همان قانون اسمبل (موتاز) را در بر دارد که در این حالت چندین موجودیت در یک موجودیت موتاز می‌گردند. در این قانون ورودیها از بین می‌روند و یک خروجی تکامل یافته ایجاد می‌شود. مثالهایی مانند موتاز قطعات یک کامپیوتر، موتاز یک تلویزیون، سوار شدن مسافر و بار همراه وی به هواپیما و غیره را مثال زد. در این حالت چنانچه حداقل یک مشتری در صفهای ورودی به قانون ASM موجود باشد با برداشتن موتاز آنها یک موجودیت خروجی ایجاد می‌کند.

مثال ۱۷-۱۰



چنانچه هواپیما در جهت ACFT حاضر باشد و بارها در صف CARG موجود باشند، فعالیت بارگیری (3) انجام می‌شود.

مثال ۱۷-۱۱



چنانچه بخوابم بر دار مشخصه گره خروجی تابعی از موجودیتهای ورودی و همراه وی در شبکه جریان باید از بخش SAVE استفاده می‌کنیم و مانند گره ACCUMULATE انتخابهای (LOW)، SUM، HIGH و MULT یکار می‌رود. اگر این بخش (SAVE) تعریف نشود. بر دار مشخصه اولین موجودیت صف به عنوان بر دار مشخصه موجودیت خروجی تعیین می‌شود.

صف ۲ در سروس دهنده ۱ و ۲ دارد ولی صف ۱ فقط سروس دهنده شماره ۱ را دارد. گره S1 از این ۲ صفت انتخاب می‌کند و گره S2 فقط صف ۲ را در S1 چون POR آمده است یعنی به ترتیبی که در شبکه ظاهر شده اول ۱ و سپس ۲. گره S1 نسبت به S2 تقدم اجرا دارد.

جدول ۱-۱۷ و ۱۷-۲ لیست قوانین انتخاب صف انتخاب سروس دهنده (فعالیت سروس) را دارد.

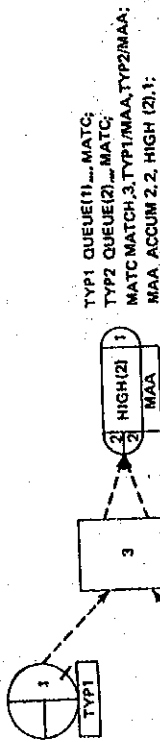
| Code | Definition |
|--------|--|
| POR | Priority given in a preferred order. |
| CYC | Cyclic Priority—transfer to first available QUEUE node starting from the last QUEUE node that was selected. |
| RAN | Random Priority—assign an equal probability to each QUEUE node that has an entry in it. |
| LAV | Priority given to the QUEUE node which has had the largest average number of entities in it to date. |
| SAV | Priority is given to the QUEUE node which has had the smallest average number of entities in it to date. |
| LWF | Priority is given to the QUEUE node for which the waiting time of its first entity is the longest. |
| SWF | Priority is given to the QUEUE node for which the waiting time of its first entity is the shortest. |
| LNO | Priority is given to the QUEUE node which has the current largest number of entities in the queue. |
| SNO | Priority is given to the QUEUE node which has the current smallest number of entities in the queue. |
| LRC | Priority is given to the QUEUE node which has the largest remaining unused capacity. |
| SRC | Priority is given to the QUEUE node which has the smallest remaining unused capacity. |
| ASM | Assembly mode option—all incoming queues must contribute one entity before a processor may begin service (this can be used to provide an "AND" logic operation). |
| NOS(N) | User written function to select a QUEUE node. N is an integer to differentiate between the use of NOS at different SELECT nodes (see Chapter 9). |

جدول ۱-۱۷ قوانین تقدم انتخاب از صف

| Code | Definition |
|--------|--|
| POR | Select from free servers in a preferred order. |
| CYC | Select servers in a cyclic manner. That is, select the next free server starting with the last server selected. |
| LBT | Select the server that has the largest amount of usage (busy time) to date. |
| SBT | Select the server which has the smallest amount of usage (busy time) to date. |
| LIT | Select the server who has been idle for the longest period of time. |
| SIT | Select the server who has been idle for the shortest period of time. |
| RAN | Select randomly from free servers according to preassigned probabilities. |
| NSS(N) | User written function to select a server. N is an integer to differentiate between the use of NSS at different SELECT nodes (see Chapter 9). |

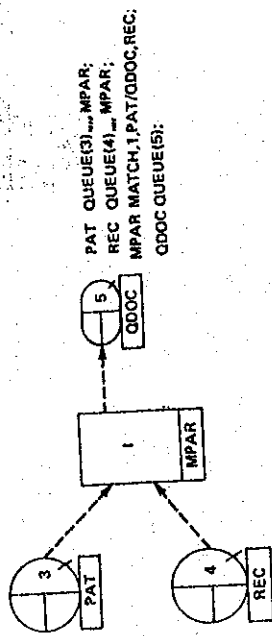
جدول ۲-۱۷ قوانین تقدم انتخاب فعالیتهای سروس

مثال ۱۷-۱۳



در گره MATCH کلیه موجودیهای گره ATRIB(3) آنها مشابه هستند به گره ACCUMULATE (هر دو دسته مشترک) وارد می‌شوند و مشخصه موجودیهای خروجی، مشخصه دوم بیشترین مقدار از موجودیهای ورودی انتخاب می‌شود.

مثال ۱۷-۱۴



۱۷.۶- مدل‌سازی با مدل SLAM

مثال ۱۷-۱۵ شبکه PERT یک پروژه تغییر و نصب طبق شکل زیر می‌باشد. شرح فعالیتها و زمانهای آن آمده است که زمان فعالیتها تصادفی با توزیع منظمی فرض شده است، سرویسها در شکل مشخص هستند.

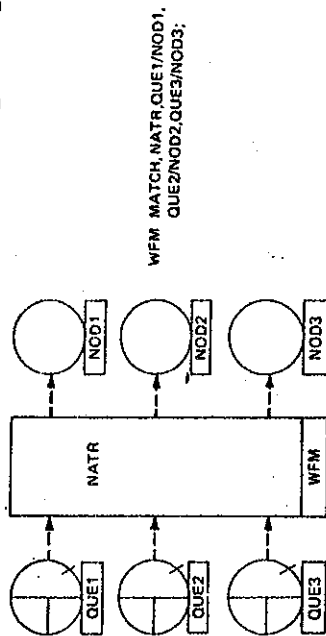
| Activity Number | Description | Mode | Minimum | Maximum | Average |
|-----------------|---|------|---------|---------|---------|
| 1 | Disassemble power units and instrumentation | 3 | 1 | 5 | 3 |
| 2 | Install new assembly | 6 | 3 | 9 | 6 |
| 3 | Prepare for retrofit check | 13 | 10 | 19 | 14 |
| 4 | Clean, inspect, and repair power units | 9 | 3 | 12 | 8 |
| 5 | Calibrate instrumentation | 3 | 1 | 8 | 4 |
| 6 | Check interfaces | 9 | 8 | 16 | 11 |
| 7 | Check assembly | 7 | 4 | 13 | 8 |
| 8 | Assemble and test power units | 6 | 3 | 9 | 6 |
| 9 | Retrofit check | 3 | 1 | 8 | 4 |

مثال ۱۷-۱۲

SLOD SELECT, ASMILOW(3), ACFT, CARG;

در این حالت خانه سوم بردار موجودیت صف ACFT (هواپیما) ATRIB(3) برای موجودیت خروجی می‌رود.

۱۷.۵- گره اتصال MATCH



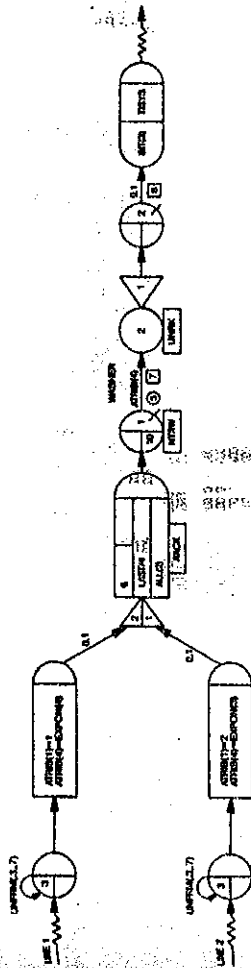
شکل ۱۷-۵ سمبل گرافیکی گره MATCH و دستور SLAM

گره دسته‌بندی موجودیهای مشابه و اتصال آنها (ارسال) به گره‌های بعدی است. در واقع موجودیهای ورودی به این گره که دارای مشخصه یکسانی هستند، توسط آن به گره مشخص تعریف شده ارسال می‌شوند.

توجه:

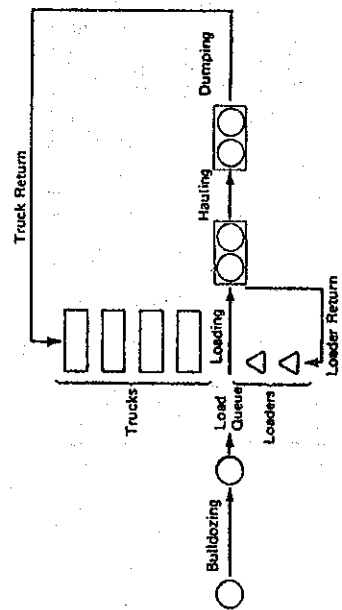
- ۱- دو طرف گره MATCH باید گره وجود داشته باشد.
- ۲- این گره عمل هدایت و ارسال موجودیها از گره به گره‌ای را صورت می‌دهد.
- ۳- طرف چپ گره MATCH حتماً گره‌های صف قرار دارد.
- ۴- متناظر با هر گره صف، گره‌ای در سمت راست گره MATCH وجود دارد.
- ۵- چنانچه برای صفی، گره متناظرش تعریف نشود، به هنگام هدایت موجودیهاش از بین می‌روند.
- ۶- می‌توان در خروجی گره MATCH مشترکی برای چند صف در نظر گرفت.

مدل SLAM سیستم:



میانگین ۱۲ است. زمان انتقال نرمال با میانگین ۲۲ و واریانس ۳ انجام می شود و زمان تخلیه دارای توزیع نرمال با میانگین ۱۸ و واریانس ۳ است. زمان بارگشت کامیونهای خالی یکپوخت بین ۲ تا ۸ است. لودر پس از هر بارگیری کامیونها به مدت ۵ واحد زمانی استراحت دارد. زمان شیب سازی ۴۸۰ واحد است.

مدل سیستم:



```

NETWORK:
: CREATE ENTITIES FROM LINE 1
:
: CREATE ENTITIES FROM LINE 2
:
: COMBINE 6 ENTITIES FROM THE SAME LINE
:
: RACK BATCH 2/1,6,,LAST/4,ALL(2),1;
: NTRW QUEUE(1),,10;
: ACT(5)/7,ATTRIB(4);
: UNRK UNBATCH,2,1;
:
: PROCESS INDIVIDUAL ENTITIES
:
: QUEUE(2);
: ACT/7,8,0,1;
: COLCT,INT(3),TISYS;
: TERMINATE;
: ENDNWORK;

```

برنامه سیستم: (مثال ۱۶-۱۷)

بخش پنجم / فصل هفدهم - گره های تصمیم گیری و منتقلی

تایخ:

SLAM II SUMMARY REPORT

SIMULATION PROJECT TRUCK HAULING BY PRITISKER
DATE 7/12/7983 RUN NUMBER 1 OF 1

CURRENT TIME 0.000000
STATISTICAL AVERAGES CLEARED AT TIME 0.000000

FILE STATISTICS**

| FILE NUMBER | ASST. NODE LABEL/TITLE | AVERAGE LENGTH | STANDARD DEVIATION | MAXIMUM LENGTH | CURRENT LENGTH | AVERAGE WAITING TIME |
|-------------|------------------------|----------------|--------------------|----------------|----------------|----------------------|
| 1 | LOAD QUEUE | 0.0000 | 0.0000 | 3 | 0 | 11.4022 |
| 2 | TRUCKS QUEUE | 1.1544 | 0.1722 | 2 | 2 | 12.1025 |
| 3 | LOADER QUEUE | 1.1544 | 0.1722 | 2 | 2 | 12.1025 |
| 4 | CUSTOMER | 4.3897 | 1.3419 | 8 | 0 | 8.7030 |

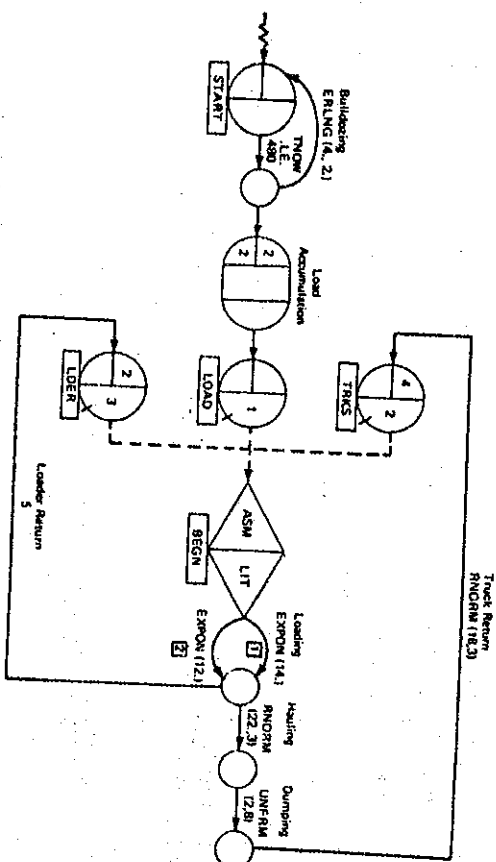
SERVICE ACTIVITY STATISTICS**

| ACTIVITY INDEX | START NODE ON ACTIVITY LABEL | SERVICE CAPACITY | AVERAGE UTILIZATION | STANDARD DEVIATION | CURRENT UTILIZATION | AVERAGE SERVICE TIME/SHOPS | MAXIMUM DLE TIME/SHOPS | MAXIMUM BEST DLE TIME/SHOPS | ENTRITY COUNT |
|----------------|------------------------------|------------------|---------------------|--------------------|---------------------|----------------------------|------------------------|-----------------------------|---------------|
| 1 | LOADQZ1 TIME | 1 | 0.3014 | 0.4589 | 0 | 0.0000 | 62.0798 | 35.8609 | 15 |
| 2 | LOADQZ2 TIME | 1 | 0.2803 | 0.4605 | 0 | 0.0000 | 62.0792 | 35.2041 | 15 |

مثال ۱۸-۱۷ تولیدکننده لباس با دریافت سفارشات پستی

یک تولیدکننده لباس که عمل تولید پیراهن و شلوار برای یک توزیع کننده مرکزی انجام می‌دهد. زمان تهیه و تولید یک دوجین پیراهن او دارای توزیع منطقی با یک مقدار مینیمم ۱۱ و ماکزیمم ۲۲ و مقدار محتمل ۱۶ است. زمان تولید یک دوجین شلوار نیز دارای توزیع منطقی با یک مقدار مینیمم ۱۱ و ماکزیمم ۲۲ و مقدار محتمل ۱۹ است. پس از تولید لباسها جهت بسته‌بندی ارسال می‌شوند که زمان انتقال آن ۷ دقیقه است. زمان بین ورود سفارشات دارای توزیع منطقی با میانگین ۳۰ می‌باشد. سفارشات ترکیبی از پیراهن و شلوار است. از سوابق سفارشات برمی‌آید که اندازه سفارشات براساس دوجینهای شلوار و پیراهن دارای توزیع تصادفی می‌باشد و احتمال آن که یک سفارش کامل شد با یک دوجین به ترتیب با احتمالاتی ۶/۱۵، ۵/۱۰، ۵/۰ است. وقتی یک سفارش کامل شد با یک گاری به محل بسته‌بندی سفارشات (انگراگ) ارسال می‌شود تا در صندوقچه قرار گیرد. حجم هر صندوقچه ۱۱۲ فوت مکعب است. پیراهنها در بسته‌های ۸ فوت مکعبی و شلوارها در بسته‌های ۱۲ فوت مکعبی بسته‌بندی می‌شوند. یک صندوقچه وقتی که ۱۰ فوت مکعب یا بیشتر پر شود برای بارگیری ارسال می‌گردد. در نیمه برای محاسبه قیمت محصول، ارزش دلاری مستوی و تعداد انتقال در هر هفته محاسبه می‌شود. ارزش دوجین پیراهن ۶۰ دلار و دوجین شلوار ۱۵۲ دلار می‌باشد. آمار براساس جمع‌آوری میانگین زمانی بین پُر کردن صندوقچه‌ها و ارزش دلاری کالا در یک صندوقچه است که طی پنج روز و روزی ۸ ساعت این آمار جمع‌آوری شود.

مدل SLAM سیستم:



برنامه:

```

1 GEN, PRITISKER, TRUCK HAULING, 7/12/83, 1, V/M;
2 LIMITS 3, 1, 50;
3 NETWORK;
4 START CREATE;
5 ACT, 1, TNOW, LE, 480;
6 GOON;
7 ACT, 1, ERLANG(4, 2), START;
8 ACT;
9 ACUM, 2, 2;
10 QUEUE(1), BEGN;
11 TRNS, QUEUE(2), BEGN;
12 LDR, QUEUE(3), BEGN;
13 BEGIN SELECT, ASM, LIT, LOAD, TRNS, LDR;
14 ACT/1, EXPON(14, 1);
15 ACT/2, EXPON(12, 3);
16 GOON;
17 ACT, 5, LDR;
18 ACT, RNORM(12, 3);
19 GOON;
20 ACT, UNIFORM(12, 8);
21 GOON;
22 ACT, RNORM(12, 3), TRNS;
23 FIN;

```

CREATE LOAD TRANSACTIONS
STOP IF AFTER 8 HOURS
ELSE
BRANCH BACK TO START
AND CONTINUE
ACCUMULATE TWO PILES
QUEUE OF LOADS
QUEUE OF TRUCKS
QUEUE OF LOADERS
ASM OF LOAD, TRNS, AND LDR
LOADER2 TIME
LOADER RESTING TIME
TRUCK HAULING TIME
TRUCK DUMPING TIME
TRUCK RETURN TIME

برنامه سیستم:

```

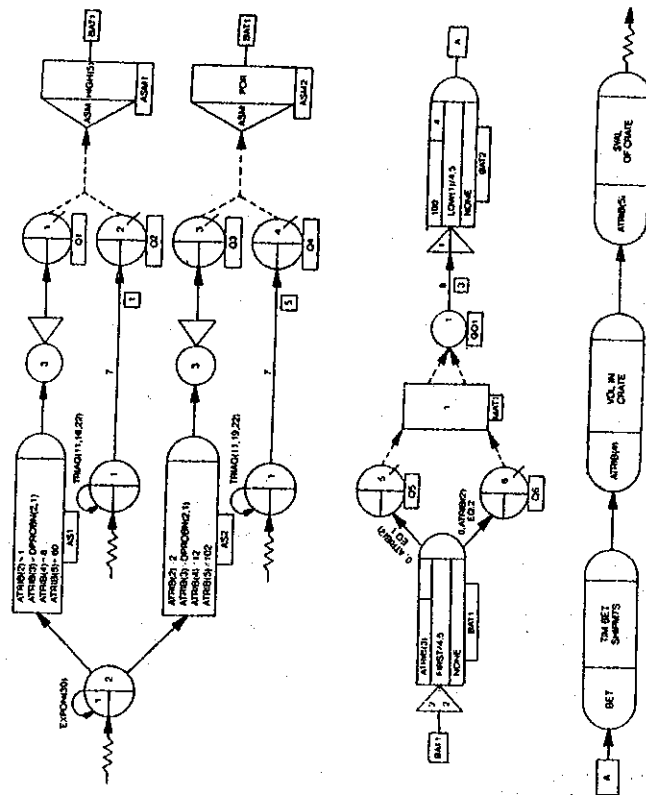
1 GEN VASEL CLOTHES MANUFACTURER, 10/14/85;
2 LIMITS 6,6,175;
3
4 DEFINE PROBABILISTIC ASSIGNMENT OF NUMBER OF DOZENS IN
5 EACH ORDER
6
7 ARRAY(1,4)/1, 2, 3, 4;
8 ARRAY(2,4)/.8, .8, .85, 1.0;
9 NETWORK;
10
11 CREATE ORDERS
12
13 CREATE EXPON(30), 1, 2;
14 ACT...AS1;
15 ACT...AS2;
16
17 AS1 ASSIGN ATRIB(2)=1, ATRIB(3)=EXPON(2,1), ATRIB(4)=8;
18 ATRIB(5)=60;
19 UNBATCH 3;
20 SPLIT INTO ENTITIES FOR 1 DOZEN
21
22 Q1 QLE(1)....ASM1;
23
24 MANUFACTURE SHIRTS
25
26 CREATE TRIAS(11...16...22)....1;
27 ACT/1,7;
28 MOV1 SHIRTS
29
30 Q2 QLE(2)....ASM1;
31
32 ASSEMBLY A DOZEN SHIRTS WITH AN ORDER
33
34 ASM1 SELECT ASM/IB(5)....Q1.Q2;
35 ACT...BAT1;
36
37 ORDERS FOR PANTS
38
39 AS2 ASSIGN ATRIB(2)=2, ATRIB(3)=EXPON(2,1), ATRIB(4)=12, ATRIB(5)=102;
40 UNBATCH 3;
41 SPLIT INTO ENTITIES FOR 1 DOZEN
42
43 Q3 QLE(3)....ASM2;
44
45 MANUFACTURE PANTS
46
47 CREATE TRIAS(11...16...22)....1;
48 ACT/5,7;
49 MOV1 PNTS
50
51 Q4 QLE(4)....ASM2;
52
53 ASSEMBLY A DOZEN PANTS WITH AN ORDER
54
55 ASM2 SELECT ASM...Q3.Q4;
56 ACT;
57
58 BATCH NO. OF DOZ. REQUIRED FOR ORDER
59
60 BAT1 BATCH/2,2, ATRIB(3), FIRST/4,5,1; BATCH SHIRTS AND PANTS SEPARATELY
61
62 ACT...ATRIB(2).ED.1.Q5;
63 ACT...ATRIB(2).ED.2.Q6;
64
65 Q5 QLE(5)....MAT1;
66 Q6 QLE(6)....MAT1;
67
68 MATCH THE SHIRT AND PANTS ENTITIES
69
70 MAT1 MATCH, 1,05/G01,05/G01;
71
72 G01 G00N,1;
73 ACT/3,8...BAT2;
74 TRANSPORT TIM
75
76 BATCH THE SHIRTS AND PANTS INTO A CRATE ENTITY
77
78 BAT2 BATCH, 100, 4, LOW(1)/4,5;
79
80 COLLECT STATISTICS ON TIME BETWEEN SHIPMENTS, THE VOLUME
81 IN EACH CRATE AND THE $ VALUE OF EACH CRATE
82
83 COLCT BET TIME BET SHIPMT;
84 COLCT ATRIB(4).VOL IN CRATE;
85 COLCT ATRIB(5).$VAL OF CRATE;
86
87 END;
88
89 INT,0,2400;
90
91 FIN;

```

شکل ۶-۱۷ مدل SLAM تولیدکننده لباس

در شکل مولدهایی عمل که تولید سفارشات و اندازه آنرا برعهده دارند نشان داده شده است. گره های ASM عمل متناظر را انجام می دهند (سفارش و اندازه آن) و به گره BAT1 برای دسته بندی وارد می شود (دو جت های شلوار و پیراهن در صف های Q4 و Q2 ذخیره شده اند). صفات موجودیت به شرح زیر تعیین شده است: ATRIB(2) تعیین کننده نوع پیراهن یا شلوار و ATRIB(3) زمان سفارش محصول ATRIB(4) تعداد دو جتین پیراهن یا شلوار در سفارش را نگه می دارد. ATRIB(5) شماره جعبه محصول و ATRIB(6) ارزش محصول را دربردارد. در گره BAT2 چنانچه حجم به ۱۰۰ فوت مکعب رسیده، از آن خارج می گردند و سپس آمارها جمع آوری می شوند.

مدل سیستم:



فصل هجدهم

ورودی ها و خروجی ها

فرمت عمومی دستورات

دستورات ISLAM از طریق یک ترمینال یا دستگاههای مشابه ورودی به پردازنده زبان داده می شوند. نکات زیر در دستورات به چشم می خورند:

- ۱- هر دستور به وسیله سه حرف اول نام آن دستور تعیین می گردد (دکتر ۳ حرف دستور کافی است).
 - ۲- فیلدهای دستورات به وسیله کاما [،] یا فست داخل پرانتز [()] جدا می شوند.
 - ۳- اسلش و پرانتز برای جدا سازی و تعریف بخش های اختیاری به کار می روند که انتخابی می باشند.
 - ۴- کاما برای جدا سازی فیلدهایی بکار می رود که انتخابی نیستند.
 - ۵- کاراکترهای خاص [/ * +] در عملیات ریاضی استفاده می شوند و در انتهای هر دستور سمی کالن [!] می آید.
 - ۶- کاراکترهای فوق نباید برای تعریف نام ها به کار رود (نامهای الفبا عددی یا برچسب ها).
- چنانچه در یک دستور ضرورتی بر ذکر مشخصات فیلد وجود ندارد باید علامت جدا کننده کاما ذکر گردد. مثلاً در یک دستور ACTIVITY بدون ذکر دوره زمان و شرط یا احتمال که فقط گره انتهایی آن نام EXIT دارد به صورت زیر است:

ACTIVITY " EXIT;

اگر تعدادی از فیلدهای انتهایی یک دستور تعریف و مشخص نیستند می توانند حذف شوند.

ACTIVITY ,;

مثال ۱-۱۸

یک دستور می تواند در یک یا چندین سطر نوشته شود ولی یک فیلد نمی تواند در چندین خط باشد. در صورتی یک دستور در چند سطر نوشته می شود که کاراکتر آخر هر سطر آن از علامت و پیرائورهای ریاضی [/ * +] یا پیرائورهای منطقی [AND] باشد و اگر غیر از موارد ذکر شده است انتهای دستور فرض می شود. در انتهای هر دستور علامت سمی کالن [!] ذکر می شود و پس از آن می تواند توضیحات بیاید. تمامی فواصل که در موارد فوق مجاز است در نظر گرفته نمی شود. برای

NETWORK, LOAD, 35

یعنی از دستگاه شماره ۳ دستورات شبکه خوانده شود. وقتی LOAD نوشته می شود، دیگر دستورات پس از NETWORK و دستور END نوشته نمی شوند. دستور END در انتهای تمامی دستورات شبکه می آید. لیست دستورات شبکه در جدول ۱۸-۱ آمده است.

منابع و مآخذ:

۱- سه حرف اول هر دستور کافی است.

۲- کاما [ا] برای تفکیک فیلدها استفاده می‌شود.

۳- اسلش برای تفکیک مقادیر انتخابی یا معمولی است.

۴- پراکنش برای تعیین ظرفیت فایل است.

۵- سعی کامل در اتمام هر دستور می آید.

[illegible]

جدول ۱-۱۸ انواع دستورات شبکه SLAM II

مثال دستور ACT:

ACTIVITY, 1, ATRIB (1). EQ. 1. AND. TNOW. LT. 100, LOOP;

که در سه خط به صورت می تواند نوشته شود:

ACTIVITY, 10,

ATTRIB(1).EQ.1.AND.

TNOW,LT-100,LOOP;

اول (یک سطری) است. داده‌های عددی می‌تواند به صورت عدد صحیح (integer) یا عدد کسری وارد شوند و می‌توانند با علامت یا بدون علامت باشند. ضمناً اعداد به صورت فرم نمایی E نیز می‌توانند وارد گردند مانند: 10 یا 10+ یا 1E+1 یا 100E-1. اگر یک عدد اعشاری برای یک فیلد عدد صحیح وارد شود بخش اعشاری آن حذف می‌گردد و اگر یک عدد صحیح جهت یک فیلد اعشاری صحیح وارد شود بخش اعشاری آن حذف می‌گردد. پس، فوقی در این اعداد 1 یا 1. قابل نیست.

فیلدهای آلفا عددی می‌توانند رشته یا کاراکتر باشد که با یک کاراکتر حرفی شروع می‌شود و مشتمل بر کاراکترهای خاص [0-9+]. کاراکتر فاصله در فیلد احتساب می‌گردد یعنی بین کاراکتر ۱ و ۲ تفاوت وجود دارد. حداکثر طول فیلد ۸۰ کاراکتر است ولی برحسب نوع فیلد ۳، ۴، ۶، ۱۶ کاراکتر، ۱۰۰ کاراکتر و غیره در مثال قابل قبول از فیلدهای فوق مانند زیر است:

GARBER, URANE, S927, K?L, iDOE

۱۸.۱- فرمت دستورات شبکه

دستورات شبکه می توانند از ستون ۷ یا بیشتر شروع شود و ستونهای ۱ تا ۵ را برای برچسب گره (label) در نظر گرفته اند. گره برچسب گره در هر کجای از ستونهای ۱ تا ۵ می تواند ظاهر شود ولی فقط ۴ تا از اکثر شروع احتساب می گردد. پس بین دو برچسب LANE1 و LANE2 تفا وتی قابل نیست دستورال شبکه با NETWORK آغاز و با ENDNETWORK پایان می یابد که می توانند از هر ستونی درج شوند. دستور NET دو فیلد اضافی دارد:

ETWORK, Option, Device;

در فیلد Option می‌تواند LOAD یا SAVE قرار گیرد که برای استفاده مجدد یا ذخیره آن در نظر گرفته شده است.

ETWORK,SAVE,35

مثال ۲-۱۸

مرتبط با انتخاب زیر است:

ILIST اگر YES، لیست دستورالعمل با پیام خطا چاپ می‌شود و در غیر آن خیر.
IECHO اگر YES، یک گزارش خلاصه چاپ می‌شود و اگر به حذف می‌گردد.
IXOT اگر YES، اگر خطایی باشد اجرا می‌شود و اگر NO اجرا نمی‌شود.
IWARN اگر YES، اگر موجودیتی قبل از ورود به گره TERMINATE از بین برود پیام یک هشدار خطا چاپ و در غیر آن حذف می‌شود.

IPRIRH اگر YES، نتایج میانی اجرای هر شبیه‌سازی با عنوان INTERMEDIATE RESULTS چاپ می‌شود.

ISMRY اگر YES، گزارش خلاصه مطابق تعریف فیلد بعدی (FSN) چاپ می‌شود، (F) اولین و آخرین و آخرین N، یعنی پس از هر N بار شبیه‌سازی چاپ شود).

IO: تعداد ستونهای گزارشات خروجی (۷۲ یا ۱۳۲ ترمینال و چاپگر) استفاده می‌شود.

۱۸.۲.۲ - دستور LIMITS

شکل دستور:

LIMITS MFILE, MATR, MNTRY;

دومین دستور برنامه شبیه‌سازی است. جهت تعیین محدودیت روی بیشترین تعداد فایل (MFILE)، بیشترین تعداد صفات هر موجودیت (MATR) و حداکثر تعداد مدخل‌های همزمان در تمامی فایلها (MNTRY) است. یعنی تعداد کل موجودیت‌هایی که در یک لحظه از مدل می‌تواند خارج شود، با این مقدار یک فاکتور اطمینان برای تخمین داریم.

۱۸.۲.۳ - دستور PRIORITY

PRIORITY / FILE, ranking / repeats;

شکل دستور:

برای تعیین معیار الریت (ranking) موجودیت‌های یک فایل (PFILE) است. که دارای چهار حالت می‌باشد:

FILO الریت خروج موجودیت از فایل به ترتیب ورود آنها می‌باشد. (اولین ورودی اولین خروجی است، پیش‌فرض).

FIFO الریت خروج موجودیت از فایل به ترتیب عکس ورود آنها می‌باشد. (آخرین ورودی اولین خروجی است).

۱۸.۲ - دستورات کنترل شبیه‌سازی

در این بخش دستورات اضافی ذکر می‌شود که در نوشتن برنامه شبیه‌سازی استفاده می‌گردد و عملیات کنترل برنامه شبیه‌سازی را برعهده دارد. این دستورات می‌توانند در هر ستونی قرار گیرند و عموماً در ستون یک نوشته می‌شوند. لیست کامل دستورات در جدول ۲-۱۸ آمده است که به ترتیب ذکر می‌گردد.

| Statement Form |
|--|
| ARRAY(ROW,NELEMENTS)/initial values;
CONTINUOUS,NNEQD,NNEQS,DTMIN,DTMAX,DTSAW,W or F or
N,NAERR,R,REERR;
ENTRY/IFL,ATRI(0),ATRI(2),...,ATRI(MATRI)/repeats;
EQU/VALENCE/SLAM II variable,name/repeats;
FIN;
GEN,NAME,PROJECT,MO/DAY/YEAR,/NNRNS,ILIST,IECHO,IXOT/IWARN,
IPRIRH,ISMRY/FSN,IO;
INITIALIZE, TTBEG,TTFIN,JICLR,NCCLR,JVAR,JIFIL;
INTLC,VAR = value,repeats;
LIMITS,MFILE,MATR,MNTRY;
MONTRO,option,TFEASTSEC,variables;
NETWORK,SAVE or LOAD,device;
PRIORITY/IFL,ranking/repeats;
RECORDPLOT/INDVAR,ID,TAPE,P or T or B,DIPLOT,TSRTTEND,KEYF;
SEEDS,ISEED(S)/R,repeats;
SEVENT,EVENT,XVAR,XDIR,VALUE,TOL;
SIMULATE;
STAT/CLCT,ID,NCEI/HLOW/HWID;
TIMST,VAR,ID;
VAR,DEPVAR,SYMBL,ID,LOAD,HORD; |

جدول ۲-۱۸ دستورات کنترل SLAM II

۱۸.۲.۱ - دستور GEN

این دستور اطلاعات عمومی درباره یک شبیه‌سازی به صورت زیر فراهم می‌آورد:

GEN,NAME,PROJECT,MONTH/DAY/YEAR,NNRNS,ILIST,IECHO,
IXOT/IWARN,IPRIRH,ISMRY/FSN,IO;

GEN اولین دستور در برنامه شبیه‌سازی است و شامل نام تحلیل‌گر، مشخصه یک پروژه، تاریخ، تعداد اجرای شبیه‌سازی و بخشهای انتخابی پروژه می‌باشد. فیلدهای NAME، IFL، تعدادی هستند با ۲۰ کاراکتر برای گزارش خروجی جهت تعیین تحلیل‌گر و پروژه است. MONTH (ماه)، DAY (روز) و YEAR (سال) اعداد صحیح هستند که با اسلش جدا می‌شوند. متغیر NNRNS که مقدار پیش فرض ۱ را دارد تعداد اجرای شبیه‌سازی را مشخص می‌کند. شش فیلد بعدی به صورت YES تعریف و

الویت خروجی موجودیت از فایل به ترتیب بیشترین مقدار صفت N موجودیت است.

الویت خروجی موجودیت از فایل به ترتیب کمترین مقدار صفت N موجودیت است.

اگر این معیار تعیین نشود پیش‌فرض آن FIFO است. پس شماره فایل در IFILE و معیار الویت در

ranking می‌باشد. می‌توان فایل تقویم پیش‌آمدها (events) را نیز تعیین الویت کرد که با ذکر NCLNR

بجای IFILE صورت می‌گیرد.

مثال ۱۸-۴

PRIORITYB, LIFONCLNR, HVF(4);

یعنی فایل ۳ به روش LIFO و فایل تقویم برای وقایع با بیشترین مقدار صفت چهارم موجودیت است.

مثال ۱۸-۵

PRIORITYB, LIFONCLNR, LVF(EVNTD);

الویت فایل ۳ براساس LIFO و جدول پیش‌آمدها براساس کمترین مقدار گد پیش‌آمد است.

۱۸.۲.۴ - دستور TIMST

شکل دستور:

برای جمع‌آوری اتوماتیک آمار از متغیرهای شناور XX(N) یا متغیرهای وضعیت یا مشتق SS(N) و

DD(N) است. جایی که N یک عدد صحیح و کاربر آن را در فایل متغیر XX(N) استفاده کرده و ID یک

الفا عددی برای نمایش آمارها در گزارش خلاصه SLAM II است. ۱۶ حرف اول کاراکتر الفا عددی

در گزارش خروجی می‌آید.

مثال ۱۸-۶

TIMST, XX(1), NUMBER IN SYSTEM;

جمع‌آوری آمار از متغیر XX(1) با عنوان NUMBER IN SYSTEM است. هیستوگرام می‌تواند روی

بخشی از زمان که متغیر VAR در یک محدوده مقادیر قرار داده باشد. تعداد سلولهای هیستوگرام

در NCEL است و HLOW و HHIGH قبلاً بیان شد.

مثال ۱۸-۷

TIMST, NNQ(1), QUEUE LENGTH, 10/0/1;

آمارگیری از موجودیتهای موجود در صف فایل ۱ با عنوان QUEUE LENGTH و ترسیم هیستوگرام در

زمانهای (۰.۰۰۱ تا ۱.۰۰۰) می‌باشد.

۱۸.۲.۵ - دستور SEEDS

شکل دستور:

هدف دستور آن است که به کاربر اجازه دهد برای ۱۰ جویبار اعداد تصادفی موجود در زبان SLAM II

SEEDS, ISBED(IS)/R, repeats;

عدد تصادفی غیر متعارف تعریف کند و مقداردهی جویبار را برای چندین اجرا شبه‌سازی کنترل کند.

ISEED به عنوان یک عدد صحیح است که با شماره جویبار (IS) در پرازنر وارد می‌شود. مقداردهی

مجدد هر جویبار به وسیله YES یا NO کنترل می‌شود (R). اگر IS تعریف نشود بطور اتوماتیک

شماره‌بندی برای آن تعیین می‌گردد.

مثال ۱۸-۸

SEEDS, 9375295(1)/YES, 6315779(9)/YES, 2734681;

برای اولین جویبار مقدار ۹۳۷۵۲۹۵ تعیین شده است که این مقدار به عنوان اولین مقدار هر اجرا

است. جویبار دوم از مقدار صفر شروع می‌گردد. جویبار نهم با مقدار ۶۳۱۵۷۷۹ در اجراهای بعدی

مقداردهی مجدد نمی‌شود و مقدار ۲۷۳۴۶۸۱ مربوط به جویبار دهم است زیرا جویبار قبلی نهم بود.

تابع XRN(IS) برای دسترسی آخرین مقدار تصادفی تولید جویبار IS می‌باشد.

۱۸.۲.۶ - دستور INTLC

شکل دستور:

INTLC, VAR=value, repeats;

برای مقداردهی متغیرهای XX(N), ARRAY(N,N), SS(N) یا DD(N) است جایی که N یک عدد صحیح

می‌باشد و هر مقداردهی با یک کاما جدا می‌شود.

مثال ۱۸-۹

INTLC, XX(1)=0, XX(2)=3.0;

۱۸.۲.۷ - دستور INITIALIZE

شکل دستور:

INITIALIZE, ITBEG, TTFIN, HCLNR/NCCLR, JIVAR, JIFIL;

برای تعیین زمان شروع (ITBEG) و زمان پایان (TTFIN) یک شبه‌سازی است و مقداردهی‌های

انتخابی برای پاک کردن آمارها، مقداردهی متغیرهای و مقدارگذاری فایلها می‌باشد. سه فیلد آخر با

YES یا NO تعیین می‌شوند. اگر YES برای فیلد JICLR باشد، آمارهای جمع‌آوری تا شماره CNCLR

پاک می‌شود. اگر NO JICLR تعیین می‌گردد آمارهای جمع‌آوری تا شماره NCCLR نباید پاک شوند.

پیش‌فرض NCCLR جمع‌آوری تمامی متغیرهای Collect است. اگر JIVAR YES تعیین شود مقدار

متغیر TNOW با زمان TTBEG تعریف و متغیرهای SS(N), DD(N) به مقادیر شروع قبلی از هر

شبه‌سازی مقداردهی گردد. اگر NO تعیین کنیم، از زمان شبه‌سازی اجرای قبلی تداوم می‌یابد. اگر

JIFIL به صورت YES تعریف شود، در هر اجرا محتوای فایلها از بین می‌رود و مقداردهی مجدد

می‌شود.

ARRAY(LJ) di JNNQ(0) JNRVSE(0) JNNRSC(0) JNNGAT(0) JNNCNT(0) JNNACT(0) JNNACT(0)

USERF(0) (تعریف نیاز دارد) باشد.

SYMBL مورد استفاده در نقشه برای تعیین متغیر وابسته است. اگر جدول نیاز باشد این را می توان پیش فرض در نظر گیریم (پیش فرض SYMBL (blank)).

ID برچسب الفبا عددی تا ۱۶ کاراکتر برای تعیین متغیر وابسته است. (پیش فرض blank) باشد

LOORD حد پائین که سمت چپ مقایسه نقشه ها است. یک مقدار می تواند حداقل مشاهده باشد

MIN است یا MIN به پائین می شود به نزدیکترین IVAL یعنی MIN(IVAL)

HLORD حد بالایی که سمت راست مقیاس نقشه ها است. یک مقدار می تواند حداکثر مشاهده باشد.

MAX یا MAX به بالا می شود به نزدیکترین IVAL یعنی MAX(IVAL)

تعریف محدوده ها برای رسم نقشه ها یا جداول می باشد. اگر کمترین مشاهدات ۵۴۷ و

MAX(۱۰۰۰) باشد، کمترین مقدار ۵۰۰ می شود. اگر بیشترین مقدار ۲۳۱۷ و MAX(۱۰۰۰) باشد،

بیشترین مقدار ۳۰۰۰ می گردد.

۱۸.۲.۱۲ - دستور SIMULATE

SIMULATE;

شکل دستور:

وقتی استفاده می گردد که اجرای مضاعف شبیه سازی ممکن باشد، در صورت اجرای یک شبیه سازی نیازی به آن نیست. به دنبال هر دستور SIMULATE کاربر می تواند تغییرات مورد نظر را اعمال کند مثلاً اعداد تصادفی جدید با دستور SEED دستورات جدید ENTRY یا مقادیر جدید برای متغیرهای XX(N) (با استفاده دستور (NNRNSL) با وضعیت های شروع یکسان است نیاز به دستور (NNRNS=1) یا چندین اجرا (NNRNS=1) نیست. اگر فقط یک اجرای شبیه سازی SIMULATE نیست، گر چه دستورات MONTR و ENTRY نیز فقط برای یک اجرا می باشند.

۱۸.۲.۱۳ - دستور FIN

FIN;

شکل دستور:

پایان تمامی دستورات SLAMMI را در بر دارد و دستورات را آماده اجرا می کند.

۱۸.۳ - برنامه راه اندازی مدلهای شبکه

دستورات در یک برنامه شبیه سازی غیر از دستورات شبکه به هر ترتیبی به صورت زیر می تواند

PortB شرح فرمت خروجی برای اطلاعات ثبت شده است. P چاپ نقشه، T چاپ جدول و B هر دو جدول و نقشه ای که باید چاپ شود (پیش فرض P).

DTPLT افزایش متغیرهای مستقل حین چاپ و پیشروی در نقشه. برای جداول رعایت نمی شود و تمامی مقادیر ثبت شده در جدول چاپ می گردند. (پیش فرض فاصله ۵/۰).

TTSRT زمانی که در آن شروع ثبت مقادیر مشخص شده است (شروع) (پیش فرض TTBEQ).

TTEND زمانی که در آن پایان ثبت مقادیر مشخص شده است (پایان) (پیش فرض TTFIN).

YES KKEVT یا NO بودن آن مشخص می کند که مقدار متغیرهای قبل یا بعد از زمان آمد ثبت شوند (پیش فرض YES).

برای هر دستور RECORD یک متغیر مستقل و یک سری متغیرهای وابسته وجود دارد. تعریف متغیرهای وابسته بلافاصله پس از RECORD در دستور VAR تعیین می شود. فرکانس ثبت داده برای یک نقشه به وسیله مقدار DTSV در یک دستور Continuous مشخص می گردد. کاربر می تواند به وسیله مقداردهی شماره نقشه، IPLOT در دستور RECORD مقداردهی مجدد کند. وقتی IPLOT تعیین شد، کاربر به وسیله صدا زدن سابروتین GPLOT(IPLT) مقادیر برای نقشه را ثبت می کند. دستورات RECORD بدون شماره نقشه باید به وسیله دستورات ورودی شماره نقشه را بدهد. وقتی متغیر ITAPE صفر شود، فضای آرایه NET/QSET برای ذخیره متغیر مستقل و هر متغیر وابسته است. تعداد مقادیری که ذخیره می شود تعداد متغیرهای وابسته و آنهایی که ثبت می شوند به علاوه یک است. وقتی دستور Continuous استفاده نشود فرکانس ثبت متغیرها توسط متغیر DTPLT تعیین می گردد (DTSV). در این وضعیت، DTSV با بزرگترین مقدار DTPLT مقدار می گیرد.

اگر KKEVT=YES است، قبل و بعد از هر پیش آمد شبیه سازی متغیرها ثبت می شوند. وقتی که ITAPE بزرگتر از صفر است شماره دستگاه چایی باید برای ذخیره مقادیر تعیین گردد. این شماره از طریق کارتهای کنترل زبان یا کارتهای مورد نیاز برنامه مشخص می شود. اگر فضای کافی نتراند به OSET تخصیص داده شود پیام کافی نبودن حافظه اعلان می گردد.

۱۸.۲.۱۱ - دستور ورودی VAR

VAR, DEPVAR, SYMBL, ID, LOORD, HIORD;

شکل دستور:

مرتبط با دستور RECORD می باشد که برای تعریف متغیرهای وابسته ای است که باید به ازای هر مقدار متغیر مستقل ثبت شود.

DEPVAR نام متغیر وابسته است. متغیر وابسته می تواند: JTNOW, SS(0), DD(0), XX(0)

جدول ۱۸-۲ تعریف خروجی‌های مرتبط با شکل ۱۸-۱

General Section

- (A1) The first 20 characters of the project title
- (A2) The first 20 characters of the analyst name
- (A3) The MONTH/DAY/YEAR
- (A4) The number of the simulation run
- (A5) The number of simulation runs to be made
- (A6) The current value of TNOW
- (A7) Time at which all statistical arrays were last cleared

Statistics for Variables Based on Observation

- (B1) The first 16 characters of the statistics label
- (B2) The arithmetic mean of the observations
- (B3) The standard deviation of the observations
- (B4) The coefficient of variation (standard deviation/mean)
- (B5) The minimum value over all observations
- (B6) The maximum value over all observations
- (B7) The number of observations

Statistics for Time-Persistent Variables

- (C1) The label for the variable
- (C2) The average value of the variable over time
- (C3) The standard deviation over time
- (C4) The minimum value of the variable over time
- (C5) The maximum value of the variable over time
- (C6) The time interval over which the statistics are accumulated
- (C7) The current value of the variable

وارد شود.

- ۱- اولین دستور GEN, LIMITS دومین دستور و FIN آخرین دستور است.
- ۲- دستورات شبکه بلافاصله بعد از NET و قبل از END می آید.
- ۳- دستور INITIALIZE قبل از دستورات ENTRY و MONTR می آید.
- ۴- دستور MONTR بعد از دستور ENDNET می آید.

۱۸.۴- گزارشات خروجی SLAM II

خلاصه گزارشات خروجی در شکل ۱۸.۱ آمده است.

SLAM II SUMMARY REPORT

PROJECT NUMBER (1) DATE (2)

CURRENT TIME (3) RUN NUMBER (4) OR (5)

STATISTICS ARE BEING CLEARED AT TIME (6)

STATISTICS FOR VARIABLES BASED ON OBSERVATION

| MEAN VALUE (7) | STANDARD DEVIATION (8) | COEFFICIENT OF VARIATION (9) | MINIMUM VALUE (10) | MAXIMUM VALUE (11) | NUMBER OF OBSERVATIONS (12) |
|----------------|------------------------|------------------------------|--------------------|--------------------|-----------------------------|
| (13) | (14) | (15) | (16) | (17) | (18) |

STATISTICS FOR TIME-PERSISTENT VARIABLES

| MEAN VALUE (19) | STANDARD DEVIATION (20) | MINIMUM VALUE (21) | MAXIMUM VALUE (22) | TIME INTERVAL (23) | CURRENT VALUE (24) |
|-----------------|-------------------------|--------------------|--------------------|--------------------|--------------------|
| (25) | (26) | (27) | (28) | (29) | (30) |

FILE STATISTICS

| FILE NUMBER (31) | ASSOC. MODE (32) | AVERAGE LENGTH (33) | STANDARD DEVIATION (34) | MAXIMUM LENGTH (35) | CURRENT LENGTH (36) | AVERAGE WAITING TIME (37) |
|------------------|------------------|---------------------|-------------------------|---------------------|---------------------|---------------------------|
| (38) | (39) | (40) | (41) | (42) | (43) | (44) |

REGULAR ACTIVITY STATISTICS

| ACTIVITY NUMBER (45) | AVERAGE UTILIZATION (46) | STANDARD DEVIATION (47) | MINIMUM UTILIZATION (48) | MAXIMUM UTILIZATION (49) | ENTRY COUNT (50) |
|----------------------|--------------------------|-------------------------|--------------------------|--------------------------|------------------|
| (51) | (52) | (53) | (54) | (55) | (56) |

SERVICE ACTIVITY STATISTICS

| ACTIVITY NUMBER (57) | START MODE OR ACTIVITY CODE (58) | SEIZE (59) | AVERAGE UTILIZATION (60) | STANDARD DEVIATION (61) | CURRENT UTILIZATION (62) | MAXIMUM UTILIZATION (63) | MAXIMUM USE TIME (64) | MAXIMUM WAITING TIME (65) | ENTRY COUNT (66) |
|----------------------|----------------------------------|------------|--------------------------|-------------------------|--------------------------|--------------------------|-----------------------|---------------------------|------------------|
| (67) | (68) | (69) | (70) | (71) | (72) | (73) | (74) | (75) | (76) |

RESOURCE STATISTICS

| RESOURCE NUMBER (77) | RESOURCE LABEL (78) | CURRENT CAPACITY (79) | AVERAGE UTILIZATION (80) | STANDARD DEVIATION (81) | MAXIMUM UTILIZATION (82) | CURRENT UTILIZATION (83) | MAXIMUM AVAILABLE (84) | MINIMUM AVAILABLE (85) |
|----------------------|---------------------|-----------------------|--------------------------|-------------------------|--------------------------|--------------------------|------------------------|------------------------|
| (86) | (87) | (88) | (89) | (90) | (91) | (92) | (93) | (94) |

DATE STATISTICS

| DATE NUMBER (95) | DATE (96) | CURRENT STATUS (97) | PCI OF TIME OPER (98) |
|------------------|-----------|---------------------|-----------------------|
| (99) | (100) | (101) | (102) |

شکل ۱۸-۱ خروجی‌های آماری جدول ۱۸-۲

- (F4) The average number of entities in service over time. If the capacity of the server is 1, this corresponds to the fraction of time the server is busy.
- (F5) The standard deviation of the number of entities in service over time.
- (F6) The current number of entities in service.
- (F7) The average number of servers blocked over time. If the capacity of the server is 1, this corresponds to the fraction of time blocked.
- (F8) If the capacity of the server is 1, this value specifies the maximum idle time of the server. If the capacity of the server is greater than 1, this value specifies the maximum number of idle servers.
- (F9) Maximum busy time for a single server or maximum number of busy servers.
- (F10) If the service activity is assigned an activity index number, the number of entities completing service; otherwise no value is printed.

Resource Utilization Statistics

- (G1) The resource number assigned by the processor
- (G2) The resource label as specified on the RESOURCE block
- (G3) The current capacity of the resource
- (G4) The average utilization of the resource over time
- (G5) The standard deviation of the resource utilization over time
- (G6) The maximum number of units of resource utilized at any one time
- (G7) The current number of units of resource in use

Resource Availability Statistics

- (H1) The resource number assigned by the processor
- (H2) The resource label as specified on the RESOURCE block

File Statistics

- (D1) The file number or event calendar
- (D2) The node type (if any) associated with the file
- (D3) The average number of entities in the file over time
- (D4) The standard deviation of the number of entities in the file over time
- (D5) The maximum number of entities in the file at any one time
- (D6) The current number of entities in the file
- (D7) The average waiting time of all entities that arrived to the file including those that did not wait

Regular Activity Statistics

- (E1) The activity index number for the activity
- (E2) The average number of entities in the activity
- (E3) The standard deviation of the number of entities in the activity
- (E4) The maximum number of entities in the activity at any one time
- (E5) The number of entities currently in the activity
- (E6) The number of entities which have completed the activity

Service Activity Statistics

- (F1) The activity number for the activity. A zero if a number is not assigned. Statistics are listed in the same order as the input statements.
- (F2) The label and type (QUEUE or SELECT) of the start node of the activity.
- (F3) The number of parallel identical servers represented by the activity.

فصل نوزدهم

مدلسازی شبکه باروتین تقریفی کاربر

در این فصل شیوه نوشتن روان پروسیجر برای مدلسازی با زبان فرتون توسط کاربر بیان می‌شود. با شرح گروه‌های EVENT و ENTER شروع می‌کنیم که کلید واسط بین مدل شبکه و گد کاربر را فراهم می‌کند. سپس پشتیبانی زیر برنامه‌های SIAM II برای پیاده‌سازی قابل، مقادیر صفات و ارائه دوره فعالیتها و مانند آن ارائه می‌شود.

۱۹.۱- گروه EVENT



شکل ۱۹-۱ سمبل گرافیکی گروه پیش آمد EVENT

این گروه برای اتصال مدل شبکه به گد واقع نوشته شده به وسیله مدلساز است. گروه EVENT سبب می‌شود که هر وقت یک موجودیت به آن گروه وارد شود، سابروتین (EVENT(JEVT)) را صدا بزند. مقدار JEVT گد پیش آمدی که باید اجرا شود تا تعیین می‌کند و M ماکزیمم تعداد فعالیتهایی است که باید بعد از پردازش گروه بیاید. نام EVENT در شروع یا بعد از ستون ۷ می‌آید و گد روان (پروسیجر) به زبان فرتون نوشته می‌شود.

شکل عمومی سابروتین EVENT

```
SUBROUTINE EVENT(Q)
GO TO (1,2),I
1 Code for event node 1
RETURN
2 Code for event node 2
RETURN
END
```

گد پیش آمد است و همان JEVT می‌باشد.

دستور GOTO گد پیش آمد ۱ را تعیین و دستور العمل مربوطه را اجرا می‌کند.

- (H3) The current number of units of resource available
- (H4) The average availability of the resource over time
- (H5) The minimum number of units of resource available at any one time
- (H6) The maximum number of units of resource available at any one time

Gate Statistics

- (I1) The gate number assigned by the processor
- (I2) The gate label as specified on the GATE block
- (I3) The current status of the gate
- (I4) The percentage of time the gate was opened

PREEMPT و AWAIT, QUEUE دارد که وضعیت آنرا جهت انتخاب از فایل نسبت به اعضای دیگر فایل تعیین می‌کند. این روشها عبارت از SLAM FIFO, LIFO, HVF, LVF می‌باشد که قبلاً شرح داده شد. در زبان SLAM II برای تشخیص فایلها از شماره‌ای که کاربر می‌دهد استفاده می‌شود که بطور اتوماتیک اطلاعات آماری هر فایل را جمع‌آوری می‌کند. مثلاً تابع NNQ(IFILE) تعداد موجودیهای فایل IFILE را می‌دهد (NNQ(2) فایل شماره ۲ را بررسی می‌کند). مدلساز از طریق زیربرنامه‌های موجود می‌تواند دسترسی داشته باشد.

۱۹.۴.۱ - سابروتین FILEM(IFILE,A)

این سابروتین صفات مشخص شده در بافر آرایه A را داخل فایل IFILE ثبت می‌کند که این امر براساس الویت موردنظر مشخص شده در فایل ذخیره می‌گردد. مثال: ذخیره یک ورودی با صف اول آن که زمان شبیه‌سازی است در فایل 1:

A(1)=TNOW

CALL FILEM(1,A)

پردازنده زبان SLAM نمی‌تواند مستقیم این ورودی را در فایل ذخیره کند و پس از بازگشت از سابروتین، به عنوان یک پیش‌آمد ورودی به فایل مربوطه در سیستم اضافه می‌گردد تا پردازش شود. اگر سروس‌دهنده بیکار باشد مستقیم به آنجا ارسال می‌شود. اگر صف پر باشد BALK می‌شود و اگر نه برای سروس گرفتن در فایل منتظر می‌ماند. زیربرنامه (سابروتین) FILEM(IFILE,A) در SLAM II امکان درج مستقیم در فایل را فراهم می‌کند.

۱۹.۴.۲ - سابروتین REMOVE(NRANK,IFILE,A)

برای برداشتن یک موجودیت از فایل IFILE براساس الویت NRANK و قرار دادن در بافر آرایه A است. مثال: برداشتن آخرین موجودیت فایل ۱ و قرار دادن در آرایه A

CALL REMOVE (1,1,A)

برداری که کاربر به عنوان آرایه A استفاده می‌کند باید بزرگتر یا مساوی حداکثر صفات موجودیت MATR به علاوه ۲ باشد. برداشتن دومین ورودی از فایل ۳ و قرار دادن در آرایه ATRIB.

مثال ۱۹-۲

CALL REMOVE (2,3, ATRIB)

مثال ۱۹-۱

```
SUBROUTINE EVENT(I)
GO TO (1,2),I
1 CALL LUNCH
RETURN
2 CALL ELUNCH
RETURN
END
```

اگر گد ۱ بیاید سابروتین LUNCH و اگر گد ۲ بیاید سابروتین ELUNCH صدا زده می‌شود.

۱۹.۲ - دسترسی به مقادیر شبکه

از طریق توابع زیر می‌توان در گره EVENT به مقادیر شبکه دسترسی داشت.

NNACT(I): تعداد موجودیهای فعال فعالیت I در زمان فعلی است.

NNCNT(I): تعداد موجودیهایی که در فعالیت I کامل شده است.

NNGAT(I): وضعیت گیت I در زمان فعلی است: I:close, O:open.

NRUSE(I): تعداد منابع در حال استفاده I است.

NNRSC(I): تعداد منابع موجود I است.

NNBL K(I,FILE): تعداد موجودیهای بلو که شده فعالیت I به وسیله فایل IFILE می‌باشد.

۱۹.۳ - گره ENTER



شکل ۱۹-۲ سمبل گرافیکی گره ENTER

برای دادن یک موجودیت دلخواه در شبکه است و از طریق زیر برنامه نوشته شده کاربر انجام می‌گیرد.

۱۹.۴ - دستکاری فایلها

یک فایل مکانیزمی برای ذخیره‌سازی صفات یک موجودیت با توجه به دیگر موجودیهای داخل آن فایل را فراهم می‌کند. در مدل‌های شبکه، فایلها برای نگهداری موجودیهای منتظر در گره‌های

بردار ST برای ۶ خانه و با مقادیر Data معلوم شده است. XNINO تعداد موجودیهای قابل ۱ می گیرد. میانگین زمان سروس در تابع GTABL تعریف شده است.

مثال ۴-۱۹ عملیات بانکی

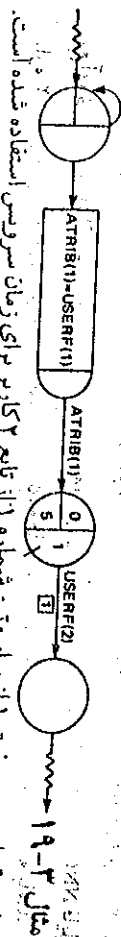
در یک بانک دو پنجره و هر کدام یک پاسخگو یا خط پاسخگونی جدا دارد. با مشاهدات مشخص شده است که زمان بین ورود مشتریان ناشانی با میانگین ۵/۰ واحد زمانی است. ازدحام فقط در آنای ساعتهایی رخ می دهد و باید آنالیز شود. زمان سروس پاسخگویان دارای توزیع نرمال با میانگین یک واحد زمانی و انحراف معیار ۳/۰ واحد زمانی است. مشتری به خط یک می رود اگر سروس دهنده یکبار یا دو خط مساوی باشند. در تمامی موارد دیگر مشتری کوتهترین خط را انتخاب می کند. به علت فضای محدود فقط ۳ نفر در هر خط توقف می کنند و این بهلاوه هر نفری است که در حال سروس دیدن می باشد (پس کلاً ۸ جا). مشتری در صورت پُر بودن صف به مکان دیگری هدایت می شود.

شرایط اولیه:

- ۱- هر دوسروس دهنده پاسخگو مشغول هستند و زمان سروس هر گرفته دارای توزیع نرمال با میانگین ۱ و انحراف معیار ۳/۰ است.
 - ۲- اولین مشتری در واحد زمانی ۱/۰ وارد می شود.
 - ۳- در مشتری در هر صف منتظر هستند.
 - هدف مدلسازی این است که بتوان با استفاده از نتایج شبیه سازی مسائل زیر را آنالیز کرد:
 - ۱- کارایی پاسخگویان = $\frac{\text{کل زمان سروس}}{\text{زمان شبیه سازی}}$
 - ۲- میانگین زمانی تعداد مشتریان سیستم.
 - ۳- زمان بین خروج از پنجره های پاسخگویی.
 - ۴- میانگین زمانی یک مشتری در سیستم.
 - ۵- میانگین تعداد مشتریان در هر صف.
 - ۶- درصد مشتریان ورودی به Bank
- سیستم برای ۱۰۰۰ واحد زمانی شبیه سازی می شود.

۵-۱۹ تابع کاربر USERF

برای آنکه کاربر بتواند با گند زبان FORTRAN از طریق یک شبکه کار کند. SLAM II تابع USERF(1FN) را ایجاد کرده و آرگومان 1FN به عنوان شماره تابع کاربر می باشد.



مثال ۳-۱۹ برای مقداردهی صف ۱ از سابروتین شماره ۱ از تابع ۲ کاربر برای زمان سروس استفاده شده است.

FUNCTION USERF(1FN)
*GO TO (1,2),1FN
1 Set USERF as the time to travel to QUEUE node
RETURN
2 Set USERF as service time to be a function of number in queue
RETURN
END

شکل عمومی نوشتن تابع:

چگونه این عمل می کنند؟ برای مثال وقتی که تابع ۱ نیاز باشد USER(1) به 5 مقداردهی می گردد. وقتی که از سابروتین برمی گردد این مقدار را (در مثال فوق) داخل ATTRIB(1) قرار می دهد. برای تابع ۲، باید تعداد موجودیهای صف را بدانیم و سپس تابع زمانی زمان سروس براساس نورات صف کم می شود که به صورت جدول شکل زیر در صف عمل می کنیم:

| نورات در صف | میانگین زمان سروس |
|-------------|-------------------|
| 0 | 10. |
| 1 | 9. |
| 2 | 8. |
| 3 | 7. |
| 4 | 9. |
| 5 | 10. |

گند برای تابع زیر مشخص است:

```

FUNCTION USERF(1FN)
  DIMENSION ST(6)
  DATA ST/0.,9.,8.,7.,9.,10./
  GO TO (1,2),1FN
  1 USERF = 5 - NNQ(1)
  RETURN
  2 XNINO = NNQ(1)
  AVEST = GTABL(ST,XNINO,0.0,5.0,1.0)
  USERF = EXPON(AVEST,1)
  RETURN
END
  
```

در این سابلروتین NL1 و NL2 جهت محاسبه تعداد موجودی خط ۱ (LEFT) و خط ۲ (RIGHT) است. اگر تعداد نفرات صف ۲ بیشتر از نفرات صف ۱ به علاوه ۲ است لذا آخرین موجودیت در فایل ۲ (NNQ(2)) از صف RIGHT QUEUE به وسیله صدا کردن REMOVE (با فایل ۲) صدا و در فایل ۱ (LEFT QUEUE) قرار داده می شود به وسیله صدا کردن FILEM (با FILE=1) انجام می گیرد. جمع آوری آمارها در گره های COLCT انجام می شود. نتایج با عنوان TIME IN SYSTEM و TIME و BET DEPART آمده است. این دستورات در شکل ۱۹-۵ نشان داده شده است.

```

1 GEN OREILLY DRIVE IN BANK, 4/12/82, 1;
2 LIMITS 2, 1, 75;
3 TINST, USERF(1), NO. OF CUST;
4 SEEDS, 4367651(1), 6121137(2);
5 NETWORK;
6 CREATE CUSTOMERS
7 PLACE CUSTOMER IN SHORTEST Q
8 TELLER 1
9 CHECK FOR JOCKEYING INTO LEFT QUEUE
10 EVENT, 1;
11 ACT, EXIT;
12 RIGHT QUEUE(2), 2, 3;
13 SELECT, SNQ, BALK(NBALK), LEFT, RIGHT;
14 COLCT, INT(1), TIME IN SYSTEM;
15 EXIT
16 COLCT, BET, TIME BET. DEPART;
17 TERM;
18 NBALK COLCT, BET, TIME BET. BALKS;
19 TERM;
20 END;
21 INIT, 0, 1000;
22 FIN;

```

شکل ۱۹-۵ دستورات ورودی مدل بانک

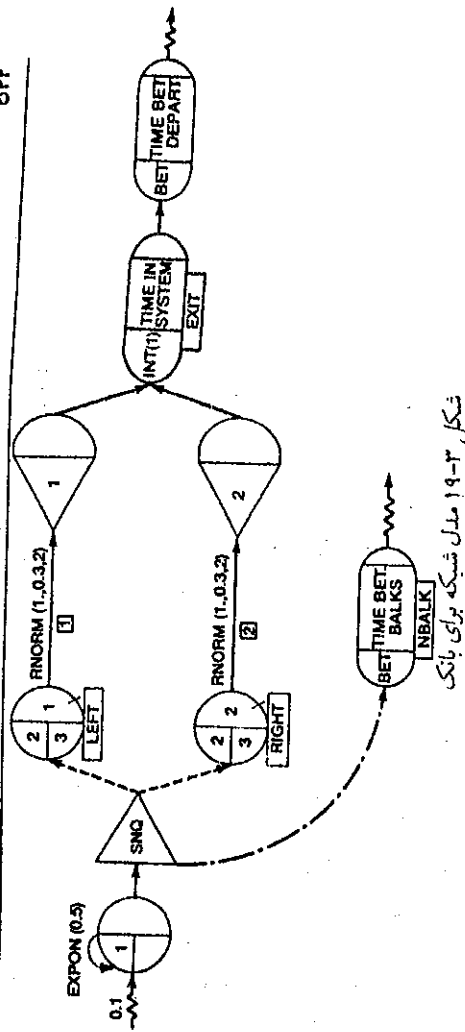
دستورات کنترلی GEN, LIMITS, TINST, SEEDS, INIT در این مدل می باشد. دستورات مشتریانی که به شکل استاندارد است. دستور TINST بدست آوردن آمار تعداد مشتریانی که به این جهت به صورت تابع تعریف شده (USERF(1) آمده است. در تابع تعریفی USERF(1) شکل ۱۹-۴، جمع آوری تعداد نفرات صف ۱ (NNQ(1)+NNACT(1)) و نفرات صف ۲ (NNQ(2)+NNACT(2)) صورت می گیرد.

```

FUNCTION USERF(1)
C*****
C CALCULATES THE TOTAL NUMBER OF CUSTOMERS IN THE SYSTEM
C*****
USERF=NNQ(1)+NNACT(1)+NNQ(2)+NNACT(2)
RETURN
END

```

شکل ۱۹-۶ تابع USERF برای مدل بانک



شکل ۱۹-۳ مدل شبکه برای بانک

این مثال از گد نوشته شده کاربرد استفاده می کند که عبارتند از (۱) استفاده از گره EVENT (۲) یک فایل عمومی موارد ارجاع بین شبکه و مدل وقایع گسته. ضمناً از گره SELECT نیز استفاده می شود. شوح مدل: از گره EVEN برای حرکت بین صفها استفاده می شود. گره Create زمان ورود موجودیهای را در مشخصه اول آنها ذخیره می کند. گره Select برای مسیریابی به صفهای چپ و راست (Right & Left) و براساس کوچکترین شماره در قانون صف (SNQ) است. صف LEFT دارای دو مشتری است و حداکثر ۳ مشتری اجازه می دهد و صف RIGHT نیز همین است. پس از فعالیت سروس به گره های EVENT با شماره ۱ و ۲ هدایت می شوند. در گره SLAM II EVENT در سابلروتین (EVENT) برای شماره های ۱ و ۲ وقایع پیش آمده را ایجاد کرده است.

```

SUBROUTINE EVENT(1)
DIMENSION A(10)
NL1=NNQ(1)+NNACT(1)
NL2=NNQ(2)+NNACT(2)
GO TO (1,2), 1
C***** IF THE NUMBER IN LANE 2 EXCEEDS LANE 1 BY 2
C***** THEN JOCKEY FROM 2 TO 1
1 IF (NL2.LT.NL1+2) RETURN
CALL REMOVE(NNQ(2), 2, A)
CALL FILEM(1, A)
RETURN
C***** IF THE NUMBER IN LANE 1 EXCEEDS LANE 2 BY 2
2 IF (NL1.LT.NL2+2) RETURN
CALL REMOVE(NNQ(1), 1, A)
CALL FILEM(2, A)
RETURN
END

```

شکل ۱۹-۴ سابلروتین EVENT برای مدل بانک

دو واحد از منابع شماره ۱ و برای مثال فوق یعنی TUG آزاد می‌کند و مشابه دستور زیر است.

۱۹.۶.۲- جانشینی ظرفیت منبع (سابرواقین) (ALTER)

سایبروتین ALTER(GRN) ظرفیت منابع IR را N واحد تغییر می دهد. مثال: تقابل منبع ۱، TUGS واحد:

CALL ALTER(1,-1)

ALTER, TUG-1;

مثال ۱۹-۵

۱۹.۷- پروسیجر تخصیص منبع نوشته کاربر
(سابروین های ALLOC و SEIZE)

گروه‌های AWAIT برای ذخیره موجودیهای منظر منابع استفاده می‌شود. تا زمانی که تعدادی از منابع موجود نباشد یک موجودیت نمی‌تواند از گره AWAIT عبور کند. وقتی که این حالت رخداد به تعداد مقضی از آن نوع ("SEIZED") ریزه می‌شود و موجودیت از گره AWAIT عبور می‌کند.

AWAIT(1), ALLOC(1);

ACT/1,10;

این دستورات نشان می‌دهد که یک موجودیت به گره AWAIT وارد می‌شود و از میان فایل 1 به ACTIVITY 1 می‌رود اگر طبق قانون 1 تخصیص منابع موجود نباشد. کد برای تخصیص قانون 1 به شرح زیر است:

```

IFLAG = 0
GO TO(1,2),1

C****
C****
C****
C****
1  IF (NNRSC(1),LE.0.OR.,NNRSC(3),LE.0) RETURN
    CALL SEIZE (1,1)
    CALL SEIZE(3,1)
    IFLAG=-1
    RETURN

```

ابتدا IFLAG صفر قرار داده‌ایم تا هیچ تخصیصی صورت نگیرد. پس برای حالت $1 = 1$ یک پرش به

دستور شماره ۱ داریم که منبع ۱ (OP1) و منبع ۳ (OP3) موجود نباشند تخصیصی صورت نمی گیرد. NRSQC تعداد منابع موجود از منبع ۱ را نشان می دهد. سپس منبع ۱ و ۳ درباره مقداردهی می شوند.

| STATISTICS FOR COEFFS BASED ON DEVIATIONS | | | | | |
|---|------------|-----------------------|------------------------|------------------|------------------|
| | MEAN | STANDARD
DEVIATION | COEFF. OF
VARIATION | MINIMUM
VALUE | MAXIMUM
VALUE |
| TRE IN COSTA | 0.2275-01 | 0.10165-01 | 0.44683-02 | 0.23595-01 | 0.53995-01 |
| TRE BEL. DEPART | 0.33582-01 | 0.16702-01 | 0.49755-01 | 0.19155-01 | 0.53995-01 |
| TRE BEL. BALS | 0.08822-01 | 0.12115-02 | 0.13645-01 | 0.11155-01 | 0.30045-01 |
| | | | | | 0.53995-01 |

| STATISTICS FOR THE PERSISTENT VARIATES | | | | |
|--|---------------|-----------------------|------------------|------------------|
| NO. OF CLUST | MEAN
VALUE | STANDARD
DEVIATION | MINIMUM
VALUE | MAXIMUM
VALUE |
| 0.4372E+01 | 0.2194E+01 | 0.4003E+02 | 0.3003E+01 | 0.1003E+04 |
| | | | | 0.7002E+04 |

| FILE
NUMBER | ASSOC
LABEL/TYPE | AVERAGE
LENGTH | STANDARD
DEVIATION | MAXIMUM
LENGTH | CURRENT
LENGTH | AVERAGE
WRITING TIME |
|----------------|---------------------|-------------------|-----------------------|-------------------|-------------------|-------------------------|
| 1 | LEFT CHAIR | 1.3105 | 1.0027 | 3 | 3 | 1.1079 |
| 2 | RIGHT CHAIR | 1.1826 | 0.8828 | 3 | 3 | 1.2313 |
| 3 | CALENDAR | 2.4534 | 0.4523 | 3 | 3 | 3.3995 |

| **SERVICE ACTIVITY STATISTICS** | | | | | | | | | |
|---------------------------------|------------------------------|------------------|---------------------|--------------------|---------------------|-----------------|--------------------------|--------------------------|--------------|
| ACTIVITY INDEX | START NODE OR ACTIVITY LABEL | SERVICE CAPACITY | AVERAGE UTILIZATION | STANDARD DEVIATION | CURRENT UTILIZATION | AVERAGE RESERVE | MAXIMUM FILE TIME/SERIES | MINIMUM BUSY TIME/SERIES | DEFECT COUNT |
| 1 | TOTAL 1 | 1 | 0.5394 | 0.2365 | 1 | 0.0007 | 2.0264 | 64.9627 | 942 |
| 2 | TOTAL 2 | 1 | 0.3140 | 0.2364 | 1 | 0.0003 | 4.1834 | 71.4719 | 912 |

فصل ۱۹-۷ خلاصه گزارش بانک
فصل ۱۹-۷ خلاصه گزارش بانک

۱۹۶- تغییر وضعیت یک منبع

می‌توان واحدهایی از یک منبع را آزاد یا ظرفیت آنرا عوض کرد. یک منبع از طریق نام یا شناسه‌اش (RESOURCE (TTCG2) 1 2 (PPTV) 2 3 شناسخته می‌شود. مثال:

RESOURCE / TUG(2), 1, 2 / BERTH(3), 3,
RESOURCE / CREW, 4;

پر دازنده زبان، TUG را به عنوان منبع ۱، BERTH منبع شماره ۲ و CREN را منبع شماره ۳ می شناسد.

۱۹.۶.۱- آزاد سازی منابع (سابروین (FREE

این سابلوگین $FREE(IR, N)$ به تعداد N واحد از منبع شماره IR را آزاد می‌سازد و آزاد شده‌ها برای
CALL FREE (12)

موجودیتهای منتظر بزرگوار دانه می شود. دستور:

دارند. بلاکهای منابع برای تعیین این سه اپراتور به صورت زیر است:

```
RESOURCE / OP(1), 1, 3;
RESOURCE / OP(2), 2, 4;
RESOURCE / OP(3), 1, 3;
```

از این تعریف می‌بینیم که یک واحد از هر منبع OP1 و OP3 از فایل‌های ۱ و ۳ تخصیص داده شده است. منبع OP2 در فایل ۲ و ۴ تخصیص یافته و منبع OP2 در اینجا نمی‌تواند استفاده شود. مقادیر عددی به وسیله SLAM II به منبع OP1 یک و منبع OP3 سه داده می‌شود.

در مدل یک شبکه، فرض کنیم که فعالیت ۱ به وسیله هر دو OP1 و OP3 نیاز به انجام داشته باشد. به علت آنکه این یک ترکیب پیچیده از تخصیص دو منبع است، سابروتین ALLOC باید استفاده شود. دستورات شبکه برای گره Awaiting و ACTIVITY به صورت زیر است:

```
SUBROUTINE ALLOC(FLAG)
COMMON/SCOM/ATTRIB(100),DDI(100),DTNOW,ILMFA,MSTOP,NCLNR
I,NCRDR,NPRINT,NNRUN,NNET,NTAPE,SSI(100),SSLI(100),TNEXT,TNOW,XXI(100)
IFLAG=0
GO TO (1,2),I
```

```
C
C ALLOCATION RULE 1 - SEIZE RESOURCES 1 AND 3
C
C 1 IF(NNRSC(1).LE.0.OR.NNRSC(3).LE.0) RETURN
C CALL SEIZE(1,1)
C CALL SEIZE(3,1)
C IFLAG = -1
C RETURN
```

```
C
C ALLOCATION RULE 2 - SEIZE RESOURCE 1 OR 3
C
C 2 SAVE SELECTED RESOURCE IN ATTRIB(I)
C
C 2 IF (NNRSC(1).LE.0.AND.NNRSC(3).LE.0) RETURN
C IF (NNRSC(1).GT.0) THEN
```

```
CALL SEIZE (1,1)
ATTRIB(I)=1
IFLAG=1
RETURN
ELSE
```

```
CALL SEIZE(3,1)
ATTRIB(I)=3
IFLAG=1
RETURN
```

```
ENDIF
CALL ERROR (3)
RETURN
END
```

مانند بحث قبل $IFLAG=0$ شده است. سپس چک می‌شود که اگر منبع ۱ موجود باشد، زیر برنامه SEIZE صدا زده می‌شود تا یک واحد از منبع شماره ۱ را برآید. $ATTRIB(1)=1$ می‌شود یعنی موجودیت فعلی به وسیله منبع ۱ پردازش شود. $IFLAG=1$ معین می‌کند که اولین موجودیت فایل ۳ باید از فایل خارج شود و IFLAG مثبت یعنی بافر صفات باید تغییر کند. قبل از صدا زدن سابروتین ALLOC در SLAM II بافر ATTRIB از اولین موجودیت فایل Awaiting روی گردان می‌شود و بقیه نیز به

$IFLAG=1$ یعنی اولین موجودیت در فایل ۱ باید از گره Awaiting جریان باید و منتهی باعث می‌شود که صفات آن تغییر نکند.

شرح یک تابع تخصیص منبع به وسیله دستورات زیر:

```
AWAIT(3/2),ALLOC(2),BALK(OUT);
ACT/5,22;
```

در گره Awaiting می‌گوید که وجوهای ورودی در فایل ۳ که یک فایل با گنجایش ۲ دارد منتظر شود و در صورت رسیدن به ۲ موجودی به گره OUT هدایت می‌شود. وقتی که یک موجودیت به فایل ۳ رسید و این فایل خالی بود سابروتین ALLOC قاضای قانون ۲ تخصیص را می‌دهد. قانون ۲ تخصیص که گد خواهد شد منبع ۱ یا ۳ را به یک موجودیت می‌دهد.

سابروتین ALLOC شامل قوانین ۱ و ۲ است که در زیر نشان داده شده است. قانون ۲ با $I=2$ بدست می‌آید. بجای نامگذاری یک نوع منبع، مدلساز می‌تواند در فایل ALLOC(1) آنرا مشخص کند. در II SLAM زمانی که یک موجودیت وارد شد سابروتین ALLOC(1,IFLAG) صدا زده می‌شود که منبع به صورت ALLOC(I) مشخص شده است. در گد کردن سابروتین ALLOC تمامی قواعد سابروتینهای FORTRAN و SLAM II استفاده می‌شود در حالی که هیچ عملی روی فایل پیش آمد اجازه داده نمی‌شود و سابروتینهای ALTER و FREE در آن نمی‌تواند صدا زده شود. اگر ما بخواهیم ALLOC :

۱- گد کاربر، ۱ برای جداسازی صدا زدن سابروتین ALLOC از گره‌های مختلف مختلف Awaiting است. ۲- فلگ IFLAG به وسیله کاربر تعیین می‌شود به SLAM II بگوید کدام تخصیص انجام می‌گیرد. اگر فلگ صفر شود هیچ تخصیصی صورت نمی‌گیرد.

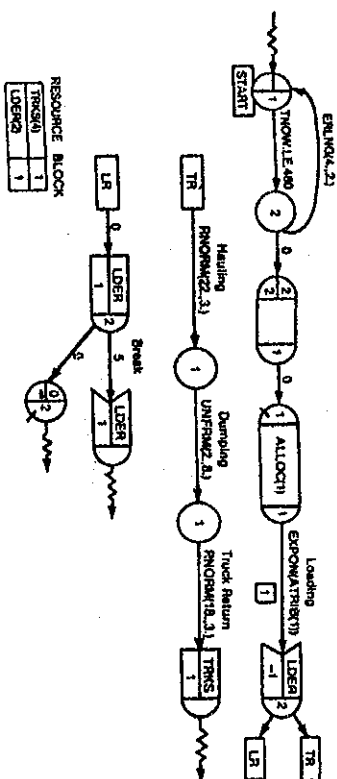
مقدار مثبت یا منفی در IFLAG می‌آید. یک فلگ مثبت معین می‌کند که بافر ATTRIB در ALLOC مقادیر صفاتی که با خروج موجودیت از فایل مشخص نگردد، تعیین شود. فلگ منفی یعنی موجودیت خارج شود و تغییر صورت نگیرد و اگر تخصیص ممکن نیست فلگ صفر می‌شود.

وقتی که منابع در گره Awaiting باید تخصیص داده شوند، به وسیله کاربر در سابروتین ALLOC تعیین می‌گردد و تعداد واحدهای مورد نیاز منابع در انتهای صدا زدن سابروتین SEIZE(IR,NU) تغییر اندازه می‌دهد. که IR عددی منبع و NU تعداد واحدهایی که باید تغییر اندازه بدهد مانند CALL SEIZE(3,2)

مثال ۱۹-۶ ترکیب و جایگذاری منابع مورد نیاز

شرح یک وضعیت به اپراتور مدل شده به عنوان منابع دارند. منابع نامهای OP1، OP2 و OP3

شده است. کامیونها و لوردها به عنوان منابع با برچسبهای TRKS و LDER یا شماره های ۱ و ۲ مشخص گردیده است. در سابر ویتن ALLOC یک تست برای تعیین وجود هر دو TRKS و LDER انجام می شود. اگر موجود باشند تخصیص می یابد و یک بارگیری تداوم می یابد. سابر ویتن ALLOC در زیر آمده است.



```

SUBROUTINE ALLOC(I,FLAG)
  DIMENSION AD
  COMMON/SCOUT/ATTRIB(100),DOL(100),DTNOW,II,MFA,MSTOP,NCLNR
  1,NCRB,NPNT,NNRUN,NMSET,NTAPE,SS(100),SS(100),TNEXT,TNOW,XX(100)
  C*** SET FLAG TO ZERO TO INDICATE NO ALLOCATION
  FLAG=0
  C*** RETURN IF NO TRUCK OR NO LOADER
  IF (NNRSC(1).EQ.0.OR.NNRSC(2).EQ.0) RETURN
  C*** SEIZE TRUCK AND LOADER
  CALL SEIZE(1,1)
  CALL SEIZE(2,1)
  C*** OBTAIN LOADER ATTRIBUTES FROM QUEUE 2
  CALL RMVQ(1,2,A)
  C*** MODIFY ATTRIB BUFFER FOR FIRST ATTRIBUTE ONLY
  ATTRIB(1)=A(1)
  FLAG=1
  RETURN
END
  
```

۱۹.۸ - تغییر وضعیت یک GATE

SLAM II توانایی باز یا بسته بودن یا استفاده از دستورات FORTRAN را فراهم می کند.

مثال ۹-۱۹

GATE / DOOR1, OPEN, 4;
GATE / DOOR2, CLOSED, 5;

بر درازنده زبان DOOR1 به عنوان دروازه شماره ۱ و DOOR2 به عنوان دروازه شماره ۲ است.

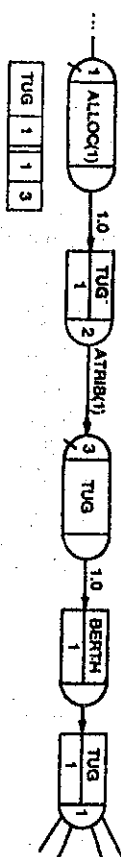
۱۹.۸.۱ - بازگردان یک GATE (سابر ویتن OPEN)

سابر ویتن OPEN(IG) یک دروازه با شماره IG را باز می کند و تمامی موجودیهای مستقر در

همین منوال سپری می گردد.

مثال ۷-۱۹ مروری بر عملیات پورت

قبلاً شرح عملیات روی پورتها داده شد. این امر در SLAM II می تواند با استفاده از ترکیب روالها پیرو سخرها در مثال ۱۹.۱ ارائه شود. مدل شبکه گره Awaiting با قانون ۱ به شرح زیر است:



فرض می کنیم که اگر TUG در پورت باشد و تانکر آماده ترک محل است؛ لذا یک TUG باید تخصیص داده شود. مدل این روال باید آخرین عملیات TUG را نگه دارد، $XX(1)=1$ اگر یک عملیات پهلوترن صورت گیرد و $XX(1)=0$ اگر TUG یکبار باشد یا یک عملیات خروج انجام دهد. وقتی TUG آزاد می شود، موجودیت در فایل ۱ برای تخصیص مشخص می گردد و قانون ۱ تخصیص برای این امر است. اگر TUG برای پهلوترن تخصیص نیافته $XX(1)$ صفر می شود و return صورت می گیرد. رویتن به شرح زیر است:

```

SUBROUTINE ALLOC(I,FLAG)
  COMMON/SCOUT/ATTRIB(100),DOL(100),DTNOW,II,MFA,MSTOP,NCLNR
  1,NCRB,NPNT,NNRUN,NMSET,NTAPE,SS(100),SS(100),TNEXT,TNOW,XX(100)
  FLAG=0
  C*** CALLED TO CHECK FOR ALLOCATION OF TUG AND BERTH FOR BERTHING
  C
  C ALLOCATE IF A) TUG WAS LAST DEBERTHING, XX(1) = 0, OR
  C B) TUG WAS BERTHING BUT NO SHIPS ARE READY TO DEBERTH
  C***
  IF (XX(1).EQ.0) GO TO 10
  IF (NMDC(3).EQ.0) GO TO 10
  RETURN
  C***
  C BERTHING WILL OCCUR IF BOTH TUG AND BERTH ARE AVAILABLE
  C***
  10 IF (NMDC(1).LE.0.OR.NMDC(2).LE.0) THEN
    XX(1) = 0.
    RETURN
  ENDIF
  CALL SEIZE(1,1)
  CALL SEIZE(2,1)
  FLAG = 1
  XX(1) = 1
  RETURN
END
  
```

مثال ۸-۱۹ ترکیب منابع در وضعیت ایستگاه خردکن

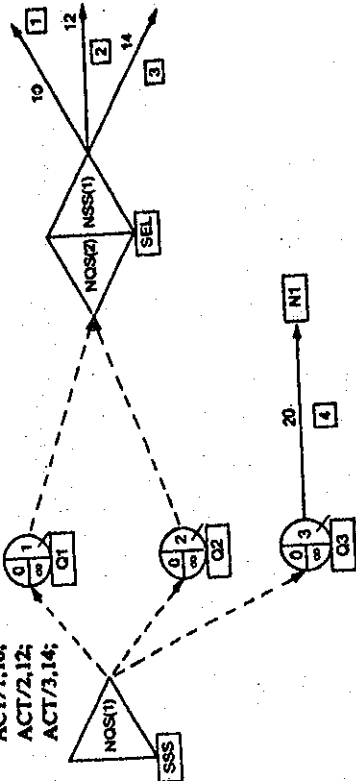
در مثال خردکن شرح عملیات با مدل SLAM II بیان شد. در مدل زیر یک گره Awaiting استفاده

NETWORK:

SSS SELECT,NQS(1),...Q1,Q2,Q3;
 Q1 QUEUE(1),0,...SEL;
 Q2 QUEUE(2),0,...SEL;
 Q3 QUEUE(3),0;
 ACT/4,20,N1;
 SEL SELECT,NQS(2),NSS(1),Q1,Q2;
 ACT/1,10;
 ACT/2,12;
 ACT/3,14;

N1 NOT SHOWN

مدل سابروتین NQS به شرح زیر است:



FUNCTION NQS(N)
 COMMON/SCOM1/ATTRIB(100),DDL(100),DDL(100),DTNOW,ILMFA,MSTOP,NCLNR
 L,NBERR,INPT,NBERR,NBSET,NTAPE,SSL(100),TNEXT,TNOW,XX(100)

GO TO (12,2,N
 1 NQS = XX(1)
 RETURN

2 NOS = 0

3 **** TEST FEASIBILITY - IS AN ENTITY WAITING FOR SERVICE?
 IF (NNQ(1).EQ.0.AND. NNQ(2).EQ.0) RETURN

IF (NNQ(1).EQ.0) THEN

NQS = 2

RETURN

ELSE

IF (NNQ(2).EQ.0) THEN

NQS = 1

RETURN

ENDIF

ENDIF

C**** SELECT QUEUE FOR WHICH SUM OF ATTRIB(0) VALUES

C**** FOR ALL ENTITIES IS THE LARGEST

IF (SUM(Q3,1).GE.SUM(Q3,2)) THEN

NQS = 1

ELSE

NQS = 2

RETURN

END

۱۹.۱۰ - انتخاب سرویس دهنده تگارش کاربر (تابع NSS)

در SLAM II زمانی که بیش از یک فعالیت سرویس بعد از یک صف یا مشابه دارد از گروه SELECT استفاده می گردد. گد نوشته شده کاربر به وسیله NSS(N) در بخش سرویس دهنده می تواند انجام گردد.

قابلیت های WAIT مشخص شده در بلاک GATE را آزاد می کند.

CALL OPEN(2)

OPEN, DOOR2;

مثال ۱۰-۱۹ باز کردن DOOR2

این دستور معادل است با:

۱۹.۸.۲ - بستن یک Gate (سابروتین CLOSX)

سابروتین CLOSX(IG) دروازه شماره IG را می بندد.

CALL CLOSX(1)

CLOSE, DOOR1;

مثال ۱۱-۱۹

این دستور معادل است با دستور:

۱۹.۹ - انتخاب صف تگارش کاربر (تابع NQS)

در مدل شبکه از گروه های SELECT برای مسیریابی یک موجودیت به یکی از سری موازی گروه های QUEUE یا انتخاب یک موجودیت از سری صف های موازی برای پردازش توسط یک سرویس دهنده است. با استفاده NQS(N) به عنوان قانون انتخاب گروه SELECT تابع نوشته شده کاربر NQS با آرگومان N جهت انتخاب صف از گروه SELECT است. شکل زیر چگونگی استفاده از تابع NQS را نشان می دهد.

در گروه (SELECT)SSS، موجودیتهای ورودی به صف های Q1، Q2، Q3 براساس مقدار XX(1) مسیریابی می شوند. در بخش دیگر نشان داده نشده مدل، مدلساز XX(1) را به ۱ یا ۲ یا ۳ مقداردهی می کند. تابع ۱ برای این انتخاب است و NOS(1) قانون انتخاب صف برای گروه انتخاب SSS می باشد. در تابع NQS(N) یک پرش به دستور ۱ (وقتی انتخاب ۱ است) صورت می گیرد. در دستور ۱، مقدار NQS برابر تعداد XX(1) قرار می گیرد که شماره فایل ورودی موجودیت است. در گروه انتخاب SEL تابع ۲ برای گزینش موجودیت از Q1 یا Q2 برای فعالیتهای ۱ و ۲ یا ۳ است و این عمل در دستور ۲ صورت می گیرد.

۱۹.۱۱- توقف یک فعالیت (ساو و تین STOPA)

در سیستم های پیچیده طول زمان فعالیت ممکن است ناشناخته و به یک دینامیک سیستم وابسته باشد. مثلاً در یک صف سرعت سروس ممکن است به نرخ ورودی بستگی داشته باشد. یک راه آن استفاده از STOPA(NTC) است این اجازه می دهد که مدل ساز فعالیت یک موجودیت خاص را متوقف کند. برای توقف یک فعالیت برای یک موجودیت شبکه از گد نوشته شده کاربر، مدل ساز باید دوره فعالیت را به صورت STOPA(NTC) مشخص کند. NTC عدد صحیح مثبت است و با آن می تواند گد های مختلفی را اجرا کند.

ACT, STOPA(1);

مثال ۱۹-۱۳

یعنی برای تمامی موجودی های فعالیت، گد ۱ در نظر گرفته شده است.

ACT, STOPA(ATRIB(3));

مثال ۱۹-۱۴

در این مثال نوع گد در مشخصه سوم صفات موجودیت وجود دارد.

مکانیزم این امر در گد ساو و تین STOPA(NTC) می آید که مثلاً جهت اجرا به این صورت صدا می شود:

مثال ۱۹-۱۵ مرکز روان پزشکی

استفاده کنندگان از یک مرکز روانی با نرخ دو نفر در روز وارد می شوند. هر مراجعه کننده یک آزمایش دارد که نمونه آن توزیع یکپارچه بین ۳۰ تا ۴۴ دارد. وقتی که مرکز پُر است مراجعه کننده اگر نمونه او بیش از ۴۱ باشد حذف نمی شود. مرکز گنجایش ۲۵ بیمار دارد. نمونه آزمایش بیمار در روز با تقریب یکپارچه در محدوده ۲/۰- تا ۲/۱ تغییر می کند. وقتی که نمونه آزمایش به ۴۸ رسید بیمار از مرکز خارج می شود. در ورود که مرکز پُر باشد، یک بیمار که آزمایش آن بیش از ۴۷ باشد از مرکز خارج می گردد.

در حالت اولیه ۱۸ بیمار با نمونه آزمایش بین ۳۰ تا ۴۴ وجود دارد. شبیه سازی برای ۱۰۰۰۰ روز و برای میانگین زمان هر بیمار در سیستم، کارآیی مرکز، تعداد استفاده کنندگان همدانی؛ جای دیگر و تعداد بیمارانی از خارج شده است.

تابع NSS را وقتی که یک موجودیت وارد صف خالی شود به وسیله گره SELECT صدا می زند. داخل تابع NSS شاخص برای انتخاب انشعاب منتهی در گره SELECT وجود دارد.

مثال ۱۹-۱۲

```
SEL SELECT,NSS(2),NSS(0),Q1,Q2;
ACT/1,10;
ACT/2,12;
ACT/3,14;
```

وقتی یک موجودیت به گره Q1 یا Q2 یک هیچ موجودیت در زمان رسیدن آن منتظر نیست می رسد، تابع NSS با یک آرگومان ۱ به منظور تعیین سروس دهنده ۱، ۲، برای پردازش صدا زده می شود. اگر تمامی این سه سروس دهنده مشغول باشند، موجودیت در فایلی مقتضی قرار می گیرد. اگر فقط یک سروس دهنده آزاد باشد آنرا برای سروس انتخاب می کند و اگر بیش از یکی آزاد باشد اولین گد شده NSS این را انتخاب می کند که گد آن به شرح زیر است:

```
FUNCTION NSS(N)
DIMENSION NSS(N)
GO TO (1),N
1 NSS=0
2 SET NUMBER AVAILABLE TO 0
3 DO 10 I=1,N
4 IF (INACTD,N) GO TO 10
5 SET FIRST AVAILABLE SERVER NUMBER TO NSS(N)
6 J=1+1
7 NSS(N)=1
8 CONTINUE
9 IF NO SERVER AVAILABLE, RETURN WITH NSS=0
10 IF (J,EQ,0) RETURN
11 XI=J+1
12 RANDOMLY SET A SERVER RANK BETWEEN 1 AND J
13 NR=UNIFORM(XI,1)
14 GET SERVER NUMBER FROM SERVER RANK VECTOR
15 NSS=NSER(NR)
16 RETURN
END
```

فر دستور ۱۰ NSS=0 می گردد چون اگر قبل از مرتب شدن NSS یک return باشد، سروس دهنده نتواند انتخاب شود. در بخش وسطی تابع وضعیت هر فعالیت سروس تست می شود و اگر سروس دهنده موجود باشد، شماره فعالیت سروس در بردار NSS(N) که ۱ شاخص سروس دهنده موجود است ذخیره می گردد. اگر هیچ کدام موجود نباشند 0=J می شود و NSS=0 باز می گردد.

اگر یکی یا بیشتر از یک سروس دهنده موجود باشند یک عدد صحیح از ۱ تا تعداد سروس دهنده ها انتخاب می شود. $XI=J+1$ می شود و یک عدد بین ۱ و XI انتخاب می گردد که در NR قرار می گیرد. سپس شماره سروس دهنده انتخاب شده NSS از NR امین محل پردازش NSS انتخاب می گردد.

مذیل میں:

THE USERF EQUAL TO EMPTY BED NUMBER

...XX(J).GT.O.) GO TO 10

US EAF-3

GO TO 20
CONTINUE

to continue

U C : F AN EMP

CALL ERROR(1)

NO RETURN

END

۱۰۰

بسم الله الرحمن الرحيم

الحمد لله رب العالمين

ازماینده

ممانند واقعیت

۵۰

اصدا کے دن مبارک۔۔۔

32

۱۰۰۰ گز

55

جی سی شود

5

EVENTS

1

۱۵۷ زیر اثر افشان و

5

شکل ۸-۱۹ مدل شبکه مرکز روانپزشکی

این مثال از گروه EVENT، سابروتین ENTER، گروه ENTER تعریف دوره STOPA و سابروتین STOPA استفاده می‌کند. همچنین تابع USERF و سابروتین INTLC نیز ارائه می‌گردد.

3

عمل آزمایش ونمره آن در SLAM II به وسیله ارائه بیماران در مرکز روانی به عنوان موجودیتهای
ACTIVITY با دوره مشخص شده به صورت STOPA(II) است. جایی که II فضای موجود یا تعداد
تختخوابهای مرکز را مشخص می کند. نمره آزمایش بیماران در ۲۵ فضای مرکز در متغیر XX(II)
اگر که فضای II اشغال نشود، $XX(II)=0$ می شود.

نگهداری می‌شود اگر که فضای II انتقال نشود، $XX(II)=0$ می‌شود.
زمان ورود بیماران که به وسیله گره Create ایجاد می‌شوند در ATRIB(1) آنها ذخیره می‌گردد و
نمره آزمایش آنها در ATRIB(2) موجود است. M عدد برای گره ASSIGN به عنوان 1 تعیین شده است.

سابروتین BUMP زیر شامل منطق پردازش برای گره (BUMP)EVENT است و فقط وقتی صدا می شود که بیمار جدید می رسد. متغیر SCORE به ۳۷ و متغیر II به 0 مقداردهی می شود. حلقه DO برای نمونه بیشتر از ۳۷ جستجو می گردد اگر یافت شد دستور ۱۰، ۱۲، ATRIB(2) را به ۳۸ مقدار می دهد. اگر بیماری خارج شود، یک موجودیت به گره ENTER (۳) وارد، با 0 XX(J)=0 بیمار خارج شده و سابروتین STOPA(J) صدا می شود. پس یک سابروتین return از سابروتین BUMP ساخته می شود. سابروتین سابروتین ENTER(2, ATRIB) اجرا می شود که سبب می شود یک موجودیت به گره ENTER در رأس جدول زمانی بندی آن وارد شود:

```
SUBROUTINE BUMP
COMMON/SCAM/ ATRIB(100), DO(100), DOL(100), DTOW, 11, MFA, MSTOP, NCLIN
1, NCONB, NPRINT, NNNUN, NNET, NTAPE, SS(100), SSL(100), TTEXT, TROW, XX(100)
SCORE=47,
J=0
C *** CHECK TO SEE IF ANY PATIENTS IN THE WARD HAVE SCORES
C *** GREATER THAN 47, IF SO DETERMINE WHICH PATIENT HAS
C *** THE HIGHEST SCORE,
DO 10 J=1,26
IF(XX(J).LE. SCORE) GO TO 10
SCORE=XX(J)
J=J+1
10 CONTINUE
IF(J.EQ.0) GO TO 20
C *** ROUTE THE NEW PATIENT THROUGH THE ENTER NODE AND INTO THE
C *** WARD. CALL STOPA FOR THE BUMPED PATIENT.
CALL ENTER(2, ATRIB)
XX(J)=0
CALL STOPA(J)
RETURN
C *** A PATIENT ELIGIBLE FOR BUMPING WAS NOT FOUND. PLACE
C *** THE NEW PATIENT INTO THE NETWORK AND MAKE HIM BALK.
20 ATRIB(2)=46.
RETURN
END
```

گد سابروتین TEST که روزانه هر نمونه آزمایش بیمار در سابروتین زیر وارد می شود. پردازش در حلقه DO برای آزمایش انجام می شود که تغییر آنرا انجام می دهد. اگر نمونه ای بیشتر یا مساوی ۳۸ باشد، یک بیمار را خارج می سازد یعنی جایی که لا اشاره می کند:

```
1 GEN OREILLY, PSYCHIATRIC WARD, 12/20/83, 1;
2 LIMITS, 0, 2, 150;
3 NETWORK;
4 ;
5 CREATE, 1, 1; DAILY RETEST OF OCCUPIED BEDS
6 TEST EVENT, 2;
7 ;
8 ;
9 CREATE, EXPON(5), 1;
10 ADMT ASSIGN ATRIB(2)-UNFPM(30, 44), 1; PATIENT ARRIVALS
11 ACT, INACT(1), GE, 25, AND; ASSIGN PATIENT SCORE
12 ATRIB(2), GT, 41, BALK; BED NOT AVAILABLE
13 ACT, INACT(1), GE, 25, BUMP; TRY TO BUMP
14 ACT, WARD; BED AVAILABLE
15 BALK COLCT, BET, TIME, BET, BALKS; COLLECT BALK STATISTICS
16 TERM;
17 ;
18 ; IS A PATIENT REPARABLE?
19 BUMP EVENT, 1; BUMPED PATIENT STATS
20 ACT, 0, ATRIB(2), GE, 48, BALK;
21 ACT;
22 TERM;
23 ;
24 ; ASSIGN PATIENT A BED
25 ENTER, 2;
26 WARD ASSIGN, II=USERF(1), XX(11)-ATRB(2);
27 ACT/1, STOPA(1);
28 COLCT, INT(1), TIME IN WARD; WAIT FOR CALL TO STOPA
29 TERM;
30 ENDMETWORK;
31 INITIALIZE, 0, 500;
32 FIN;
```

دو پیش آمد خارج مدل شبکه گد شده است. اولین پیش آمد به وسیله ورود یک موجودیت به گره (BUMP) EVENT مقداردهی می شود و در سابروتین BUMP گد شده است. دومین پروسه پیش آمد روزانه در نمونه آزمون بیمار تغییر می کند و در صورت بیمار را خارج می سازد. منطق برای این پیش آمد در سابروتین TEST گد شده است. سابروتین زیر (EVENT) گد پیش آمده ها را در سابروتین مقتضی نشان می دهد:

```
SUBROUTINE EVENT(1)
GO TO (1,2), 1
C *** EVENT CODE 1---SUBROUTINE BUMP
1 CALL BUMP
RETURN
C *** EVENT CODE 2---SUBROUTINE TEST
2 CALL TEST
RETURN
END
```

SLAM II SUMMARY REPORT

STIMULON PROJECT PSYCHIATRIC WARD BY DRELLY
DATE 12/29/1983 RUN NUMBER 1 OF 1

CURRENT TIME 0.000E+00
STATISTICS WARD'S CLEARED AT TIME 0.000E+00

STATISTICS FOR VARIABLES BASED ON OBSERVATIONS

| | MEAN
VALUE | STANDARD
DEVIATION | Coeff. of
VARIATION | MINIMUM
VALUE | MAXIMUM
VALUE | NUMBER OF
OBSERVATIONS |
|---------------------------------|------------------------|------------------------|------------------------|------------------------|------------------------|---------------------------|
| TIME BET. PAUSE
TIME IN WARD | 0.800E+00
0.220E+02 | 0.120E+01
0.400E+01 | 0.140E+01
0.360E+00 | 0.600E+00
0.510E+01 | 0.100E+02
0.450E+02 | 500
500 |

REGULAR ACTIVITY STATISTICS

| ACTIVITY
INDEX/LABEL | STANDARD
UTILIZATION | MAXIMUM
UTIL | CURRENT
UTIL | BRUTY
COUNT |
|-------------------------|-------------------------|-----------------|-----------------|----------------|
| 1 IN WARD | 24.000 | 0.500 | 25 | 25 |

SUBROUTINE TEST
COMMON/SCOM1/ ATRIB(100),DDL(100),DDL(100),DTNOW,11,MFA,MSTOP,NCLNR
1,NCRDR,NPRINT,NNRUN,NSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)

C *** DAILY UPDATE PATIENT SCORES AND SEE IF THE PATIENT IS
C *** WELL ENOUGH (SCORE AT LEAST EQUAL TO 48) TO LEAVE.
C *** IF PATIENT IS WELL ENOUGH TO LEAVE CALL STOPA WITH
C *** THE PATIENT'S BED NUMBER
C

```
DO 10 J=1,25
  IF(XX(J).EQ.0.) GO TO 10
  XX(J)=XX(J)+UNFRM(-2,1,2,1)
  IF(XX(J).LT.48.) GO TO 10
  CALL STOPA(J)
  XX(J)=0.
  10 CONTINUE
  RETURN
  END
```

سابروتین INTLC مقداردهی اولیه شبیه سازی را انجام می دهد، ۱۸ بیمار را از طریق گروه (2)ENTER وارد شبکه می کند:

SUBROUTINE INTLC
COMMON/SCOM1/ ATRIB(100),DDL(100),DDL(100),DTNOW,11,MFA,MSTOP,NCLNR
1,NCRDR,NPRINT,NNRUN,NSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
ATRIB(1)=TNOW

C *** ASSIGN PATIENT SCORES AND PUT 18 PATIENTS IN THE SYSTEM
C

```
DO 10 J=1,18
  ATRIB(2)=UNFRM(30.,44.,1)
  CALL ENTER(2,ATRIB)
  RETURN
  END
```

خلاصه نتیجه شبیه سازی به صورت زیر است. آمار زمان هر بیمار در مرکز، زمان بین خالی بودن مرکز و کارائی ۲۵ فضای مرکز درمانی را بررسی می کند. که میانگین ۲۴/۸ از ۲۵ فضای موجود کارائی دارد نشان می دهد باید فضا اضافه شود.

بخش ششم

تعارفات

دستورات زبان SLAM II

جداول آماری

منابع



تمرینات

بخش ۱:

- ۱-۱- شبیه سازی را تعریف کنید.
- ۱-۲- تفاوت های شبیه سازی و مطالعه مستقیم در چیست؟
- ۱-۳- مراحل یک مطالعه شبیه سازی را بیان کنید.
- ۱-۴- سیستم را تعریف کنید و اجزای آنرا شرح دهید.
- ۱-۵- مفاهیم عدم قطعیت و قطعیت را شرح دهید و یک مثال اجتماعی برای هر کدام مطرح کنید.
- ۱-۶- یک مرتب سازی به روش عدم قطعیت برای ۱۰ عدد به زبان Dilag را بنویسید.
- ۱-۷- مدل چیست؟ انواع آنرا نام ببرید.
- ۱-۸- علت توجه بیشتر به مدل های گرافنی و تصویری چیست؟
- ۱-۹- مفهوم اجرای مدل را با یک مثال برای مدل ریاضی و مدل گرافنی بیان کنید.
- ۱-۱۰- با یک مثال تفاوت یک مدل گرافیکی اجرایی و غیراجرایی را بیان کنید.
- ۱-۱۱- ویژگی های یک مدل ریاضی را بررسی کنید.
- ۱-۱۲- تفاوت سیستم های پیوسته و گسسته را با یک مثال شرح دهید.
- ۱-۱۳- شرایطی که یک سیستم قطعی را به یک سیستم تصادفی تبدیل می کند چه نام دارد؟ درجه اهمیت آن در یک سیستم قطعی چیست؟
- ۱-۱۴- ارتباط یک سیستم تصادفی و سیستم قطعی در چیست؟
- ۱-۱۵- نقش شبیه سازی در مفهوم مجازی را بررسی کنید.
- ۱-۱۶- یک شبیه ساز هواپیما نیاز به چه اجزایی دارد؟
- ۱-۱۷- تفاوت و شباهت های شبیه سازی دستی و کامپیوتری را بررسی کنید.
- ۱-۱۸- بررسی کنید که برای شبیه سازی یک اسلایدر نیاز به چه پارامترهایی است؟
- ۱-۱۹- برای شبیه سازی سیستم آسانسور یک برج چند طبقه با حرکات افقی هر طبقه و عمودی در طبقات توسط اطافک آسانسور، سیستم و عناصر آنرا تحلیل و فهرست نمایید. (با فرض حرکت ۲ اطافک پاسخ دهید).
- ۱-۲۰- روتین مرتب سازی موازی قطعی را برای ۶ عدد ترسیم کنید (با زبان Dilag).

۲-۹- با استفاده از مقادیر ۲۷۲۹ و ۸۵۸۳ برای ۳۵ و ۲۶ روش میان ضریب پنج عدد تصادفی چهار

رقعی تولید کنید.

۲-۱۰- با استفاده از مقادیر ۶۷۸۷ برای ۴ و ۴۱۲۹ برای ۳۵ و روش ضریب ثابت، سه عدد تصادفی

چهار رقی تولید کنید.

۲-۱۱- براساس روش هم نهشتی سه عدد تصادفی دو رقی تولید کنید. با فرض آنکه $x_0=8$ و $x_1=27$ و $c=47$ و $m=100$ باشد.

۲-۱۲- اعداد تصادفی متوالی $r_1=0/45$ ، $r_2=0/57$ ، $r_3=0/89$ ، $r_4=0/11$ و $r_5=0/66$ را برای t_1 تا t_5 و t_6 را بیابید.

۲-۱۳- با استفاده از مقادیر $m=1000$ و $a=43$ و $x_0=17$ و $x_1=1000$ چهار عدد تصادفی سه رقی براساس روش هم نهشتی ضریبی تولید کنید.

۲-۱۴- با فرض تولید اعداد تصادفی $0/54$ ، $0/68$ ، $0/10$ و $0/68$ براساس روش آزمون کولموگروف - اسمیرنوف یکخواختی آنها را بررسی کنید.

۲-۱۵- یکخواختی مثال ۴.۱۷ را با فرض دسته‌های ۹ تایی بررسی کنید.

۲-۱۶- با بررسی مقدماتی ۱۰۰۰ عدد چهار رقی اطلاعات زیر بدست آمده است:

| چهار رقم مشابه | سه رقم مشابه | دو رقم | یک رقم | چهار رقم متفاوت | توایر مشابه |
|----------------|--------------|--------|--------|-----------------|-------------|
| ۲ | ۲۴ | ۱۷ | ۳۹۲ | ۵۶۵ | ۵۱۰ |

با آزمون یوکر استقلال آنها را بررسی نمایید. ($\alpha = 0/05$)

۲-۱۷- در ارتباط با مولدهای هم نهشتی زیر جهت حداکثر شدن پیرو توالی اعداد تصادفی، x_0 باید

دارای چه شرطی باشد؟

الف) زبان SIMSCRIPT با مولد هم نهشتی برای کامپیوتر CDC با خصوصیات زیر:

ب) مولد "سوپر دیر" کامپیوتر IBM 360 با خصوصیات زیر:

ب-۱) $m=722$ $C=0$ $a=4049$

ب-۲) $m=706$ $C=217$ $a=4901$

ب-۳) $m=1024$ $C=0$ $a=6007$

۲-۱۸- با استفاده از مولد هم نهشتی زیر ۴ عدد دو رقی بین صفر و ۲۴ تولید کنید.

$a=9$ $c=25$ $x_0=12$ ($b=6$)

بخش ۲:

۲-۱- برای اعداد تصادفی x_1 و x_2 که دارای توزیع نمایی با پارامتر $\lambda=1$ است، $P(x_1+x_2 > 2)$ را محاسبه کنید.

۲-۲- نشان دهید که توزیع هندسی یک توزیع بدون حافظه است.

۲-۳- تعداد شلیک یک توپ هوایی هارین دارای توزیع بواسون با میانگین $0/8$ است.

الف) احتمال آنکه بیش از ۲ شلیک در سال داشته باشد چقدر است؟

ب) احتمال آنکه فقط یک شلیک در سال داشته باشد چقدر است؟

۲-۴- اگر عمر یک ماهواره در مدار قرار گرفته (به سال) به وسیله pdf زیر تعریف شود:

$$f(x) = \begin{cases} 0/4 & 0 \leq x < 1 \\ e^{-0/4x} & x \geq 1 \end{cases}$$

سایر

الف) احتمال آنکه ماهواره بعد از ۵ سال هنوز از بین نرفته باشد چیست؟

ب) احتمال آنکه ماهواره بین ۳ و ۶ سال از زمان قرار گرفتن در مدار از بین برود چیست؟

۲-۵- برای یک متغیر تصادفی x با توزیع نمایی، مقدار λ را بیابید که در رابطه زیر صدق کند:

$$P(x < 2) = 0/9 P(x < 4)$$

۲-۶- اگر علامت ماشینها دارای شکل زیر باشد:

(عدد عدد حرف حرف) حرف براساس وزن ماشین تعیین می شود ولی عدد تصادفی است و در محدوده ۱ تا ۹۹۹ می باشد.

الف) احتمال آنکه دو عدد بعدی (یک عدد تصادفی) بزرگتر یا مساوی ۵۰۰ باشد چیست؟

ب) احتمال آنکه جمع دو عدد بعدی (یک عدد تصادفی) بزرگتر یا مساوی ۱۰۰۰ باشد چیست؟

۲-۷- زمان خرابی یک باتری کادیم دارای توزیع دبل با پارامترهای $\tau=0$ ، $\theta=\frac{1}{4}$ و $\alpha=\frac{1}{2}$ سال است.

الف) چه کسری از باتری‌های خرابی بین $\frac{1}{4}$ و $\frac{1}{2}$ سال دارند؟

ب) چه کسری از باتری‌های خرابی معادل میانگین عمر باطریها دارند؟

ج) چه کسری از باتری‌های خرابی تا $\frac{1}{4}$ سال دارند؟

۲-۸- با استفاده از مقدار اولیه ۶۳۹۳ و روش میان مربعی ده عدد تصادفی چهار رقی تولید کنید.

۲-۲۶ اگر تابع توزیع تجمعی متغیر تصادفی x به صورت زیر باشد. مولدی بیاید که مقادیر x را تولید کند:

$$F(x) = \begin{cases} 0 & x \leq 0 \\ \frac{1}{3}x & 0 < x \leq 1 \\ \frac{1}{3} & 1 < x \leq 2 \\ \frac{2}{3} - \frac{1}{6}x & 2 < x \leq 3 \\ \frac{1}{2} & 3 < x \leq 4 \\ \frac{1}{4} & 4 < x \leq 8 \\ 1 & x > 8 \end{cases}$$

۲-۲۷ اگر تابع چگالی احتمال یک متغیر تصادفی به صورت زیر باشد. مولدی بیاید که اعداد تصادفی با این چگالی را تولید کند.

$$f(x) = \begin{cases} \frac{1}{|1-x|} & 0 < x < 2 \\ \phi & \text{سایر} \end{cases}$$

۲-۲۸ با استفاده از روش حذفی، یک مولد عدد تصادفی با توزیع گاما تولید کنید.

$$(1/\theta) = 2/2 = 1/\theta^2 = 2/1 \text{ و } \theta = 2/3 \text{ و } \theta = 0/3545$$

۲-۲۹ مولد اعداد تصادفی x را بیایید که pdf آن عبارتست از:

$$f(x) = \begin{cases} e^{2x} & -\infty < x < 0 \\ e^{-2x} & 0 < x < \infty \end{cases}$$

۲-۳۰ مولد اعداد تصادفی x را بیایید که pdf آن عبارتست از:

$$f(x) = \begin{cases} \frac{1}{2}(x-2) & 2 \leq x \leq 3 \\ \frac{1}{2}(2-\frac{x}{3}) & 3 < x \leq 6 \\ 0 & \text{سایر} \end{cases}$$

۲-۳۱ مولد تصادفی با توزیع مثلثی در محدوده (۱ و ۱۰) و مد $x=4$ را بیایید.

۲-۳۲ مولد تصادفی با توزیع مثلثی در محدوده (۱ و ۱۰) و میانگین ۴ را بیایید.

۲-۳۳ با فرض pdf زیر یک مولد تصادفی برای این توزیع طرح کنید:

$$f(x) = \begin{cases} \frac{1}{3} & 2 \leq x \leq 2 \\ \frac{1}{24} & 2 < x \leq 10 \\ 0 & \text{در غیر این صورت} \end{cases}$$

۲-۱۹ برنامه‌ای به زبان C، فرترن یا بیسیک بنویسید که با دریافت پارامترهای m و c و براساس روشهای مختلف همبستگی پنج عدد تصادفی تولید کند.

۲-۲۰ از خصوصیات یک مولد همبستگی آن است که هیچ عدد موجود متوالی برای دفعه دوم در آن ظاهر نمی شود. راه ساده‌ای برای رفع این مورد پیشنهاد کنید.

۲-۲۱ برنامه‌ای به زبان C، فرترن یا بیسیک بنویسید که روشهای آزمون یکپارختی را برای یک سلسله تصادفی دلخواه بررسی کند.

۲-۲۲ استقلال مثال ۲-۲۲ را با فرض $\alpha=0/1$ بررسی کنید.

۲-۲۳ آزمون یوکر مثال ۲-۲۴ را با فرض ۲ رقم بررسی کنید.

۲-۲۴ چنانچه ۵۰۰۰ نمونه به صورت زیر اتفاق افتاده باشد با استفاده از روش خمی دو آنرا تحلیل

نمایید که آیا توزیع یکپارخت دارد؟

| محدوده | رخداد |
|---------|-------|
| ۰۰۰-۰۹۹ | ۴۶۸ |
| ۱۰۰-۱۹۹ | ۵۱۹ |
| ۲۰۰-۲۹۹ | ۴۸۰ |
| ۳۰۰-۳۹۹ | ۴۹۵ |
| ۴۰۰-۴۹۹ | ۵۱۵ |
| ۵۰۰-۵۹۹ | ۴۲۹ |
| ۶۰۰-۶۹۹ | ۴۹۷ |
| ۷۰۰-۷۹۹ | ۵۵۵ |
| ۸۰۰-۸۹۹ | ۴۶۳ |
| ۹۰۰-۹۹۹ | ۵۷۹ |

۲-۲۵ با فرض تصادفی بودن طول زمان مکالمات تلفنی، اگر دارای توزیع تجمعی زیر باشند. مولد اعداد تصادفی طرح کنید که طول زمان مکالمات تلفنی را تولید کند:

$$F(x) = \begin{cases} 1 - \frac{1}{2} e^{-(x/2)^2} & x > 0 \\ 0 & x \leq 0 \end{cases}$$

- ۳-۲- شیوه تبدیل یک فلورچارت برنامه نویسی به مدل شبکه پتری را با یک روژن نمونه ارائه نمایید.
- ۳-۴- مسئله معروف Reader/Writer با استفاده از n خواننده و نویسنده (شبکه پتری) را ترسیم کنید.
- ۳-۵- روابط تئوری شبکه پتری در رابطه با مسئله (۱-۳) را بنویسید.
- ۳-۶- اساس روابط شبکه پتری بر چه استوار است؟ مفاهیم Place, Token و Transition را شرح دهید.
- ۳-۷- شیوه تبدیل یک ماشین کامپیوتری (که روی pop کار می کند) را به مدل شبکه پتری شرح دهید.
- ۳-۸- مدل شبکه پتری وسیله مدلسازی چه سیستمهایی می باشد.
- ۳-۹- مفهوم Zeroesting (یا تست صفر) در شبکه پتری چیست؟ و چه کاربردی دارد؟
- ۳-۱۰- یک کامپیوتر با ۳ پروسه و ۴ منبع را با مدل شبکه پتری مدلسازی کنید. کارخوان، چاپگر، دیسک و دو بخش حافظه. هر پروسه می تواند در یک بخش از حافظه اجراء شود. پروسه ها عبارتند از:
- الف) پروسه ۱، تقاضای کارخوان و چاپگر را می دهد و سپس هر دو این منابع را آزاد می کند.
- ب) پروسه ۲، تقاضای کارخوان و دیسک را می دهد و سپس کارخوان را آزاد می کند و تقاضای چاپگر می دهد و در نهایت هر دو چاپگر و دیسک را رها می کند.
- ج) پروسه ۳، هر سه منبع را یکجا می خواهد و سپس تمامی آنها را رها می کند.
- ۳-۱۱- تمرین ۳۸-۴ را با مدل شبکه پتری ترسیم نمایید.
- ۳-۱۲- آیا می توانید مثال ۳۸-۴ را بازنوایی اجرا نمایید و شرایط اولیه آنرا فراهم کنید.
- ۳-۱۳- با استفاده از مدل شبکه پتری مدل یک جمع کننده سریال را طراحی کنید.
- ۳-۱۴- مسئله غذاخوردن فلاسفه (dining philosophers) پیشهادی دایکسترا را که با شبکه پتری طراحی شده بررسی و به روابط پتری تبدیل کنید (تصویر جلد کتاب).
- ۳-۱۵- آیا مدل شبکه پتری برای مدلسازی سیستم عالم یک کامپیوتر مفید است؟ چرا؟
- ۳-۱۶- ابزاری کامپیوتری طراحی کنید که با آن بتوان مدل شبکه پتری را طراحی و اجرا نمود.
- ۳-۱۷- ابزاری کامپیوتری طراحی کنید که با آن بتوان مدل شبکه پتری را طراحی و اجرا نمود.
- ۳-۱۸- تعیین کنید کدامیک از موارد زیر با استفاده از شبکه پتری قابل پیش بینی است:
- الف) تئوری صف
- د) جنگ نظامی
- ه) سیستمهای سیاسی
- و) سیستمهای اقتصادی
- ز) مدلسازی زبانهای طبیعی
- ۳-۱۹- تفاوت بین "e" و "e" را که در بخشهای ۲-۸ و ۳-۸ معرفی شده اند توضیح دهید.
- ۳-۲۰- الگوریتم ۱-۸ را بر روی جدول ۱-۸ انجام دهید.
- ۳-۲۱- برای اجرای هدر الگوریتم ۸-۱ قوانین را تعریف کنید. فرض کنید که ماشینها هم خوان هستند.

۳۳-۲- با فرض eff زیر یک مولد تصادفی برای این توزیع بنویسید:

$$F(x) = x^{1/6} \quad 0 \leq x \leq 2$$

۳۵-۲- با فرض eff زیر برای یک عدد تصادفی x:

$$F(x) = \frac{x(x+1)(2x+1)}{6(n+1)(2n+1)} \quad x = 1, 2, \dots, n$$

وقتی که $n=4$ است، سه مقدار x با استفاده از $R_1=0/83$ ، $R_2=0/24$ و $R_3=0/57$ را تولید کنید.

۳۶-۲- یک زیر برنامه BETA بنویسید که با استفاده از روش حذفی اعداد تصادفی با توزیع بتا را تولید کند.

۳۷-۲- اگر تابع توزیع احتمال ویبل به این صورت تعریف شود:

$$F(y) = 1 - e^{-ay^c} \quad y \geq 0, \quad c > 1, \quad a \geq 0$$

با استفاده از تبدیل معکوس یک زیر برنامه به نام WEIB بنویسید که N نمونه تصادفی با توزیع فوق را تولید کند.

۳۸-۲- اختلاف و شباهت شبیه سازی و روش مونت کارلو در چیست؟

۳۹-۲- برنامه ای بنویسید که با استفاده از روش مونت کارلو عدد π را تقریب بزند.

۴۰-۲- با استفاده از ایده مونت کارلو برنامه ای بنویسید که یک تصویر را داخل یک دایره روی صفحه مانیتور ظاهر کند. بدینوسیله است که موردهای افقی و عمودی نباید مشهود باشند و تصویر به تدریج به حالت وضوح تبدیل گردد.

۴۱-۲- برنامه ای به زبان C، فرتن یا پسیک برای مثالهای ۳-۵ و ۵-۵ بنویسید.

بخش ۳:

۳-۱- با استفاده از مدل شبکه پتری یک جمع کننده طراحی نمایید.

۳-۲- با استفاده از مدل پتری یک ضرب کننده طراحی نمایید.

۳-۲۲- الگوریتم λ^3 را تغییر دهید تا تمام ماشین را شامل شود.

۳-۲۳- الگوریتم ۸-۱ را روی ۸-۳ انجام دهید (با زبان برنامه‌نویسی مورد علاقه خودتان) یک دستورالعمل یا رسم‌الخط برای تعیین کردن سیستم ماشین‌های با حالت متناهی، ابداع کنید، جداول ۸-۴ و ۸-۵ را در این عمل بکار ببرید و از برنامه خود برای حداقل رساندن ماشینها استفاده (ماشینهای مختلف را به یک ماشین تبدیل کنید و اجرای نتیجه را شبیه‌سازی کنید).

۳-۲۴- روش جداول ۸-۶ و ۸-۷ را PROMELA در مدلسازی کنید.

۳-۲۵- در PROMELA عمل‌های chan, run را انجام دهید. سیستم مدلسازی شده را غیر محصور کنید.

۳-۲۶- یک الگوریتم پیدا کنید که مشخص کند، چه صف پیغامی از تعریف ارتباط ماشین با حالت متناهی در داخل آن استفاده شده است.

۳-۲۷- در (spash othama) آیا دو حالت ماشین مترادف هستند. اگر یکی از ۲ حالت شامل یک گذار غیر قابل اجرا باشد که حالت بعدی فاقد آن است؟

۳-۲۸- برنامه‌ای به زبان C بنویسید که یک مدل از جمع کننده پایتری را با مدل گرافیکی FSM دریافت کند و با تولید دو رشته تصادفی ورودی، سلسله عملیات (وضعیتها) را نمایش دهد و خروجی را تولید نماید (جمع آنها).

۳-۲۹- برنامه‌ای به زبان C بنویسید که با دریافت یک مدل گرافیکی FSM جدول خلاصه شده آنرا تولید کند.

۳-۳۰- ساختار داده‌ای یک مدل براساس گذار و وضعیت را طراحی کنید. قوانین اجرا را برای آن بنویسید.

بخش ۴:

۴-۱- مدل مسأله صف و یک سرویس دهنده را بنحوی تغییر دهید تا شبیه‌سازی برای مدتی معادل ۶۰ ساعت انجام شود.

۴-۲- در تمرین فوق تغییری بدهید تا هیچ متقاضی جدیدی بعد از ساعت ۶۰ وارد سیستم نشود.

۴-۳- زیر برنامه‌ای بنویسید که به صورت تصادفی (با توزیع دلخواه) پیش آمدهای مسأله صف و یک سرویس دهنده را تولید و در جدول FEL (فایل) ذخیره کند.

۴-۴- با استفاده از روش زمان‌بندی پیش آمدها سیستم اتار (M و N) را شبیه‌سازی کنید.

۴-۵- یک بزرگراه دو شهر را به یکدیگر متصل می‌کند. هر ۱۵±۲۰ ثانیه یک وسیله نقلیه شهر اول را به مقصد شهر دوم ترک می‌کند. از وسائط نقلیه مذکور ۲۰٪ دارای یک سرنشین، ۳۰٪ دارای ۲ سرنشین، ۱۰٪ دارای ۳ سرنشین و ۱۰٪ نیز دارای ۴ سرنشین هستند. ۳۰٪ وسائط نقلیه را اتوبوسها تشکیل می‌دهند که هر یک ۴۰ مسافر را حمل می‌کند. مدت مسافرت بین ۲ شهر برای هر وسیله نقلیه طبق توزیع آماری 10 ± 60 دقیقه تعریف می‌شود. مدت زمان لازم برای ورود ۵۰۰۰ نفر مسافر از شهر اول به شهر دوم را تخمین بزنید.

۴-۶- برنامه‌ای به زبان C بنویسید که خروجی مانند شکل ۳-۶-۱۰ برای صف و یک سرویس دهنده با تولید ورودی‌های تصادفی ترسیم کند.

۴-۷- مسافران پروازهای ورودی به یک فرودگاه از دو راهرو به محوطه اصلی فرودگاه و قسمت دریافت بار (چمدان) وارد می‌شوند. نرخ ورود مسافر به راهروی شماره یک و دو، به ترتیب عبارت از یک مسافر در 2 ± 10 ثانیه و یک مسافر در 5 ± 10 ثانیه است. مدتی که مسافر راهروی شماره یک و دو را طی می‌کند، به ترتیب 5 ± 3 ثانیه و 10 ± 25 ثانیه است. پس از آنکه مسافران به محوطه اصلی فرودگاه وارد شدند، ۶۰ درصد آنها به قسمت دریافت بار می‌روند. مدت لازم برای تحویل به قسمت بار با توزیع 3 ± 10 (ثانیه) تعریف می‌شود. این مسأله را برای ۵۰۰ مسافر شبیه‌سازی کنید و تعداد مسافرانی را که به قسمت بار مراجعه می‌کنید تخمین زده و آنرا با 30 ± 60 ٪ مقایسه کنید.

۴-۸- مرکز محاسبات یک دانشگاه دارای یک ترمینال کامپیوتری است. دانشجویان طبق توزیع 10 ± 15 (دقیقه) به آن مراجعه کرده و طبق توزیع 6 ± 12 دقیقه آنرا مورد استفاده قرار می‌دهند. به هنگام مراجعه دانشجویان لزوماً ترمینال آزاد نیست. ۶۰٪ از این دانشجویان پس از گذشت ۱۰ دقیقه مجدداً به ترمینال مراجعه می‌کنند، اگر باز هم ترمینال آزاد نباشد، ۵۰ درصد آنها (یعنی ۵۰ درصد از ۶۰ درصد) پس از گذشت ۱۵ دقیقه به ترمینال مراجعه می‌کنند (برای بار سوم). این مسأله را شبیه‌سازی کرده و در مقابل ۵۰۰ دانشجویی که موفق به استفاده از ترمینال می‌شوند تعداد دانشجویانی را تخمین بزنید که موفق به استفاده از ترمینال نشده‌اند (فرض کنید مرکز محاسبات در کلیه ساعات شبانه‌روز برای دریافت تقاضا و راه‌اندازی آن باز است).

۴-۹- شرکتی برای تأمین نیازهای محاسباتی خود به خرید و راه‌اندازی یک سیستم کامپیوتری اقدام کرده است. تقاضاها طبق توزیع 10 ± 10 (دقیقه) دریافت شده و پردازش می‌شوند. مدت پردازش هر تقاضا طبق توزیع 7 ± 7 (دقیقه) تعریف می‌شود. اداره کنندگان سیستم فوق، هر 60 ± 60 (دقیقه) باعث از کار افتادن آن می‌شوند. مدت اینگونه خرابی‌ها 4 ± 8 دقیقه است. پس از رفع خرابی، تقاضائی که احتمالاً به طور ناقص انجام شده است، کاملاً انجام نمی‌شود یعنی بخشی از تقاضا که پیش از وقوع

۱۲- مشتریان یک بانک با فواصل زمانی 35 ± 40 ثانیه از راه می‌رسند و به طور کاملاً تصادفی به یکی از دو باجه صندوق مراجعه می‌کنند. مدت راه‌اندازی مشتری توسط هر صندوقدار با توزیع 25 ± 75 ثانیه تعریف می‌شود. هرگاه مشتری در یکی از دو صف صندوق قرار گیرد تا اتمام دریافت خدمت در همان صف باقی خواهد ماند. برخی از مشتریان چنین پیشنهاد کرده‌اند که سیستم دو صفی به سیستم یک صفی با دو صندوق تغییر یابد. کدام سیستم برای متقاضیان سریعتر است؟ شبیه‌سازی را برای مدت ۲ ساعت و ۱۵ دقیقه انجام دهید بدون اینکه طی ۱۵ دقیقه اول اطلاعاتی گردآوری کنید. دو نحوه مختلف برای تشکیل صف و دریافت سروس را براساس ضریب اشتغال سروس دهنده، متوسط مدت انتظار متقاضی در صف و درصد متقاضیانی که تا شروع ارائه خدمت بیش از یک دقیقه و بیش از سه دقیقه به انتظار می‌مانند مقایسه کنید.

۱۴-۳- در یک منطقه وسیع شهری، هر 10 ± 15 دقیقه یک آمبولانس را راهی ارائه سروس به متقاضیان تلفنی می‌شود. با ورود آمبولانس به مقصد در 15 درصد از موارد چنین معلوم می‌شود که متقاضی واقعاً نیازی به دریافت خدمت ندارد. مدت رفت و برگشت آمبولانس در چنین مواردی با توزیع آماری 2 ± 12 دقیقه تعریف می‌شود. بقیه موارد تقاضای تلفنی به دو گروه تقسیم می‌شوند. گروه اول را تقاضاهای اضطراری تشکیل می‌دهند و برای آنها طبق توزیع آماری 55 ± 25 دقیقه صورت می‌گیرد. مدت راه‌اندازی تقاضاهای گروه دوم طبق توزیع آماری 10 ± 20 دقیقه تعریف می‌شود. از جمیع تقاضاهای دو گروه اخیر، 15 درصد را تقاضاهای گروه اول تشکیل می‌دهد. اگر فرض شود که با در اختیار داشتن تعداد کثیری آمبولانس می‌توان هر تعداد تقاضا را در هر لحظه از زمان پاسخگوئی کرد، مسئله را طوری شبیه‌سازی کنید که 500 تقاضا کاملاً راه‌اندازی شود و در انتها، متوسط مدت راه‌اندازی تقاضا تخمین زده شود.

الف- چنین فرض کنید که در مسئله قبل تنها یک آمبولانس وجود دارد. کدیه تقاضاهای تلفنی که طی مدت سروس دهی آمبولانس می‌رسند، تا زمان برگشتن آن در صف انتظار می‌مانند. آیا یک آمبولانس از عهده راه‌اندازی حجم تقاضای موضوع مسئله برمی‌آید.

ب- مسئله قبل را با تعداد ۱ آمبولانس (4 یا 2 یا 1) شبیه‌سازی کرده و این چهار سیستم را براساس طول مدت انتظار یک تقاضا در صف، درصد تقاضاهایی که ناچار از پیوستن به صف انتظار هستند و درصد زمانی که آمبولانس ارائه سروس دهی شده است، مقایسه کنید.

۱۵-۳- یک فرزند یگان مهاجر را شکار می‌کند. تا موقعی که او بیست برنده را شکار نکرده باشد باید در موقعیت ثابتی بماند. هر شلیک مدتی طبق توزیع آماری 1 ± 2 ثانیه طول می‌کشد. مدت تیر کردن تنگ هر با توزیع آماری 1 ± 3 ثانیه تعریف می‌شود. شکارچی از یک تنگ دولر استفاده می‌کند. به

خرابی انجام شده پس از رفع خرابی، دیگر نباید انجام شود. این مسئله را برای مدت ۲۴ ساعت شبیه‌سازی کنید و نتیجه‌هایی برای مدت به سر بردن یک تقاضا در سیستم و مدت تأخیر تقاضاهایی که طی زمان راه‌اندازی آنها سیستم دچار خرابی می‌شود ارائه کنید.

۱۰-۴- یک مرکز فوری کامیوتر دارای دو ترمینال است. دانشجویان با نرخ 2 ± 8 دقیقه به این مرکز وارد می‌شوند. راه‌اندازی تقاضاهای دانشجویان ممکن است به خاطر راه‌اندازی تقاضای استادان که با نرخ 2 ± 3 دقیقه به مرکز می‌رسند موقتاً قطع شود. در این مرکز یک تحلیل‌گر سیستم کار می‌کند که می‌تواند انجام تقاضای هر که را موقتاً قطع کند. البته ابتدا به قطع تقاضای دانشجویان و بعد به قطع تقاضای استادان اقدام می‌شود. در هر مرتبه تحلیل‌گر سیستم مدتی معادل 4 ± 6 دقیقه ترمینال را اشغال می‌کند و در فواصل زمانی 5 ± 20 دقیقه به ترمینال باز می‌گردد. دانشجویان و استادان مدتی معادل 4 ± 6 دقیقه ترمینال را مورد استفاده قرار می‌دهند. اگر تقاضایی موقتاً قطع شود، آن تقاضا به ابتدای صف انتظار منتقل می‌شود. این مسئله را شبیه‌سازی کنید به طوری که 50 تقاضا از استادان یا 50 تقاضا از جانب تحلیل‌گر سیستم انجام شود. نرخ قطع موقت تقاضاها در (ساعت) و متوسط زمان صف انتظار دانشجویان را تخمین بزنید.

۱۱-۴- پیامهایی که باید مخابره شوند با توزیع 10 ± 35 ثانیه از راه می‌رسند. زمانهای مخابره پیام با توزیع آماری 5 ± 20 ثانیه تعریف می‌شوند. در فواصل زمانی 3 ± 6 دقیقه، خط مخابره به پیامهای اضطراری که 3 ± 10 ثانیه طول می‌کشند اختصاص می‌یابد. تحت اینگونه شرایط، پیامی که مخابره آن ناتمام مانده به اول صف انتظار منتقل می‌شود تا مجدداً عمل مخابره آن انجام گیرد ولی برای این منظور می‌بایست چنین پیامی را مدت 2 دقیقه آماده کرد تا قابل مخابره شود. این مسئله را برای مدت 90 دقیقه شبیه‌سازی کنید و درصد مدتی که خط به وسیله پیامهای معمولی (غیر اضطراری) اشغال است را تخمین بزنید.

۱۲-۴- سی کامیون به حمل اجزاء و قطعات یک نوع هواپیمای باری از توران به شیراز اختصاص داده شده‌اند. براساس تجارب قبلی، چنین تصور می‌شود که مدت سفر هر کامیون باید با توزیع آماری 2 ± 6 ساعت تعریف شود. چهل درصد از راننده‌ها در توره‌خانه‌های بین راه توقف می‌کنند که این امر باعث صرف 5 ± 15 دقیقه می‌شود.

الف- اگر سی شبیه‌سازی این مسئله را طوری طراحی کنید که بتواند زمان ورود آخرین کامیون به مقصد را تخمین بزند (اگر احتمال توقف هر راننده در بین راه 40 درصد باشد).

ب- اگر دقیقاً چهل درصد از راننده‌ها در بین راه توقف کنند، اگر طوری طراحی کنید که بتواند زمان ورود آخرین کامیون به مقصد را تخمین بزند.

هر پرنده حداقل دو دفعه شلیک می‌شود و پُرکردن تفنگ بعد از هر تیراندازی به هر پرنده صورت می‌گیرد. هر 10 ± 2 ثانیه یک پرنده از بالای سر شکارچی پرواز می‌کند. احتمال موفقیت شکارچی در هر شلیک ۷۵ درصد است. شکار یست پرنده چقدر طول می‌کشد؟

۴-۱۶- هر 25 ± 1 ثانیه یک مشتری به قسمت فروش مواد پروتئینی در یک فروشگاه بزرگ مراجعه می‌کند. این قسمت به فروش گوشت مرغ و ماهی اشتغال دارد. ۵۰ درصد از مشتری‌ها خریدار گوشت مرغ و ماهی خریدار ماهی هستند. یست درصد بقیه خریدار هر دو نوع گوشت هستند. هر سرویس‌دهنده برای راه‌اندازی هر سفارش مدتی طبق توزیع آماری 45 ± 20 ثانیه صرف می‌کند. فرض کنید در هر لحظه از زمان به تعداد کافی سرویس‌دهنده برای راه‌اندازی مشتریان وجود دارد. این مسأله را تا لحظه راه‌اندازی سفارش (های) 200 آمین مشتری شبه‌سازی کنید.

۴-۱۷- در مسأله قبل حداکثر تعداد سرویس‌دهنده‌های مورد نیاز طی مدت شبه‌سازی چقدر است؟ آیا در اختیار داشتن این تعداد سرویس‌دهنده تضمین می‌کند که هیچگاه یک مشتری در صف انتظار قرار نگیرد؟

۴-۱۸- مسأله قبل را با تعداد $10, 20, 30, 40$ شبه‌سازی کنید. هرگاه سرویس‌دهنده‌ها مشغول باشند یک صف تشکیل می‌شود. به ازاء هر مقدار داده شده برای میانگین تعداد سرویس‌دهنده‌های مشغول را تخمین بزنید.

۴-۱۹- هر 15 ± 2 ساعت یک سفارش را به کارگاهی می‌رساند. کارگاه دارای یک سرویس‌دهنده است و سه نوع سرویس C و B و A را ارائه می‌کند. پنجاه درصد از سفارشات باید سرویس نوع A را دریافت کنند. سفارشات که سرویس B و C را نیاز دارند، به ترتیب ۳۰ درصد و ۲۰ درصد است. ارائه سرویس از نوع C و B به ترتیب طبق توزیعهای آماری 15 ± 5 ساعت، 10 ± 3 ساعت و 10 ± 3 ساعت صورت می‌گیرد. این مسأله را برای راه‌اندازی ۵۰ سفارش شبه‌سازی کنید. درصدهای مربوط به انواع سرویس را با نتایج بدست آمده از شبه‌سازی مقایسه کنید. آیا نتایج معقول به نظر می‌رسند؟

۴-۲۰- در یک مرکز سنجش شنوایی، معاینات از چند مرحله تشکیل می‌شود. مدت انجام معاینات مقدماتی طبق توزیع آماری 3 ± 1 دقیقه تعریف می‌شود. هشتاد درصد از بیماران در این مرحله سالم تشخیص داده شده و به مرحله بعد می‌روند. نیمی از یست درصد دیگر بیماران نیاز به معاینات بیشتری در همین مرحله دارند که مدت این معاینات مطابق توزیع آماری 2 ± 1 تعریف می‌شود. این بیماران مجدداً تحت معاینات مقدماتی قرار می‌گیرند و با همان احتمال هشتاد درصد ممکن است سالم تشخیص داده شوند. دیگر از یست درصد از معاینات سنجش شنوایی موفق بیرون نمی‌آیند و برایان مضرب دارو و کتوریز می‌شود و درود مراجعین به مرکز با توزیع آماری 5 ± 2 دقیقه تعریف

می‌شود. این مسأله را شبه‌سازی کنید و مدت لازم برای اتمام معاینه ۲۰۰ مراجعه‌کننده را تعیین کنید. (بدیهی است مراجعینی که دارو مصرف می‌کنند در شمار افراد سالم قرار نمی‌گیرند.)

۴-۲۱- در مشتق معادله (۱-۱) فرض شده که هزینه سفارش مجدد، مقدار ثابت و مستقل از تعداد واحدهای سفارش شده باشد. حالتی را در نظر بگیرید که هزینه جایگذاری سفارش برای q واحد و بدست آوردن آنها $r = m + aq$ می‌باشد که m و a دو مقدار ثابت اند، نشان دهید که این حالت در نهایت به معادله (۱-۱) می‌رسد.

۴-۲۲- تابع $f(x)$ حداقل x_0 می‌رسد اگر اولین مشتق $\frac{df}{dx} = 0$ و دومین مشتق $\frac{d^2f}{dx^2} > 0$ در x_0 باشد. دومین حالت را برای معادله (۱-۱) و (۵-۱۱) با ارزیابی $\frac{d^2f}{dx^2}$ ثابت کنید.

۴-۲۳- مدل ساده‌استفاده شده در مشتق فرمول اندازه ۱۰۰ معادله (۱-۱) فرض شده که سه پارامتر (تقاضای D ، هزینه سفارش مجدد c و هزینه حمل k) باشند. در عمل ممکن است خطاهایی در برآورد کردن این پارامترها بوجود آید. به طور دقیق آنالیز کنید تا نشان دهید که x درصد خطا در کدام از پارامترها فقط $x/2$ درصد خطا در هزینه کلی C بوجود می‌آورد (در نزدیکی EOQ).

همچنین نشان دهید که خطای x درصدی در هر پارامتر موجب خطای x درصدی در مقدار سفارش اقتصادی می‌شود.

۴-۲۴- یک دیاگرام شبه شکل (۱-۱) بکشید که برای محیط تولید صعود و سقوط انبار را نشان دهد. با استفاده از این دیاگرام نشان دهید که فرمول (۷-۱۱) بیشترین مقادیر صرفه اقتصادی را در چنین محیطی می‌دهد.

۴-۲۵- یک سیستم انبار شبه‌سازی شده با فرض توزیع یکسان تقاضاها، آن شبه‌سازی را تکرار کنید با فرض اینکه تقاضای توزیع نرمال باشد با میانگین ۵۰ و انحراف معیار ۱۰ است. دو نتیجه را با هم مقایسه کنید.

۴-۲۶- زمان هدایت را در مسئله شبه‌سازی انبار از ۳ روز به توزیع ارلنگ تغییر دهید. با تأخیر میانگین مقدار $\beta = 3$ روز و پارامتر $m = 2$ خروجی هر دو را مقایسه کنید.

۴-۲۷- شبه‌سازی اجرا شده در بخش ۳-۱۱ را تکرار کنید، تحت شرایط تغییر یافته: تقاضا از توزیع پواسون پیروی می‌کند در هر روز فقط یکبار با مقدار ۴ واحد. سطح سرویس را با میانگین حداقل موجودی به صورت منحنی بدست آورید.

۴-۲۸- (تأییدارها) مقدار اولیه به سیستم انباری بدهید، مانند تمام سیستمهای دینامیکی که مشخصه تأییدار برای زمانهای قبل از خود دردهی از خود نشان می‌دهد. سیستم انبار توضیح داده شده را شبه‌سازی کنید. تحت هر شرط آغازین متفاوت (مثلاً موجودی اولیه ۵۰۰ واحد و ۰ واحد) و

انبار را بطور پروردیدک بازنی می کنیم اما نقطه سفارش را در جای خود می گذاریم تا سطح موقعیت انبار را به Z برسائیم، فقط اگر موقعیت انبار در هر زمان بازنی در سطح P یا پایین تر باشد. اگر از P بالاتر باشد ما هیچ کاری نمی کنیم تا بازنی بعدی برسد.

۳۳-۴- در میان سه طرح سفارش مجدد: (P,Q) , (P,Z) , (P,Q) کدام رویه فکر می کنید بهترین هزینه را بدست می دهد؟ چرا؟

۳۴-۴- یک برنامه C برسید تا سیستم پیش گیری آماری را انجام دهد. مقدار α از ۰.۱ تا ۱.۰ افزایش می باید با گامهای ۰.۱

این برنامه ۲۴ مقدار داده را می خواند و آنرا در یک آرایه ذخیره می کند. برای ارزیابی اثرات هر مقدار α پیش بینی خطا برای هر پیش گیری باید چاپ شود.

۳۵-۴- برنامه ای به زبان C برسید که شبکه فعالیت مثال ۱۲۸ را پیاده سازی کند.

۳۶-۴- فرض کنید دو پروژه مختلف در اختیار شما است. این دو پروژه را به کارهای جدا، بدون هیچ رویهم اندازی تقسیم کنید. تقدم و تأخر کارها را تعیین کنید و برای هر کدام یک شبکه فعالیت ترسیم کنید.

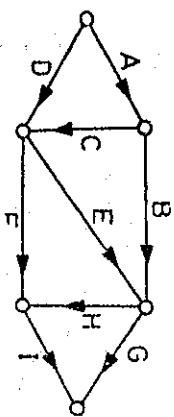
۳۷-۴- برنامه ای به زبان C برسید که با دریافت مدل شبکه PERT (دایره ها و اتصالات) به صورت گرافیکی بتواند آنرا اجرا و بررسی کند:

الف) نقاط بحرانی را تعیین کند.

ب) شبکه را اجرا نماید.

ج) مدل انتخابی شما چه مدلی است؟ آیا مدل شبکه پتری جویگی این امر است؟

۳۸-۴- یک شبکه فعالیت با ۹ فعالیت به صورت زیر داده شده است. تمامی دوره فعالیتها دارای توزیع یکساخت با محدوده $\pm 20\%$ درصد از میانگین هستند. میانگین مقادیر دوره فعالیتها $A=6.5, B=4.5, C=5.5, D=7.5, E=6.5, F=4.5, G=5.5, H=6.5, I=7.5$ و عبارتند از $0.1, 0.2, 0.1, 0.2, 0.1, 0.2, 0.1, 0.2, 0.1$ شبکه را برای 1000 اجرا شبیه سازی کنید تا هیستوگرام مشابه شکل ۹-۱۲ بدست آید. شبیه سازی را با روش تحلیل قطعی با استفاده از مقادیر میانگین مقایسه کنید.



موجودی روزانه در دسترس را با زمان اولیه ۲۵ روز طرح کنید (برای مشاهده ناپایداریست). در مورد منحنی ها بحث کنید.

۲۹-۴- (جستجو برای رویه بهینه) برای سیستم انبار ما فقط ۵ رویه متفاوت (P,Q) را در نظر می گیریم و بهترین آنها را انتخاب می کنیم. بهترین آنها حتماً بهترین تمام (P,Q) های ممکن نیست. چگونگی بهترین را جستجو می کنید؟ (نشان دهید) طرح و اجرای یک جستجو روی دو مقدار کنترل Q برای پیدا کردن رویه بهینه با استفاده از روش صعود دشوار).

۳۰-۴- (انبار خراب شدنی) محصولات فاسد شدنی مانند میوه، سبزیجات، تکه های یخ و روزنامه ها فقط برای مدت محدودی می توانند انبار شوند، چون بعد از مدتی مقداری یا تمامی ارزش خود را از دست می دهد. مثال شرایط زیر را در نظر گرفته است: روزنامه فروشی روزنامه های روزانه را ۲۰ ریال می خرد و ۳۵ ریال می فروشد. آنهایی که فروخته نمی شود بعداً به عنوان کاغذ باطله هر کدام را ۵ ریال می فروشد. توزیع تقاضای روزانه تخمین زده شده مانند زیر است:

| احتمالات | تقاضا (تعداد نسخه ها) |
|----------|-----------------------|
| 0.05 | 100 |
| 0.10 | 110 |
| 0.20 | 120 |
| 0.35 | 130 |
| 0.25 | 140 |
| 0.05 | 150 |

برنامه کامپیوتری بنویسید که رویه های مختلف را شبیه سازی کنید. (برای تعیین بهترین تعداد نسخه که روزنامه فروش باید تهیه کند تا سود بیشتری پیدا کند).

۳۱-۴- یک برنامه کامپیوتری بنویسید تا یک سیستم انبار را شبیه سازی کند که از طرح مرور دورهای (TZ) استفاده کند. سه هزینه تقاضا، زمان هدایت و موجودی اولیه هم هستند. (مانند سیستم شبیه سازی ایالت) برنامه خود را برای مقایسه هزینه های چندین رویه (TZ) برای پرورد ۵۰۰ روز استفاده کنید.

۳۲-۴- ما دو نوع سفارش مجدد با طرح و نقشه در نظر گرفتیم به نامهای (P,Q) و (TZ) . در این مسئله شما باید ترکیب هر دو را شبیه سازی کنید که به نام رویه (P,Z) نامیده می شود. در اینجا ما موقعیت

۴-۴۲- فرض کنید خروجی ۴۰ عدد x_1 به صورت زیر است. محاسبه کنید: الف) میانگین نمونه‌ها
ب) واریانس.

| | | | | | |
|----------------|-----------------|----------------|----------------|----------------|----------------|
| $\frac{9}{1}$ | $\frac{8}{9}$ | $\frac{10}{3}$ | $\frac{9}{9}$ | $\frac{11}{5}$ | $\frac{11}{3}$ |
| $\frac{10}{8}$ | $\frac{10}{6}$ | $\frac{9}{3}$ | $\frac{9}{9}$ | $\frac{9}{8}$ | $\frac{4}{7}$ |
| $\frac{10}{8}$ | $\frac{10}{8}$ | $\frac{10}{7}$ | $\frac{10}{1}$ | $\frac{9}{4}$ | $\frac{9}{5}$ |
| $\frac{9}{8}$ | $\frac{10}{10}$ | $\frac{9}{9}$ | $\frac{9}{4}$ | $\frac{10}{8}$ | $\frac{10}{7}$ |
| $\frac{9}{8}$ | $\frac{10}{3}$ | $\frac{10}{4}$ | $\frac{9}{5}$ | $\frac{10}{3}$ | $\frac{11}{1}$ |

۴-۴۵- سلسله اعداد موجود در تمرین ۴-۴۲ را به ۸ بلوک تقسیم کنید که طول هر یک ۵ است. میانگین و واریانس داده‌های بلوک شده را محاسبه کنید.

۴-۴۶- شبیه‌سازی را برای سیستم انبار اجرا کنید. فلچرپارت برای $p = 120$, $q = 80$, $stock = 0$ در نظر گرفته شده است. در نظر بگیرید موجودی روزانه را به عنوان خروجی و موقعیتی برای ۱۰۰ روز به منظور نشان دادن پایداری تکرار کنید ($stock = 100$ واحد).

۴-۴۷- شبیه‌سازیهای بخش ۴-۱۱ را برای ایجاد یک سیستم انبار مطلوب بررسی نمایید.

۴-۴۸- فرض کنید x_1, x_2, \dots, x_n نمونه‌های مستقلی هستند که میانگین‌شان μ_1 و μ_2 و واریانس آنها σ_1^2 و σ_2^2 باشد. \bar{x} میانگین‌های دو نمونه را بیابید. واضح است که متغیرهای تصادفی \bar{x} و \bar{y} میانگین μ_1 و μ_2 و واریانس $\frac{\sigma_1^2}{n}$ و $\frac{\sigma_2^2}{n}$ دارند. متغیرهای تصادفی (\bar{x}, \bar{y}) میانگین (μ_1, μ_2) و واریانس $(\frac{\sigma_1^2}{n}, \frac{\sigma_2^2}{n})$ دارند.

الف) نشان دهید که متغیر $t = \frac{(\bar{x} - \bar{y}) - (\mu_1 - \mu_2)}{S\sqrt{1/n + 1/m}}$ دارای توزیع t دانش آموز (student-t) با درجه آزادی $n+m-2$ است و S^2 میزان بدست آمده واریانس می‌باشد

با فرض معادله زیر:

$$S^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2 + \sum_{j=1}^m (y_j - \bar{y})^2}{n+m-2}$$

ب) دو سری از نمونه‌های زیر را در نظر بگیرید:

$$X = 79, 84, 120, 120, 108, 114, 103, 122, 120$$

$$Y = 54, 99, 99, 91, 103, 90, 113, 108, 87, 100, 80$$

محاسبه کنید x, y, S^2 این فرضیه را که دو میانگین با هم برابر هستند بررسی کنید. این معنا که $(\mu_1 = \mu_2)$ مساوی صفر است.

بخش ۵:

۵-۱- زبانهای شبیه‌سازی گسسته را دسته‌بندی کنید و برای هر دسته مثالی بزنید.

۵-۲- ساختار زبانهای پروسه‌گرا و مزایا آنرا بیان کنید.

۵-۳- تفاوت زبانهای GPSS, SLAM و SIMULA چیست؟

۵-۴- تفاوت زبانهای پیوسته و گسسته در چیست؟

۵-۵- دو نمونه از زبانهای پروسه‌گرا و فعالیت‌گرا را بیان کنید.

۴-۳۹- نشان دهید که در بازی قمار احتمال برنده شدن بازیکن $288/990$ می‌باشد. بنابراین در هر بازی که صورت می‌گیرد در 488 بازی برنده و در 502 بازی شکست می‌خورد. همچنین مقدار قابل انتظار هر بازی که برنده شد، $14/990 = 0/014$ است.

۴-۴۰- در این تمرین از شما خواسته شده که به منظور تقلیل واریانس در شبیه‌سازی از اصل نمونه‌های متقابل استفاده شد. (شکل ۱۳۸) برای 500 جفت انجام شده عبارت است از بدست آوردن مدت زمان فعالیت مفرض، ابتدا با یکاگیری اعداد تصادفی V_1, V_2 در فرمول پاکس مولر برای دومین جفت از اعداد (U_1, U_2) و (U_3, U_4) به منظور ایجاد مدت فعالیت. با استفاده از بخش ۴-۱۳ برنامه را براساس خصوصیات فوق بنویسید و نتایج اجرای یک شبیه‌سازی با طول 1000 را با یک جفت اجرای شبیه‌سازی با طول 500 را مقایسه کنید.

۴-۴۱- فرمول ۴-۱۳ برای تعیین اندازه نمونه با این فرض که میانگین نمونه x توزیع نرمالی داشته باشد استفاده می‌شود. در بعضی از روشهای شبیه‌سازی این فرض ممکن است سؤال برانگیز باشد. در این صورت اصل نابرابری چپی شاف می‌تواند مورد استفاده قرار بگیرد. این اصل بیانگر این موضوع می‌باشد که در یک سری از اندازه‌های مستقل y_1, y_2, \dots, y_n حداقل این کسر $(\frac{1}{k_2})$ بین تقسیمات استاندارد (K) میانگین μ خواهد بود. به عبارت دیگر $\frac{1}{K} \leq \frac{y_i - \mu}{\sigma} \leq \frac{1}{K}$ این فرمول را با نمونه x استفاده می‌کنیم و به این صورت تبدیل می‌شود:

از این اختلاف برای اندازه نمونه n که برای تعیین میانگین x بین $K \pm \mu$ و $c = 0/05$ برای چندمین مقدار K از $0/5$ تا $0/5$ استفاده کنید. مقایسه کنید مقادیر n را که براساس استفاده از معادله ۴-۱۳ بدست آمده است و نشان دهید که اگر $c = 0/05$ باشد می‌بایست 513 بار نمونه انتخاب کنید.

۴-۴۲- در بسیاری از مطالعات، شبیه‌سازی خروجی به شکل مطلوب می‌باشد به عنوان مثال در شبیه‌سازی یک جنگ افزار ارتش (قابلیت اطمینان) نشان دهید در مواقعی که خروجی دارای نوع خاصی است. تعداد نمونه‌های مورد نیاز n برای تعیین محاسبه میزان مطلوب در محدوده کسر f با

$$n = \left(\frac{1-f}{f^2} \right)^2 \text{ برابر } n = \left(\frac{1-f}{f^2} \right)^2 \text{ می‌باشد.}$$

ضریب اطمینان $(1-\alpha)$ برابر $n = \left(\frac{1-f}{f^2} \right)^2$ می‌باشد.

۴-۴۳- دو عامل ثابت پشت سرهم برای احتمال $1-\alpha$ می‌باشد.

۴-۴۴- در بخش ۱-۱۳ وقت میانگین x را مشاهده کردیم. بعضی اوقات با دقت واریانس نیز سروکار داریم. فرمولی مشابه معادله ۴-۱۳ برای تعیین تعداد مشاهدات مورد نیاز n را به منظور محاسبه واریانس q در محدوده $\pm 2p$ درصد با احتمال $(1-\alpha)$ بدست آورید

$$n = 1 + \frac{\left(\frac{2}{p} \right)^2}{\left(\frac{1-f}{f^2} \right)^2}$$

