قسمت اول

مقدمه‌ای بر شبیه‌سازی سیستمهای گسسته-پیشامد

۱

مقدمه‌ای بر شبیه‌سازی

شبیه‌سازی تقلیدی از عملکرد فرایند یا سیستم واقعی با گذشت زمان است. شبیه‌سازی، صرفنظر از اینکه با دست یا به‌وسیلهٔ کامپیوتر انجام شود، به ایجاد ساختگی تاریخچهٔ سیستم، و بررسی آن به منظور دستیابی به نتیجه‌گیریهایی در مورد ویژگیهای عملکرد سیستم واقعی مربوط می‌شود.

همچنانکه یک سیستم با گذشت زمان تکوین می‌یابد، رفتار آن با ایجاد مدل شبیه‌سازی بررسی می‌شود. این مدل معمولاً به شکل مجموعه‌ای از فرضهای مربوط به عملکرد سیستم است. این فرضها در چارچوب رابطه‌های ریاضی، منطقی و نمادین بین نهادها یا اهداف موردنظر سیستم بیان می‌شود. با ایجاد و معتبرسازی مدل، می‌توان آن را برای تفحص دربارهٔ پرسشهای بسیار گوناگونی از نوع «چه شود اگر» در مورد سیستم واقعی به‌کار برد. تغییرات انجامپذیر در سیستم را می‌توان ابتدا شبیه‌سازی کرد تا تأثیرشان بر عملکرد سیستم پیش‌بینی شود. شبیه‌سازی به منظور بررسی سیستمهای در دست طراحی نیز پیش از ایجاد آنها کاربردپذیر است. پس، ایجاد مدل شبیه‌سازی، هم به منزلهٔ ابزار تحلیل برای پیش‌بینی تأثیر تغییرات سیستمهای موجود و هم به‌عنوان ابزار طراحی برای پیش‌بینی عملکرد سیستم جدید در مجموعه‌های گوناگون شرایط کاربردپذیر است.

در برخی موارد می‌توان مدلی چنان ساده ایجاد کرد که به راحتی تمام با روشهای ریاضی «حل شود». چنین راه‌حلهایی را می‌توان با استفاده از حساب دیفرانسیل، تئوری احتمال، روشهای جبری، یا سایر روشهای ریاضی به‌دست آورد. این راه‌حلها معمولاً چند پارامتر عددی را دربرمی‌گیرد که معیارهای سنجش عملکرد سیستم نام دارند. اما بسیاری از سیستمهای واقعی چنان پیچیده‌اند که حل ریاضی مدلهایشان در عمل ناممکن است. در این‌گونه موارد، به منظور تقلید رفتار سیستم با گذشت زمان، می‌توان از شبیه‌سازی عددی کامپیوتری استفاده کرد. با شبیه‌سازی، چنان داده‌هایی فراهم می‌آید که گویی سیستم واقعی را مشاهده می‌کرده‌ایم. از داده‌های به‌وجود آمده از شبیه‌سازی،

برای براورد کردن معیارهای سنجش عملکرد سیستم استفاده می‌کنند.

در این کتاب بررسی مقدماتی مفاهیم و روشهای گونه‌ای از طراحی مدل شبیه‌سازی را ارائه می‌کنیم که طراحی مدل شبیه‌سازی پیشامدهای گسسته نام دارد. در فصل اول، ابتدا در این‌باره که چه وقت باید از شبیه‌سازی استفاده کرد، مزایا و ایرادهای شبیه‌سازی و زمینه‌های واقعی به‌کارگیری آن بحث می‌کنیم. سپس، مفاهیم سیستم و مدل را بررسی می‌کنیم و سرانجام، خلاصه‌ای از گامهای مربوط به ساختن مدل شبیه‌سازی سیستم و به‌کارگیری آن را ارائه می‌دهیم.

۱-۱ شبیه‌سازی چه وقت ابزار مناسبی شمرده می‌شود؟

در دسترس بودن زبانهای ویژهٔ شبیه‌سازی، تواناییهای محاسباتی گسترده با هزینهٔ رو به کاهش هر محاسبه، و پیشرفتهایی در روشهای شبیه‌سازی، این مبحث را به‌صورت یکی از رایجترین و پذیرفته‌ترین ابزار تحقیق در عملیات و تحلیل سیستمها در آورده است. شرایطی را که در آن شبیه‌سازی ابزار کاربردپذیر مناسبی است نویسندگان بسیاری از جمله نیلور و همراهان [۱۹۶۶] مورد بحث قرار داده‌اند. شبیه‌سازی را می‌توان برای انجام مقاصد زیر به‌کار گرفت:

۱. با شبیه‌سازی بررسی و آزمایش رابطه‌های متقابل هر سیستم یا زیرسیستم پیچیده میسر است.

۲. تغییرات اطلاعاتی، سازمانی و محیطی را می‌توان شبیه‌سازی کرد و به مشاهدهٔ تأثیر این تغییرات بر رفتار مدل پرداخت.

۳. شناخت به‌دست آمده از طریق طراحی مدل شبیه‌سازی، ممکن است به هنگام پیشنهاد انجام اصلاحات در سیستم در دست بررسی، ارزش فراوانی داشته باشد.

۴. با ایجاد تغییر در ورودیهای شبیه‌سازی و بررسی خروجیهای به‌دست آمده، می‌توان شناخت ارزشمندی دربارهٔ مهمترین متغیرها و چگونگی رابطهٔ متقابل آنها به‌دست آورد.

۵. شبیه‌سازی را می‌توان همچون ابزاری آموزشی به منظور تقویت روشهای تحلیلی پاسخیابی به‌کار گرفت.

۶. از شبیه‌سازی می‌توان به منظور آزمایش طرحها یا خط‌مشیهای جدید پیش از اجرای آنها استفاده کرد و آمادگی لازم را برای رو‌به‌رو شدن با پیشامدهای ممکن به‌دست آورد.

۷. شبیه‌سازی را می‌توان به منظور تحقیق دربارهٔ پاسخهای تحلیلی مورد استفاده قرار داد.

۱-۲ مزایا و معایب شبیه‌سازی

هر چند شبیه‌سازی ابزار مناسبی برای تحلیل در موارد بسیار است، تحلیلگر سیستم پیش از به‌کارگیری این روش در هر مورد خاص، باید مزایا و عیبهای آن را در نظر داشته باشد. مزایای اساسی شبیه‌سازی، که اشمید و تیلور [۱۹۷۰] و سایرین دربارهٔ آن بحث کرده‌اند، به شرح زیر است:

۱. پس از ساختن هر مدل می‌توان به منظور تحلیل طرحها یا خط‌مشیهای پیشنهادی، بارها آن را به‌کار گرفت.

۲. از روشهای شبیه‌سازی می‌توان در کمک به تحلیل هر سیستم پیشنهادی استفاده کرد، هر چند که داده‌های ورودی تقریبی و ناقص باشد.

۳. معمولاً دستیابی به داده‌های شبیه‌سازی بسیار کم‌هزینه‌تر از فراهم آوردن داده‌های مربوط به سیستم حقیقی است.

۴. به‌کار بردن روشهای شبیه‌سازی معمولاً آسانتر از روشهای تحلیلی است. بنابراین، شمار استفاده‌کنندگان بالقوهٔ روشهای شبیه‌سازی بسیار بیشتر از روشهای تحلیلی است.

۵. در حالی که معمولاً مدلهای تحلیلی به فرضهای ساده‌کنندهٔ بسیار نیاز دارند تا از لحاظ ریاضی کاربردپذیر شوند مدلهای شبیه‌سازی چنین محدودیتهایی ندارند. با استفاده از مدلهای تحلیلی، معمولاً تحلیلگر می‌تواند تنها تعدادی محدود از معیارهای سنجش عملکرد سیستم را محاسبه کند، در صورتی‌که داده‌های تولیدشده از مدلهای شبیه‌سازی به منظور براورد هر معیار سنجش متصور عملکرد کاربردپذیر است.

۶. در برخی موارد شبیه‌سازی تنها وسیلهٔ یافتن راه‌حل مسأله است.

اشمید و تیلور فهرست عیبهای شبیه‌سازی که باید پیش از به‌کارگیری آن بررسی شود را نیز ارائه کرده‌اند:

۱. مدلهای شبیه‌سازی مربوط به کامپیوترهای رقمی ممکن است پرهزینه باشند، زیرا ساخت و معتبرسازی آنها به زمان قابل‌توجهی نیاز دارد.

۲. معمولاً، به اجراهای فراوانی در مورد هر مدل شبیه‌سازی نیازمندیم و همین مسأله ممکن است به هزینه‌های زیادی برای به‌کارگیری کامپیوتر بیانجامد.

۳. گاهی شبیه‌سازی را در شرایطی به‌کار می‌برند که روشهای تحلیلی کافی به نظر می‌رسد. این وضعیت در مواردی پیش می‌آید که استفاده‌کنندگان با روش شبیه‌سازی آشنا می‌شوند و آموخته‌های ریاضی خود را به فراموشی می‌سپرند.

در مقام دفاع از شبیه‌سازی، باید گفت که دو ایراد نخست اشمید و تیلور (و دیگران، مثل ادکینز و پوج [۱۹۷۷])، با در دسترس قرارگرفتن زبانهای مخصوص شبیه‌سازی و کامپیوترهایی با قدرت روزافزون که به ازای هر واحد پول، عملیات بیشتری انجام می‌دهند، اصلاح شده است. در باب چند زبان مخصوص شبیه‌سازی در فصل ۳ بحث کرده‌ایم.

۱-۳ زمینه‌های کاربرد

کاربردهای بسیاری از شبیه‌سازی در انواع زمینه‌های بسیار وجود داشته است. هیلی‌یر و لیبرمن [۱۹۸۰] مثالهای زیر را برای نمایانیدن توانایی گستردهٔ روش شبیه‌سازی برمی‌شمارند:

۱. شبیه‌سازی عملیات در فرودگاههای بزرگ توسط شرکتهای هواپیمایی به منظور آزمودن تغییرات خطی‌مشیها و عملکردهای خود (مثلاً ظرفیت نگهداری و تعمیر، امکانات سوار و پیاده

کردن مسافر، هواپیمای کمکی، و . . .).

۲. شبیه‌سازی گذر وسایل حمل و نقل از تقاطعی که چراغهای راهنمایی دارد با برنامهٔ منظم زمانی، به منظور تعیین بهترین توالیهای زمانی.

۳. شبیه‌سازی عملیات نگهداری و تعمیر به منظور تعیین شمار بهینهٔ افراد گروههای تعمیراتی.

۴. شبیه‌سازی جریان شارژنشدهٔ ذرات از سپر تشعشعی به منظور تعیین شدت تشعشعی که از سپر می‌گذرد.

۵. شبیه‌سازی عملیات فولادسازی به منظور ارزیابی تغییرات در طرز انجام عملیات و ظرفیت و ترکیب امکانات.

۶. شبیه‌سازی اقتصاد کشور به منظور پیش‌بینی تأثیر تصمیمات مربوط به خط‌مشی اقتصادی.

۷. شبیه‌سازی جنگهای نظامی بزرگ مقیاس به منظور ارزیابی سیستمهای تسلیحاتی تدافعی و تهاجمی.

۸. شبیه‌سازی سیستمهای بزرگ مقیاس توزیع و کنترل موجودی به منظور اصلاح طراحی اینگونه سیستمها.

۹. شبیه‌سازی تمامی عملیات هر بنگاه تجاری به منظور ارزیابی تغییرات وسیع در خط‌مشی‌ها و عملیات آن و همچنین فراهم آوردن امکان شبیه‌سازی عملیات تجاری به منظور آموزش مدیران.

۱۰. شبیه‌سازی سیستم ارتباطات تلفنی به منظور تعیین ظرفیت اجزای مورد نظر که از لحاظ ارائهٔ رضایت‌بخش خدمت در اقتصادی‌ترین سطح ممکن لازم است.

۱۱. شبیه‌سازی عملکرد حوضهٔ توسعه‌یافتهٔ رودخانه‌ای به منظور تعیین بهترین ترکیب سدها، کارخانه‌های تولید برق و عملیات آبیاری چنانکه بتوان سطح مطلوب مهار سیلابها و توسعهٔ منابع آب را تأمین کرد.

۱۲. شبیه‌سازی عملیات خط تولید به منظور تعیین مقدار فضای لازم برای انبار کردن مواد در دست تولید.

۱-۴ سیستمها و پیرامون سیستم

برای مدلسازی سیستم، درک مفهوم سیستم و مرز سیستم لازم است. سیستم را به منزلهٔ گروهی از اشیاء تعریف می‌کنند که در راستای تحقق مقصودی معین در چارچوب رابطه یا وابستگی متقابل منظم به هم پیوسته باشند. مثالی از سیستم عبارت از سیستم تولیدی ساخت خودرو است. ماشینها، قطعات و کارگران با هم در امتداد خط مونتاژ کار می‌کنند تا وسیلهٔ نقلیه‌ای با کیفیت بالا تولید کنند.

هر سیستم اغلب تحت تأثیر تغییراتی قرار می‌گیرد که در خارج از سیستم روی می‌دهند. گفته‌می‌شود که چنین تغییراتی در *پیرامون* سیستم روی می‌دهند [گوردون، ۱۹۷۸]. در مدلسازی

سیستمها لازم است که مرز بین سیستم و پیرامون آن تعیین شود. چگونگی تعیین این مرز ممکن است به مقصود از مطالعهٔ سیستم بستگی داشته باشد.

مثلاً، در مورد سیستم کارخانه، می‌توان عوامل کنترل‌کنندهٔ ورود سفارشها را خارج از اختیار کارخانه و در نتیجه بخشی از پیرامون آن به‌شمار آورد. اما اگر قرار باشد تأثیر عرضه بر تقاضا را در نظر بگیریم، بین محصول کارخانه و ورود سفارشها رابطه‌ای وجود خواهد داشت و چنین رابطه‌ای را باید همچون یکی از فعالیتهای سیستم مورد توجه قرار داد. همچنین، در مورد سیستمی چون بانک، ممکن است حدی بر بیشترین نرخ بهرهٔ پرداختی وجود داشته باشد. در بررسی تنها یک بانک، این مسأله به منزلهٔ محدودیتی پیرامونی است. اما در مقام بررسی تأثیرات قوانین پولی بر صنعت بانکداری، تعیین حد در زمرهٔ فعالیتهای سیستم شمرده می‌شود [گوردون، ۱۹۷۸].

۱-۵ اجزای سیستم

به منظور درک و تحلیل سیستم، چند واژه را تعریف می‌کنیم. نهاد، عنصری مورد توجه در سیستم است. خصیصه، ویژگی نهاد است. هر فعالیت نمایشگر دوره‌ای زمانی با طول مشخص است. اگر دربارهٔ بانکی بررسی می‌کنیم، مشتریان را می‌توان نهاد دانست، موجودی حسابهای جاری آنها را خصیصه و سپرده‌گذاری را فعالیت به حساب آورد.

مجموعهٔ نهادهایی که کل سیستم را در مورد یک بررسی شکل می‌دهد ممکن است در بررسی دیگر تنها زیرمجموعه‌ای از کل سیستم باشد [لا و کلتون، ۱۹۸۲]. مثلاً، اگر بانک پیش گفته به منظور تعیین تعداد تحویلداران مورد نیاز برای دریافت و پرداخت مورد بررسی قرار گیرد، سیستم را می‌توان بخشی از کل بانک، مشتمل بر تحویلداران دائم آن و مشتریان منتظر در صف تعریف کرد. اگر مقصود بررسی را به تعیین تعداد تحویلداران ویژهٔ مورد نیاز (برای کارسازی چکهای بانک، فروش چکهای مسافرتی، و . . .) تعمیم دهیم، تعریف سیستم را نیز باید وسیعتر در نظر بگیریم.

مجموعهٔ متغیرهای لازم برای تشریح سیستم در هر زمان، با توجه به اهداف بررسی را حالت سیستم تعریف می‌کنیم. در بررسی بانک، متغیرهای ممکن حالت عبارت‌اند از: تعداد تحویلداران سرگرم کار، تعداد مشتریان منتظر در صف یا در حال خدمتگیری و زمان ورود مشتری بعدی. پیشامد را رویدادی لحظه‌ای تعریف می‌کنیم که بتواند حالت سیستم را تغییر دهد. واژهٔ درونزا به منظور تشریح فعالیتها و پیشامدهایی که در درون سیستم رخ می‌دهند و واژهٔ برونزا به منظور تشریح فعالیتها و پیشامدهای پیرامونی که سیستم را تحت تأثیر قرار می‌دهند به‌کار می‌رود. در بررسی مربوط به بانک، ورود هر مشتری، پیشامدی برونزا و کامل‌سازی خدمتدهی به هر مشتری پیشامدی درونزاست.

جدول ۱-۱ فهرست مثالهایی از نهاد، خصیصه، فعالیت، پیشامد، و متغیرهای وضعیت را در مورد چند مسأله ارائه می‌کند. تنها بخشی از فهرست اجزای سیستم نشان داده شده است.

جدول ۱-۱ مثالهایی دربارهٔ سیستم و اجزای آن.

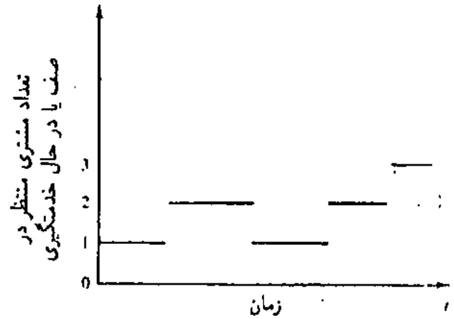
سیستم	نهادها	خصیصه‌ها	فعالیتها	پیشامدها	متغیرهای حالت
بانک	مشتریان	ماندهٔ حساب جاری	سپرده‌گذاری	ورود، ترک	تعداد خدمت‌دهنده‌های مشغول، تعداد مشتریان منتظر
قطار سریع‌السیر	مسافران	مبدأ، مقصد	سفر	ورود به ایستگاه، رسیدن به مقصد	تعداد مسافران منتظر در هر ایستگاه، تعداد مسافران در سفر
تولید	ماشینها	سرعت، ظرفیت، آهنگ از کارماندگی	جوشکاری، برش	ازکارماندگی	وضعیت ماشینها (مشغول، بیکار، یا از کارمانده)
ارتباطات	پیامها	طول، مقصد	مخابره	ورود به مقصد	تعداد پیامهای در انتظار مخابره
موجودی	انبار	ظرفیت	خارج‌سازی کالا از انبار	تقاضا	سطوح موجودی، تقاضاهای پس‌افت

تا هنگامی که مقصود از بررسی معلوم نشود، نمی‌توان فهرستی کامل پدید آورد. بسته به مقصود بررسی، جنبه‌های گوناگون سیستم مورد توجه قرار می‌گیرد و آن‌گاه ممکن است بتوان فهرست اجزاء را کامل کرد.

۱-۶ سیستمهای گسسته و پیوسته

سیستمها را می‌توان در دو ردهٔ گسسته و پیوسته جا داد. «سیستمهای انگشت‌شماری در عمل به‌طور کامل گسسته یا پیوسته‌اند، اما چون یک نوع تغییر در اکثر سیستمها نقش مسلط دارد، معمولاً امکان رده‌بندی سیستمها در دو ردهٔ گسسته یا پیوسته فراهم است» [لا و کلتون، ۱۹۸۲]. (سیستم گسسته سیستمی است که متغیر(های) حالت در آن تنها در مجموعه‌ای از نقاط گسستهٔ زمان تغییر کند. بانک، مثالی در مورد سیستم گسسته است زیرا متغیر حالت تعداد مشتری حاضر در بانک، تنها وقتی تغییر می‌کند که یک مشتری وارد یا خدمتدهی به یک مشتری کامل شود. شکل ۱-۱ چگونگی تغییر مشتریان را تنها در مقاطع گسسته‌ای از زمان نشان می‌دهد.

(سیستم پیوسته سیستمی است که متغیر(های) حالت در آن به‌صورت پیوسته طی زمان تغییر کند. یک مثال، مربوط به تارک آب پشت سد است. در جریان بارش هر رگبار و تا مدتی پس از آن، آب در دریاچهٔ پشت سد جریان می‌یابد. از سوی دیگر، به منظور مهار سیلاب و تولید برق، آب سد تخلیه می‌شود. تبخیر نیز سطح آب را کاهش می‌دهد. شکل ۱-۲ نشان می‌دهد که چگونه متغیر حالت تارک آب پشت سد در مورد این سیستم پیوسته تغییر می‌کند.



شکل ۱-۱ متغیر حالت در سیستم گسسته.

شکل ۱-۲ متغیر حالت در سیستم پیوسته.

۱-۷ مدل سیستم

گاهی به بررسی سیستم به این منظور رو می‌کنیم که به روابط بین اجزای آن پی‌بریم یا چگونگی عمل آن را در شرایط به‌کارگیری یک خط‌مشی تازه پیش‌بینی کنیم. گاهی برای بررسی سیستم، تجربه در مورد خود سیستم امکانپذیر است. اما این امکان همیشه فراهم نیست. ممکن است هنوز سیستم جدیدی وجود نداشته و آن سیستم به‌صورت فرضی یا در مرحلهٔ طراحی موجود باشد. حتی اگر سیستم موجود باشد، ممکن است انجام تجربه در مورد آن عملی نباشد. مثلاً، دو برابر کردن آهنگ بیکاری به منظور تعیین اثر اشتغال بر تورم ممکن است معقول یا میسر نباشد. در مورد بانک، کاستن از تعداد خدمت‌دهندگان باجه به منظور بررسی اثر آن بر طول صف انتظار ممکن است مشتریان را چنان خشمگین سازد که حسابهای خود را به بانک رقیب منتقل کنند. بنابراین، بررسی سیستمها اغلب با مدلی از سیستم انجام می‌شود.

مدل به منزلهٔ معرّف هر سیستم است که به منظور بررسی آن تعریف می‌شود. در اکثر بررسیها، در نظر گرفتن همهٔ جزئیات سیستم لازم نیست؛ بدین ترتیب، مدل نه تنها جانشینی برای سیستم

www.matlabdl.com

است، بلکه ساده‌سازی سیستم نیز هست [مایرم و مایرم ۱۹۷۴]. از سوی دیگر، مدل باید به اندازهٔ کافی دربردارندهٔ جزئیات باشد تا اجازه دهد نتیجه‌هایی معتبر در مورد سیستم حقیقی گرفته شود. به سبب تغییر هدف تحقیق در سیستم، ممکن است به مدلهای متفاوتی نیاز باشد.

درست به همان‌گونه که نهادها، خصیصه‌ها، و فعالیتها اجزای سیستم‌اند، مدلها را نیز به گونه‌ای همانند معرفی می‌کنند. اما مدل تنها اجزای مربوط به بررسی را در بر می‌گیرد. دربارهٔ اجزای مدل با تفصیل بیشتری در فصل ۳ بحث خواهیم کرد.

۱-۸ هنر مدلسازی

فرایندی را که طبق آن مهندسان و مدیران برای سیستمهای تحت بررسی خود به مدلسازی می‌پردازند باید یک هنر ابتکاری قلمداد کرد. هر مجموعه از قوانین مدلسازی تنها در چارچوب خاصی قابل اعمال است. متأسفانه، تمام تحقیقات علمی با ساختار منطقی پیشامدها گزارش می‌شود و سعی در توجیه نتایج به‌دست آمده دارد. ساختار منطقی مورد اشاره، حتی اگر به نحوهٔ انجام تحقیق بستگی هم داشته باشد، این بستگی ناچیز است. در واقع، هیچ یک از گزارشهای علمی، پایه‌های شروع کج و غلط، فرضیه‌های اشتباه، ناراحتی ناشی از به نتیجه نرسیدن کوششها و . . . را منعکس نمی‌کند. صرفاً پس از حصول نتایج است که مقالات علمی به گزارش این مطلب می‌پردازد که مسأله چیست و تحلیلگر چگونه درصدد حل مسأله برآمده است. به این ترتیب، برای مدلساز بی‌تجربه خطری بزرگتر از باور داشتن ساختار منطقی گزارشهای فوق نیست. چون او چنین خواهد پنداشت که تنها راه کشف راه‌حلها، ساختار مزبور است و وقتی در عمل با پیشرفت کند کارها مواجه شود نومید و دلسرد خواهد شد. یک مدلساز باتجربه به این نکته واقف است که فرایند فکری و روحی ساختن یک مدل، از آنچه که در گزارشهای علمی به چشم می‌خورد بسیار متفاوت است.

روش صحیح مدلسازی چنین است که با مدلی بسیار ساده کار را شروع کنیم و به‌تدریج به کامل کردن آن بپردازیم. مسائل واقعی بسیار پیچیده‌تر از آن است که کاملاً آن را درک و توصیف کنیم. هر مسأله، معمولاً از تعداد بیشماری متغیر، پارامتر، محدودیت، جزء و رابطه تشکیل می‌شود. به هنگام مدلسازی می‌توان سعی در به‌کارگیری تعداد زیادی از واقعیات کرد و زمانی بسیار طولانی را صرف گردآوری داده‌ها و شناخت روابط کرد. مثلاً، عمل سادهٔ نوشتن یک نامه را در نظر بگیرید. می‌توان ترکیب شیمیایی کاغذ، جوهر و پاک‌کن را به تفصیل مطالعه کرد، یا آثار شرایط هوا بر رطوبت موجود در کاغذ و تأثیر آن بر اصطکاک نوک قلم بر کاغذ را مورد بررسی قرار داد، یا به توزیع احتمال حروف در جمله‌های نامه توجه کرد. اما، اگر تنها دلیل بررسی مسألهٔ فوق این باشد که آیا نامه فرستاده می‌شود یا نه، هیچکدام از جزئیات فوق ربطی به اصل موضوع پیدا نمی‌کند. پس، به هنگام بررسی هر مسأله باید اغلب خصوصیات واقعی مربوط به آن را ندیده گرفت و فقط آن دسته از خصوصیات را که مستقیماً به هدف بررسی مسأله ربط پیدا می‌کند به‌صورتی انتزاعی در بررسی شرکت داد. هر مدل حالت ساده شده و انتزاعی یک مسألهٔ واقعی است. در صورتی که

انتزاعی کردن مسأله به‌طرز صحیحی صورت گیرد تقریب مفیدی از مسألهٔ واقعی، یا دست کم، از بخشی از آن عاید می‌شود.

به منظور ایجاد مدلی مفید باید از یک فرایند دو مرحله‌ای تجزیه و ترکیب استفاده کرد. منظور از تجزیه، ساده کردن سیستم از راه حذف جزئیات یا از طریق پذیرش فرضهایی است که روابط حاکم بر عوامل را مهارپذیر می‌کند. مثلاً، می‌توان رابطهٔ موجود بین دو متغیر را خطی فرض کرد حتی اگر نشانه‌هایی دال بر غیرخطی بودن آن در دست باشد. بنابراین، مهندس برق با مدلی کار می‌کند که مقادیر مقاومتها و خازنها در آن ثابت فرض می‌شود. چنین فرضی جز ساده کردن مدل نیست زیرا خصوصیات برقی اجزاء فوق توابعی از رطوبت، دما، عمر و . . . است. یک مهندس مکانیک نیز با مدلهایی کار می‌کند که در آنها مثلاً، گازها کامل فرض می‌شود یا رسانایی به‌صورت یکنواخت در نظر گرفته می‌شود. این نوع ساده کردن در اکثر موارد کاربردی قابل قبول شمرده می‌شود زیرا نتایج به‌دست آمده از مدلهای ساده شده هنوز قابل استفاده است.

در مدیریت نیز عمل ساده کردن به منظور ایجاد مدلهای مفید کاربرد دارد. مثلاً، مدیر می‌تواند طبیعت متغیرهای احتمالی را غیراحتمالی فرض کند یا تابع توزیع احتمال متغیرهای تصادفی را کاملاً شناخته شده در نظر بگیرد. عمل ساده کردن مدل معمولاً به یکی از راههای زیر انجام می‌شود:

- تبدیل متغیرها به مقادیر ثابت
- حذف متغیرها یا ادغام آنها در یکدیگر
- فرض خطی بودن روابط
- افزودن محدودیتهای بیشتر
- تحدید حدود سیستم

عمل ساده کردن مدل را تا جایی می‌توان ادامه داد که مدل از لحاظ ریاضی قابل حل شود. از این مرحله به بعد، عمل کامل کردن مدل شروع می‌شود. طبیعت تکاملی مدلسازی امری اجتناب‌ناپذیر است. در واقع، با حل شدن مسألهٔ در دست بررسی، مسائل تازه‌ای پیدا می‌شود یا درجهٔ بالاتری از واقعیت مطلوبیت می‌یابد. پیدایش مسائل تازه و مطلوبیت یافتن شرایط جدید به اصلاح مدل و تهیهٔ راه‌حلهای بهتر می‌انجامد. فرایند ایجاد مدلی ساده و کامل‌کردن آن اثرات مثبتی نیز از نظر کاربرد دارد. در واقع، سرعت و مسیر تکامل به دو عامل اصلی وابسته است. اولین عامل، انعطاف‌پذیری ذاتی مدل و عامل دوم نیز رابطهٔ بین مدلساز و کاربر است. اگر مدلساز و به‌کارگیرندهٔ مدل همکاری نزدیک داشته باشند، ماحصل کوششهای آنها، یعنی مدل، از کیفیت مناسبی برای تأمین اهداف و معیارهای مسأله برخوردار خواهد بود.

افراد با استعدادی از هنر و شیوهٔ مدلسازی برخوردارند که از قوهٔ ابتکار و تجربه‌های قابل توجه در زمینهٔ بررسی و مطالعهٔ سیستمها برخوردار باشند.

۱-۹ انواع مدلها

مدلها را می‌توان در مدلهای ریاضی یا فیزیکی رده‌بندی کرد. مدل ریاضی در معرفی سیستم از نمادها و معادله‌های ریاضی استفاده می‌کند. مدل شبیه‌سازی، نوعی خاص از مدل ریاضی سیستم است.

علاوه بر این، مدلهای شبیه‌سازی را می‌توان در مدلهای ایستا یا پویا، قطعی یا تصادفی و گسسته یا پیوسته رده‌بندی کرد. مدل ایستای شبیه‌سازی که گاهی شبیه‌سازی مونت کارلو نامیده می‌شود، معرف سیستم در لحظه‌ای خاص از زمان است. مدلهای پویای شبیه‌سازی، سیستمها را با توجه به تغییرشان با گذشت زمان معرفی می‌کنند. شبیه‌سازی بانک از ۹:۰۰ صبح تا ۴:۰۰ بعدازظهر مثالی از شبیه‌سازی پویاست.

مدلهای شبیه‌سازی بدون هرگونه متغیر تصادفی را در ردهٔ مدلهای قطعی قرار می‌دهند. مدلهای قطعی مجموعهٔ مشخصی از ورودیها دارند که به مجموعه‌ای یگانه از خروجیها می‌انجامد. ورودیهای مطب یک دندانپزشک به‌صورت قطعی رخ می‌دهد اگر تمام بیماران در زمانهای از پیش تعیین شده وارد شوند. مدل تصادفی شبیه‌سازی یک یا چند متغیر تصادفی را به منزلهٔ ورودی در بر دارد. ورودیهای تصادفی به خروجیهای تصادفی می‌انجامد. چون خروجیها تصادفی‌اند، تنها می‌توان آنها را براوردهایی از ویژگیهای واقعی سیستم به‌شمار آورد. شبیه‌سازی بانک معمولاً همراه با مدتهای تصادفی بین دو ورود و مدتهای تصادفی خدمتدهی است. بنابراین، در شبیه‌سازی تصادفی، معیارهای خروجی ــ مانند متوسط تعداد افراد منتظر، متوسط مدت انتظار هر مشتری ــ را باید براوردهایی آماری از ویژگیهای واقعی سیستم تلقی کرد.

سیستمهای گسسته و پیوسته را در بخش ۱-۶ تعریف کردیم. مدلهای گسسته و پیوسته نیز همین‌طور تعریف می‌شوند. اما مدل گسستهٔ شبیه‌سازی را همواره برای مدلسازی سیستم گسسته به‌کار نمی‌برند، همان‌طور که از مدل پیوستهٔ شبیه‌سازی نیز همیشه برای مدلسازی سیستم پیوسته استفاده نمی‌کنند. به‌علاوه، مدلهای شبیه‌سازی ممکن است آمیخته، یعنی هم گسسته و هم پیوسته باشند. انتخاب به‌کارگیری مدل گسسته یا پیوستهٔ (یا آمیختهٔ) شبیه‌سازی، تابعی از ویژگیهای سیستم و هدف بررسی است. بدین ترتیب، یک کانال ارتباطات را در صورتی که به ویژگیها و حرکت هر پیام پر بها داده شود می‌توان به‌صورت گسسته مدلسازی کرد. به طریق وارون، اگر جریان تجمیعی پیامها در کانال مهم شمرده شود، مدلسازی سیستم با استفاده از شبیه‌سازی پیوسته ممکن است مناسبتر باشد. مدلهای گسسته، پویا، و تصادفی را در این کتاب بررسی می‌کنیم.

۱-۱۰ شبیه‌سازی سیستمهای گسسته‌ـپیشامد

این کتاب دربارهٔ شبیه‌سازی سیستم مبتنی بر پیشامدهای گسسته است. شبیه‌سازی سیستمهای گسسته پیشامد عبارت است از مدلسازی سیستمهایی که متغیر حالت در آنها تنها در مجموعه‌ای از مقاطع گسستهٔ زمان تغییر می‌کند. مدلهای شبیه‌سازی را با روشهای عددی تجزیه و تحلیل می‌کنند نه با روشهای تحلیلی. روشهای تحلیلی برای «حل کردن» مدل، منطق استقرایی ریاضی

را به‌کار می‌گیرند. مثلاً، به منظور تعیین خط‌مشی کمترین هزینه در مورد برخی مدلهای موجودی می‌توان از حساب دیفرانسیل استفاده کرد. روشهای عددی در «حل» مدلهای ریاضی از شیوه‌های محاسباتی استفاده می‌کنند. در مورد مدلهای شبیه‌سازی که روشهای عددی را به‌کار می‌گیرند، مدلها «اجرا» می‌شوند و نه حل؛ یعنی بر اساس فرضهای مدل، سابقه‌ای ساختگی از سیستم ایجاد و به منظور براورد معیارهای عملکرد سیستم واقعی، مشاهدات گردآوری و تجزیه و تحلیل می‌شوند. چون مدلهای شبیه‌سازی مسائل واقعی نسبتاً بزرگ‌اند و مقدار داده‌هایی که لازم است ذخیره‌سازی و پردازش شوند چشمگیر است، معمولاً اجراها به کمک کامپیوتر صورت می‌گیرد. اما از راه شبیه‌سازی دستی مدلهای کوچک می‌توان آگاهی قابل‌توجهی به‌دست آورد. خلاصه، این کتاب دربارهٔ شبیه‌سازی سیستمهای مبتنی بر پیشامدهای گسسته است که مدلهای مورد توجه آن از طریق عددی و معمولاً به کمک کامپیوتر تحلیل می‌شوند.

این کتاب به شبیه‌سازی سیستمهای گسسته پیشامد می‌پردازد و مثال زیر تنها مثال در زمینهٔ شبیه‌سازی پیوسته است که در آن گنجانیده‌ایم.

■ **مثال ۱-۱ بررسی ارتباط موجود بین دو جمعیت آکل و مأکول (مثالی از شبیه‌سازی پیوسته)**

مدلهای آکل و مأکول (یا میهمان و میزبان) در زیست‌شناسی توسط نویسندگان زیادی مورد بررسی قرار گرفته است. محیطی را در نظر بگیرید که از جمعیت آکل و جمعیت مأکول تشکیل می‌شود و این دو جمعیت با یکدیگر ارتباط دارد. طبیعت ارتباط مورد بحث چنین است که جمعیت مأکول منبع غذایی جمعیت آکل شمرده می‌شود. برای مثال، جمعیت آکل ممکن است از کوسه‌ها تشکیل شود و ماهیان کوچک محیط نیز جمعیت مأکول را به‌وجود آورند. چنین فرض کنید که تعداد جمعیت آکل و تعداد جمعیت مأکول در لحظهٔ t، به‌ترتیب، با $y(t)$ و $x(t)$ نمادگذاری شود. علاوه بر این، فرض کنید که جمعیت مأکول از منبع غذایی کافی برخوردار است و در صورت عدم وجود جمعیت آکل می‌تواند با آهنگ رشد $rx(t)$ توسعه یابد ($r > 0$). می‌توان r را به‌صورت مابه‌التفاوت دو آهنگ زادومیر طبیعی تعبیر کرد. چون دو جمعیت آکل و مأکول با هم در ارتباط‌اند، منطقی است اگر فرض کنیم که آهنگ مرگ‌ومیر ناشی از وجود چنین ارتباطی برای جمعیت مأکول با حاصلضرب اندازهٔ دو جمعیت، یعنی $x(t)y(t)$ نسبت مستقیم دارد. بدین ترتیب، اگر a ضریب ثابت و مثبتی باشد، آهنگ کلی تغییر در جمعیت مأکول، dx/dt، به طریق زیر تعریف می‌شود:

$$dx/dt = rx(t) - ax(t)y(t).$$

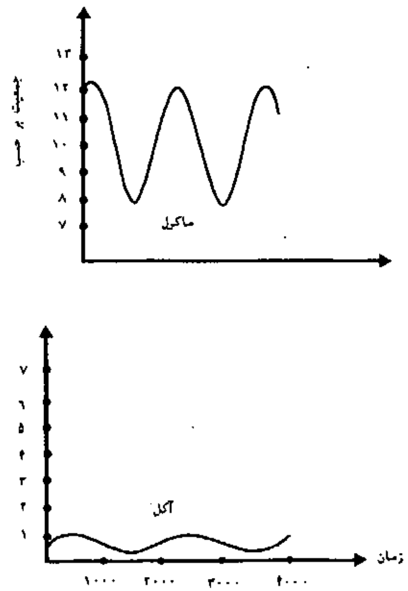
چون جمعیت آکل برای بقای خود به جمعیت مأکول متکی است، در صورت عدم وجود جمعیت مأکول، آهنگ تغییر جمعیت آکل $-sy(t)$ می‌شود ($s > 0$). به‌علاوه، ارتباط موجود بین دو

جمعیت باعث می‌شود که آهنگ افزایش جمعیت آکل نیز با $x(t)y(t)$ نسبت مستقیم داشته باشد. بنابراین، آهنگ کلی تغییر در جمعیت آکل نیز به ازای $b > ۰$ به شرح زیر تعریف می‌شود:

$$dy/dt = -sy(t) + bx(t)y(t)$$

اگر شرایط شروع به‌صورت $x(۰) > ۰$ و $y(۰) > ۰$ تعریف شود، نتیجهٔ حل مدل متشکل از دو معادلهٔ بالا ناظر به صدق روابط $x(t) > ۰$ و $y(t) > ۰$ به ازای همهٔ مقادیر t خواهد بود. به این ترتیب، جمعیت مأکول هیچ‌گاه توسط جمعیت آکل منقرض نمی‌شود. نتیجه که در قالب مجموعهٔ $\{x(t), y(t)\}$ مشخص می‌شود، تابعی متناوب از زمان است. به بیان دیگر، مثبتی مانند T وجود دارد به‌طوری که به ازای مقادیر ۱، ۲، ... برای n، روابط $x(t + nT) = x(t)$ و $y(t + nT) = y(t)$ برقرار است. حصول چنین نتیجه‌ای نامنتظره نیست. هرگاه جمعیت آکل رو به افزایش گذارد، جمعیت مأکول رو به کاهش می‌گذارد. کاهش یافتن جمعیت مأکول باعث کند شدن آهنگ افزایش جمعیت آکل می‌شود و این به نوبهٔ خود جمعیت آکل را کاهش می‌دهد و جمعیت مأکول را بالا می‌برد.

حل عددی دو معادلهٔ بالا به ازای ورودیهای $r = ۰/۰۰۱$، $a = ۲ \times ۱۰^{-۶}$، $s = ۰/۰۱$، $b = ۱۰^{-۶}$، $x(۰) = ۱۲۰۰۰$، و $y(۰) = ۶۰۰$ در شکل ۱-۳ نشان داده شده است. ■



شکل ۱-۳ حل عددی مدل آکل و مأکول.

گهگاه با مسائلی درگیر می‌شویم که به‌طور کامل گسسته یا به‌طور کامل پیوسته نیست. در مورد این نوع مسائل باید مدلهایی طراحی کرد که در برگیرندهٔ برخی از جنبه‌های شبیه‌سازی گسسته و شبیه‌سازی پیوسته باشد. این نوع شبیه‌سازی را شبیه‌سازی آمیخته می‌نامیم.

۱-۱۱ جاذبه‌های شبیه‌سازی به‌عنوان ابزار تجزیه و تحلیل مسأله

شبیه‌سازی سیستمهای گسسته پیشامد با کامپیوتر و یا به‌طور خلاصه شبیه‌سازی کامپیوتری، خصوصیاتی دارد که آن را از دید تحلیلگران به‌صورت ابزار جالبی در می‌آورد. با شبیه‌سازی کامپیوتری می‌توان زمان را فشرده کرد به‌نحوی که فعالیتهای چند سال در ظرف چند دقیقه و گاهی در ظرف چند ثانیه شبیه‌سازی شود. با استفاده از این امتیاز، تحلیلگر می‌تواند طرحهای متنوعی را با صرف زمان ناچیزی در مورد مسألهٔ واقعی به اجرا گذارد و ارزیابیهایی از آنها به‌دست آورد.

شبیه‌سازی کامپیوتری از عهدهٔ بسط دادن زمان نیز برمی‌آید. در واقع، با تعبیهٔ این امکان که در فواصل زمانی کوتاه در خلال ساعت شبیه‌سازی، داده‌های موردنظر تحلیلگر تولید و چاپ شود، شناخت قابل‌توجهی از ریزه‌کاریهای تغییرات ساختاری سیستم به‌دست می‌آید که دست یافتن چنین شناختی بر اساس زمان واقعی میسر نیست. در مواردی که شناخت کافی از طبیعت تغییرات درونی سیستم موجود نباشد، ارزش این مزیت بهتر آشکار می‌شود.

از جمله ملاحظات اساسی در انجام هر تجربه، امکان تشخیص و مهار کردن منابع تغییر (پراکندگی) است. اهمیت چنین امکانی به خصوص در مواردی آشکار می‌شود که هدف تحلیل آماری رابطهٔ بین عوامل مستقل (ورودی) و وابسته (خروجی) پی گرفته شود. امکان تشخیص و مهار کردن منابع تغییر در دنیای واقعی، اساساً تابعی از سیستم در دست بررسی است. به هنگام استفاده از شبیه‌سازی کامپیوتری، تحلیلگر ناچار است که به منظور اجرای مدل خود، مشخصاً به تعیین منابع تغییر و میزان تأثیر هر منبع بپردازد. چنین نیازی او را قادر می‌کند که منابع ناخواستهٔ تغییرات را از حیطهٔ بررسی حذف کند. در عین حال، این امکان تحلیلگر را ملزم می‌کند تا با بذل توجه کافی به سیستم، شناخت مناسبی در زمینهٔ تشریح کمّی منابع تغییرات ورودی که از حیطهٔ بررسی حذف نشده‌اند کسب کند.

به‌هنگام ثبت نتایج یک آزمایش واقعی (غیر انتزاعی)، ارتکاب خطای اندازه‌گیری اجتناب‌ناپذیر است. دلیل چنین امری این واقعیت است که هیچ ابزار سنجش کامل و بدون خطایی برای ثبت نتایج آزمایشهای فیزیکی وجود ندارد. از طرف دیگر، امکان ارتکاب خطای اندازه‌گیری در شبیه‌سازی کامپیوتری وجود ندارد زیرا مدل شبیه‌سازی (یعنی برنامهٔ کامپیوتری) اعدادی تولید می‌کند که از تأثیر تغییرات ناشی از دخالت عوامل خارجی و غیر قابل کنترل مصون است. البته، به‌خاطر محدود بودن طول کلمهٔ یک کامپیوتر، امکان ایجاد بی‌دقتی ناشی از گرد کردن مقادیر عددی وجود دارد ولی با بذل دقت ناچیزی از جانب تحلیلگر، می‌توان این منبع تغییر را چنان مهار کرد که به‌صورت قابل گذشت درآید. به‌طور مشخص با استفاده از کلمات با طول مضاعف می‌توان بی‌دقتی مورد

www.matlabdl.com

بحث را مهار کرد.

در جریان برگزاری یک آزمایش، گهگاه این نیاز مطرح می‌شود که آزمایش موقتاً متوقف شود تا نتایج به‌دست آمده تا لحظهٔ قطع آزمایش بررسی شود. این نیاز ایجاب می‌کند که همهٔ پدیده‌های درگیر در آزمایش، وضعیت فعلی خود را تا لحظه‌ای که ادامهٔ آزمایش شروع می‌شود حفظ کنند. در آزمایشهای واقعی به‌ندرت می‌توان همهٔ فعل و انفعالات را کاملاً متوقف کرد. شبیه‌سازی کامپیوتری از این امتیاز برخوردار است و به منظور استفاده از آن باید در بخش متوقف کنندهٔ برنامه دستورالعملهایی را در نظر گرفت که وضعیت حاکم بر مدل را به‌طور کامل ثبت کند. با شروع مجدد برنامه، شرایط لحظهٔ قطع برنامه، شرایط شروع تازه را تشکیل می‌دهد و تداوم اجرای برنامه دچار اختلال نمی‌شود.

امتیاز دیگر شبیه‌سازی کامپیوتری، قابلیت اجرای مدل به‌طور مکرر و تحت شرایط شروع یکسان است. در پایان هر اجرای شبیه‌سازی کامپیوتری، تحلیل نتایج ممکن است حاکی از این مطلب باشد که اگر داده‌های بیشتری گردآوری می‌شد پاسخگویی به برخی پرسشها ممکن بود آسانتر باشد. در چنین شرایطی، می‌توان با افزودن جملات بیشتر به برنامهٔ کامپیوتری، داده‌های مورد نیاز را تولید کرد به‌نحوی که شرایط شروع اجرای تازه کاملاً مانند شرایط شروع در اجرای پیشین باشد. عملکرد مدل در این دو اجرا همانند است با این تفاوت که در اجرای دوم داده‌های بیشتری گردآوری و چاپ می‌شود. چون اجرای مجدد مدل به مضاعف شدن وقت مورد نیاز اجرای کامپیوتری می‌انجامد، باید پیش از اجرای مدل به تعیین داده‌های لازم پرداخت تا در هزینه‌ها صرفه‌جویی شود.

با شبیه‌سازی کامپیوتری به دوباره‌سازی (و نه تکرار) یک آزمایش نیز توانا می‌شویم. منظور از دوباره‌سازی اجرای مجدد آزمایش با ایجاد تغییرات مورد نظر در پارامترهای مربوط به شرایط عملکرد آن است. مثلاً، منظور از یک دوباره‌سازی مستقل این است که بدون ایجاد کوچکترین تغییری در مدل شبیه‌سازی، مجدداً آن را اجرا کنیم به‌طوری که دنبالهٔ اعداد تصادفی به‌کار رفته در آن از دنبالهٔ اعداد تصادفی مصرف شده در اجرای اول مستقل باشد.

همچنانکه قبلاً توضیح دادیم، هرگاه نتوان با استفاده از روشهای تحلیلی راه‌حلی برای یک مسأله ارائه داد، شبیه‌سازی کامپیوتری را می‌توان به‌طور جدی به‌عنوان ابزار تحقیق مورد بررسی قرارداد. اگر قرار شود از شبیه‌سازی کامپیوتری به منظور تحلیل مسأله استفاده شود می‌بایست به برخی از ویژگیهای مدلسازی مانند ساده کردن مسأله نگاهی دوباره کرد.

به موجب مطالبی که قبلاً عرضه شد، هر چه جزئیات بیشتری در ایجاد مدل شرکت داده شود امکان حصول راه‌حل دقیق (تحلیلی) کمتر می‌شود. معمول این است که به منظور مهار مسأله و طراحی مدل برای آن، اقدام به ساده کردن مدل می‌کنند. عمل ساده کردن تا جایی ادامه می‌یابد که بررسی مدل ساده‌شده هنوز مفید باشد و ساده کردن بیشتر آن مفید تشخیص داده نشود. مدلی در این حد ساده شده را مختصرترین مدل می‌نامند. اگر نتوان مختصرترین مدل را از راه تحلیلی حل کرد و استفاده از شبیه‌سازی کامپیوتری برای تجزیه و تحلیل آن در نظر گرفته شود، می‌توان بیشترین جزئیات را در طراحی مدل مسأله شرکت داد. گیراترین امتیاز شبیه‌سازی برای طراحان مدل جز

این نیست که می‌توان بیشترین عوامل اعم از عمده و جزئی را در مدل شبیه‌سازی دخالت داد تا مدلی واقعی‌تر طراحی شود و بدان سبب نتایجی تولید شود که انطباق نزدیکتری با واقعیت داشته باشد. در هر حال، به هنگام افزودن عوامل جزئی به مدل باید جانب احتیاط را رعایت کرد. زیرا دخالت دادن جزئیات فراوان در مدل ناظر به صرف زمان و منابع مالی بیشتر در زمینهٔ شناخت سیستم، طراحی مدل (برنامهٔ کامپیوتری) و اجرای آزمایشی و نهایی مدل است.

۱-۱۲ مونت‌کارلو و شبیه‌سازی

تعاریف متعددی برای روش مونت‌کارلو ارائه شده است. در این کتاب، روش مونت‌کارلو یا شبیه‌سازی مونت‌کارلو به طریق زیر تعریف می‌شود:

> مونت‌کارلو روشی است که به منظور حل کردن مسائل غیرتصادفی یا برخی مسائل تصادفی که گذشت زمان هیچ نقش اساسی در آنها ندارد از اعداد تصادفی استفاده می‌کند.

منظور از اعداد تصادفی در تعریف بالا متغیرهای تصادفی مستقل با توزیع آماری یکنواخت در محدودهٔ [۰ , ۱] است. بر اساس تعریف فوق، مونت‌کارلو روشی ایستا و نه پویا شمرده می‌شود. نام مونت‌کارلو در خلال جنگ دوم جهانی به‌عنوان اسم رمز به این روش داده شد؛ سالهایی که روش مورد بحث در زمینهٔ حل مسائلی که مربوط به تولید بمب اتمی می‌شد مورد استفاده قرار می‌گرفت. در مقابل روش مونت‌کارلو می‌توان روش شبیه‌سازی را قرار داد. گرچه در شبیه‌سازی نیز مانند مونت‌کارلو از اعداد تصادفی استفاده می‌شود ولی تشابه دو روش در همین‌جا به پایان می‌رسد. در واقع، عامل زمان در شبیه‌سازی دخالت دارد و به بیان دیگر، شبیه‌سازی روشی پویا محسوب می‌شود. علاوه بر این، اکثر مسائلی که شبیه‌سازی می‌شود طبیعتی تصادفی دارد. چون در فصلهای بعد به تفصیل در مورد شبیه‌سازی مطالبی ارائه شده است، در سطور زیر به‌اختصار توضیحاتی در زمینهٔ موارد استفاده از مونت‌کارلو عرضه می‌کنیم.

(الف) یک مورد کاربرد مونت‌کارلو مربوط به حل مسائل غیرتصادفی با استفاده از اعداد تصادفی است. به‌عنوان مثال، فرض کنید که قصد براورد $I = \int_a^b g(x)dx$ را داریم که انتگرال تابع حقیقی $g(x)$ را نمی‌توان از طریق تحلیلی پیدا کرد. برای اینکه با استفاده از روش مونت‌کارلو این مسألهٔ غیرتصادفی را حل کنیم متغیر تصادفی Y را به‌صورت $Y = (b-a)g(X)$ تعریف می‌کنیم، که X یک متغیر تصادفی یکنواخت در محدودهٔ $[a,b]$ است (یعنی $X \sim U[a,b]$). بدین ترتیب، می‌توان نشان داد که امید ریاضی Y به شرح زیر به‌دست می‌آید:

$$
\begin{aligned}
E(Y) = E[(b-a)g(X)] &= (b-a)E[g(X)] \\
&= (b-a)\int_a^b g(x)f_X(x)dx \\
&= (b-a)[\int_a^b g(x)dx/(b-a)] = I
\end{aligned}
$$

در رابطهٔ فوق، تابع چگالی متغیر تصادفی X با $f_X(x)$ نمادگذاری شده و به‌صورت $f_X(x) = \frac{1}{b-a}$ تعریف می‌شود. بدین ترتیب، مسألهٔ براورد انتگرال به مسألهٔ تقریب‌زدن امید ریاضی Y تبدیل شده است. به منظور یافتن تقریبی برای $E[Y] = I$ از میانگین نمونه، یعنی

$$\bar{Y}(n) = \sum_{i=1}^{n} Y_i/n = (b-a)\sum_{i=1}^{n} g(X_i)/n$$

استفاده می‌کنیم به‌طوری که X_1، X_2، ...، X_n مستقل و دارای توزیع آماری یکنواخت در محدودهٔ $[a,b]$ باشند.

به منظور درک بهتر منطق مثال فوق چنین استدلال می‌کنیم که $\bar{Y}(n)$ متوسط مساحت n مستطیل است که پایهٔ همهٔ آنها معادل $(b-a)$ و ارتفاع هر یک از آنها معادل $g(X_i)$ است $(i = 1, \ldots, n)$. علاوه بر این، می‌توان نشان داد که روابط $E[\bar{Y}(n)] = I$ و $Var[\bar{Y}(n)] = Var(Y)/n$ برقرار است. چون $Var(Y)$ ثابت است، با بزرگ شدن n می‌توان به‌طور دلخواه $\bar{Y}(n)$ را به I نزدیک کرد. در زمینهٔ چگونگی تولید مقادیر X_i توضیحات کافی در فصل ۸ ارائه خواهد شد.

چون روشهای کاراتری برای تقریب زدن انتگرالهای ساده وجود دارد، احتمال کاربرد روش مونت‌کارلو در مورد مثالی از نوع بالا چندان زیاد نیست. در واقع، اگر انتگرال مضاعف و تابع $g(x)$ نیز تابعی پیچیده باشد، انتخاب مونت‌کارلو به‌عنوان روش تحلیل معقولتر است.

ب) مورد دیگر کاربرد مونت‌کارلو نمونه‌گیری از توزیعهای آماری مجهول است. نمونه‌گیری از توزیعهای آماری مجهول به مدت چندین دهه در مبحث آمار ریاضی مورد استفاده قرار داشته‌است. هدف از این نمونه‌گیری یافتن توزیع آماری هر متغیر تصادفی یا یک (یا چند) پارامتر آن است. متغیر تصادفی مورد بحث را متغیر پاسخ می‌نامیم. متغیر پاسخ تابعی از یک یا چند متغیر تصادفی شناخته شده است. به منظور ارائهٔ براوردی برای توزیع آماری متغیر پاسخ، مقادیری برای همهٔ متغیرهای تصادفی ورودی تولید می‌کنیم و مقدار نظیر از متغیر پاسخ را بر اساس آنها محاسبه می‌کنیم. این طرز نمونه‌گیری را آن‌قدر تکرار می‌کنیم که براوردی از توزیع آماری متغیر تصادفی ایجاد شود. مثالی از این مورد کاربرد مونت‌کارلو در اواخر فصل ۲ عرضه می‌شود. این مثال مربوط به براورد تابع توزیع تقاضا در اثنای مهلت تحویل در یک مسألهٔ کنترل موجودی است.

مثالهای دیگر نمونه‌گیری از توزیعهای آماری مجهول مربوط به بررسیهای فراوانی می‌شود که در زمینهٔ انسجام آماره‌ها انجام می‌گیرند. اگر توزیع آماری یک آماره نسبت به نقض فرضیات شکل‌دهندهٔ خود حساسیت کمتری نشان دهد آماره را منسجم‌تر به‌شمار می‌آورند. برای مثال، آمارهٔ

$$t = \sqrt{n}(\bar{X} - \mu)/S$$

را در نظر بگیرید. اگر $\bar{X}$ و S بر اساس مشاهدات نرمال X_i تعریف شوند $(i = 1, \ldots, n)$، آمارهٔ فوق توزیع آماری t با $(n-1)$ درجهٔ آزادی خواهد داشت. اگر $\bar{X}$ و S بر اساس مشاهدات

غیرنرمال X_i تعریف شود می‌توان از روش مونت‌کارلو به منظور بررسی توزیع آماری آمارهٔ فوق استفاده کرد. حاصل این‌گونه بررسی این است که با افزایش مقادیر n توزیع آماری آمارهٔ مورد بحث به t با $(n-1)$ درجهٔ آزادی میل می‌کند؛ به‌عبارت دیگر، آمارهٔ فوق منسجم است.

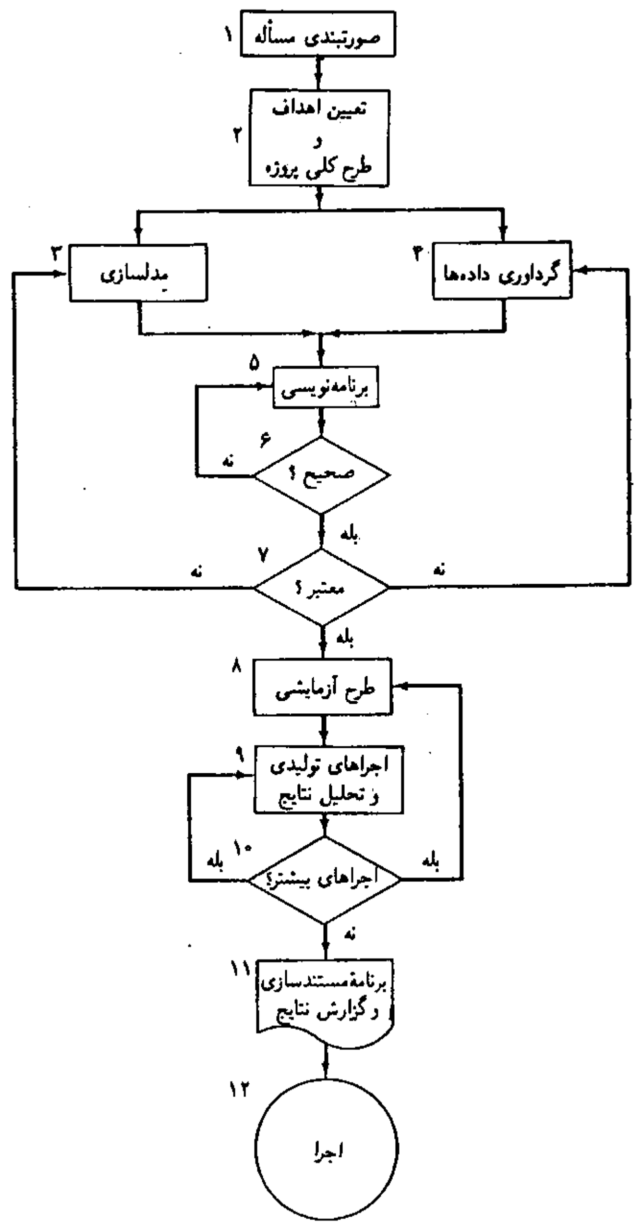
۱-۱۳ گامهای اساسی در بررسی مبتنی بر شبیه‌سازی

شکل ۱-۴ مجموعهٔ گامهایی را نشان می‌دهد که مدلساز را در بررسی مبتنی بر شبیه‌سازی به‌طور کامل و مطمئن هدایت می‌کند. شکلهای همانند همراه با تشریح گامها را می‌توان در منابع دیگر [شنون، ۱۹۷۵؛ گوردون، ۱۹۷۸؛ لا و کلتون، ۱۹۸۲] نیز یافت. عدد جنب هر نشانهٔ شکل ۱-۴ به شرحی مفصلتر در متن اشاره دارد. گامهای اساسی بررسی مبتنی‌بر شبیه‌سازی به‌شرح زیر است:

۱. *صورتبندی مسأله.* هر بررسی مبتنی بر شبیه‌سازی را باید با صورتبندی مسأله شروع کرد. اگر سیاستگذاران یا صاحبان مسأله، آن را به پیش کشند، تحلیلگر باید از درستی درک خود دربارهٔ آن اطمینان حاصل کند. اگر تحلیلگر مسأله را صورتبندی کند، درک صحیح سیاستگذاران از آن و توافق آنها با چگونگی صورتبندی آن مهم شمرده می‌شود. در مواردی، با پیشروی در بررسیها، ارائهٔ صورت دیگری از مسأله لازم می‌شود، هر چند که شکل ۱-۴ چنین امکانی را نشان نمی‌دهد. سیاستگذاران و تحلیلگران در موارد بسیاری از وجود مسائل، بسیار پیشتر از شناخت از ماهیت آن آگاهی دارند!

۲. *تعیین اهداف و طرح کلی پروژه.* اهداف شبیه‌سازی پرسشهایی را مطرح می‌کند که باید پاسخ آنها را با استفاده از شبیه‌سازی به‌دست آورد. در این مورد باید تصمیم گرفت که آیا با توجه به صورتبندی مسأله و اهداف اظهار شده برای آن، شبیه‌سازی روش مناسبی برای تحلیل مسأله شمرده می‌شود یا نه. با قبول این فرض که رأی بر مناسب بودن شبیه‌سازی است، طرح کلی اجرا باید دربردارندهٔ سیستمهای مختلف قابل بررسی و روشی در زمینهٔ ارزیابی کارایی هر یک از آنها باشد. طرح کلی اجرا، از جمله، باید در بردارندهٔ برنامه‌هایی برای بررسی مبتنی بر شبیه‌سازی برحسب تعداد افراد درگیر در بررسی، هزینهٔ بررسی و تعداد روزهای لازم برای اجرای هر گام از کار همراه با نتایج قابل حصول در پایان هر مرحله باشد.

۳. *مدلسازی.* ساختن مدل سیستم را کاری به یک سان هنری و علمی می‌شناسند. شنون [۱۹۷۵] بحثی تفصیلی دربارهٔ این گام ارائه کرده است. «هر چند ارائهٔ مجموعه دستورالعملهایی که در هر مورد به ایجاد مدلهای موفق و مناسب بیانجامد میسر نیست، دستوراتی کلی وجود دارد که می‌توان به آنها عمل کرد» [موریس، ۱۹۶۷]. هنر مدلسازی با استعداد تجرید خصوصیات اساسی مسأله، انتخاب و اصلاح فرضهای اساسی مشخص‌کنندهٔ سیستم، و سپس، غنی‌سازی و کار بر روی مدل تا زمانی که به نتایج تقریبی مناسبی دست یابیم، تقویت می‌شود. بنابراین، مناسبترین شیوه، آغاز کار با مدل ساده و پیچیده کردن تدریجی آن است. اما، پیچیدگی مدل نباید از آن حد که تأمین‌کنندهٔ مقصودهای ایجاد مدل است بیشتر شود. نقض این اصل تنها به افزایش هزینه‌های



شکل ۱-۴ گامهای اساسی در بررسی مبتنی بر شبیه‌سازی.

مدلسازی و هزینه‌های کامپیوتر می‌انجامد. ایجاد یک تناظر یک‌ـبه‌ـیک بین مدل و سیستم حقیقی لازم نیست بلکه تنها دست یافتن به چکیدهٔ سیستم واقعی مورد نیاز است.

توصیه می‌شود که استفاده‌کننده از مدل در ساختن مدل شرکت جوید. شرکت دادن استفاده‌کننده از مدل در این کار، هم کیفیت مدل به‌دست آمده را بالا می‌برد و هم بر اطمینان خاطر استفاده‌کننده از مدل در مرحلهٔ به‌کارگیری آن می‌افزاید. (در فصلهای ۲ و ۳ چند مدل شبیه‌سازی را تشریح می‌کنیم. در فصلهای ۵ و ۶ مدلهایی از صف و موجودی را ارائه می‌دهیم که از راه تحلیلی قابل حل‌اند. به هر صورت، صرفاً با تجربه کردن سیستمهای واقعی‌ـ نه مسائل کتابی‌ـ می‌توان هنر مدلسازی را «آموزش داد».)

۴. گردآوری داده‌ها. بین ساختن مدل و گردآوری داده‌های ورودی مورد نیاز، رابطهٔ متقابل مداومی وجود دارد [شنون، ۱۹۷۵]. همچنانکه پیچیدگی مدل تغییر می‌کند، عناصر داده‌ای مورد نیاز نیز تغییر می‌کنند. به‌علاوه، چون گردآوری داده‌ها بخش بزرگی از مجموع مدت مورد نیاز برای انجام شبیه‌سازی را در بر می‌گیرد، لازم است که آن را تا حد ممکن زود و معمولاً همراه با مراحل اولیهٔ مدلسازی آغاز کرد.

اهداف بررسی تا حدود زیادی نوع داده‌هایی را که باید گردآوری شوند تعیین می‌کند. مثلاً، در بررسی مربوط به بانک اگر قصد آموختن دربارهٔ طول صفهای انتظار به سبب تغییر تعداد خدمت‌دهندگان را داشته باشیم، انواع داده‌های مورد نیاز توزیع مدتهای بین دو ورود (در زمانهای مختلف روز)، توزیع مدتهای خدمتدهی و پیشینهٔ توزیع طول صفهای انتظار در شرایط متفاوت خواهد بود. این آخرین داده‌ها به منظور معتبرسازی مدل شبیه‌سازی مورد استفاده قرار می‌گیرد. (در فصل ۹ در مورد گردآوری داده‌ها و تجزیه و تحلیل آنها بحث می‌کنیم. در فصل ۴ توزیعهای آماری را که اغلب در ساخت مدل شبیه‌سازی مطرح می‌شوند به بحث می‌گذاریم.)

۵. برنامه‌نویسی. چون از اکثر سیستمهای واقعی مدلهایی نتیجه می‌شود که به مقدار معتنابهی ذخیره‌سازی و محاسبات اطلاعاتی نیاز دارند، مدل را باید برای کامپیوتر رقمی برنامه‌نویسی کرد. مدلساز باید تصمیم بگیرد که آیا باید مدل را به یکی از زبانهای عمومی مانند فرترن برنامه‌نویسی کند یا به یکی از زبانهای خاص شبیه‌سازی مانند GPSS، SIMSCRIPT، یا SLAM. زبان عمومی به زمان برنامه‌نویسی بسیار طولانیتری نیاز دارد ولی معمولاً بسیار سریعتر از زبانهای خاص روی کامپیوتر اجرا می‌شود. اما، به‌طورکلی، زبانهای خاص چنان برنامه‌نویسی (و تصحیح برنامه) را تسریع می‌کنند که تعداد مدلسازان استفاده‌کننده از آنها پیوسته روبه افزایش است.

۶. وارسی برنامه. وارسی، مربوط به برنامهٔ کامپیوتری آماده شده برای مدل شبیه‌سازی است. آیا برنامهٔ کامپیوتری به خوبی کار می‌کند؟ در مورد مدلهای پیچیده، برنامه‌نویسی کامل مدل به طریقی موفقیت‌آمیز بدون مقدار قابل‌توجهی غلط‌گیری، امری دشوار است، اگر ناممکن نباشد. اگر پارامترهای ورودی و ساختار منطقی مدل به طریقی صحیح در برنامه وارد شده باشد، وارسی کامل شده است. در کامل کردن این گام، بیش از هر چیز از عقل سلیم استفاده می‌شود. (در فصل ۱۰

به بحث دربارهٔ وارسی مدلهای شبیه‌سازی پرداخته‌ایم.)

۷. *معتبرسازی مدل.* معتبرسازی مشخص کردن این است که آیا مدل معرف دقیقی از سیستم واقعی هست یا نه. معتبرسازی معمولاً از طریق محک زدن مدل انجام می‌گیرد، یعنی فرایند تکرارشونده‌ای که ناظر به مقایسهٔ مدل با رفتار سیستم واقعی، بهره‌برداری از موارد افتراق بین آنها و شناخت به‌دست آمده از این طریق به منظور وارسی مدل است. این فرایند تا جایی تکرار می‌شود که دقت مدل قابل‌قبول تشخیص داده شود. در مثال بانک که در بالا آمد، داده‌های مربوط به طول صف انتظار در شرایط فعلی گردآوری شد. آیا مدل شبیه‌سازی از عهدهٔ دوباره‌سازی این معیار عملکرد سیستم برمی‌آید؟ این، یک وسیلهٔ معتبرسازی است. (در فصل ۱۰ به معتبرسازی مدلهای شبیه‌سازی پرداخته‌ایم.)

۸. *طرح آزمایشی.* گزینه‌هایی را که قرار است شبیه‌سازی شوند باید تعیین کرد. اغلب، تصمیم مربوط به اینکه کدام گزینه‌ها باید شبیه‌سازی شوند ممکن است تابع اجراهایی باشد که کامل و تجزیه و تحلیل شده‌اند. در هر طرح سیستم که شبیه‌سازی می‌شود، باید تصمیمهایی در مورد طول دورهٔ راه‌اندازی، طول مدت اجراهای شبیه‌سازی و تعداد دوباره‌سازیهای هر اجرا اتخاذ کرد. (در فصلهای ۱۱ و ۱۲ به بحث دربارهٔ مطالب مربوط به طرح آزمایشی پرداخته‌ایم.)

۹. *اجراهای مدل و تحلیل نتایج.* اجراهای مکرر مدل و سپس تحلیل آنها به منظور براورد معیارهای عملکرد طرحهایی از سیستم که شبیه‌سازی می‌شوند به‌کار می‌رود. (تجزیه و تحلیل تجربه‌های شبیه‌سازی در فصلهای ۱۱ و ۱۲ مورد بحث قرار گرفته است.)

۱۰. *اجراهای بیشتر؟* بر اساس اجراهای کامل شده، تحلیلگر تعیین می‌کند که آیا اجراهای دیگری مورد نیاز است یا نه و اگر چنین است، این اجراها از چه طرحی باید پیروی کنند.

۱۱. *مستندسازی برنامه و گزارش نتایج.* به‌دلایل متعدد، مستندسازی برنامه لازم است. اگر قرار باشد برنامه توسط همان تحلیلگر یا تحلیلگران دیگر باز هم مورد استفاده واقع شود، درک چگونگی کارکرد برنامه ممکن است لازم باشد. این امر اطمینان به برنامه را چنان تقویت خواهد کرد که استفاده‌کنندگان از مدل و سیاستگذاران بتوانند تصمیمهایی بر اساس تجزیه و تحلیل بگیرند. به‌علاوه، اگر قرار باشد برنامه توسط همان تحلیلگر یا تحلیلگر دیگری وارسی شود، انجام این خواسته را با مستندسازی کافی می‌توان به گونه‌ای قابل‌توجه آسان کرد. تنها یک تجربه با برنامه‌ای که به‌قدر کافی مستندسازی نشده است معمولاً برای قانع کردن تحلیلگر در مورد لزوم این گام مهم کافی است. دلیل دیگر مستندسازی مدل این است که استفاده‌کنندگان از آن بتوانند به اختیار پارامترهای مدل را تغییر دهند تا روابط بین پارامترهای ورودی و معیارهای عملکرد خروجی را مشخص کنند، یا پارامترهای ورودی را که معیار عملکرد خروجی خاصی را «بهینه می‌کند» تعیین کنند.

نتیجهٔ هرگونه تحلیل باید به روشنی و دقت گزارش شود. با انجام این اقدام استفاده‌کنندگان از مدل (اینک سیاستگذاران) می‌توانند صورتبندی نهایی مسأله، گزینه‌های سیستم مورد نظر، ملاک مقایسهٔ گزینه‌ها، نتایج آزمایشها و راه‌حل پیشنهادی مسأله را بررسی کنند. به‌علاوه، اگر قرار باشد

تصمیمها در سطحی بالاتر توجیه شوند، گزارش باید دلایل توجیهی برای استفاده‌کننده از مدل یا تصمیم‌گیرنده را در بر داشته باشد و به اعتبار مدل و فرایند مدلسازی بیفزاید.

۱۲. *اجرا.* موفقیت گام اجرا به این موضوع بستگی دارد که یازده گام پیش از آن چقدر خوب انجام شده باشند. موفقیت این گام همچنین به میزان شرکت دادن استفاده‌کنندهٔ نهایی مدل در تمام فرایند شبیه‌سازی، از سوی تحلیلگر، بستگی دارد. اگر استفاده‌کننده از مدل به‌طور کامل در فرایند مدلسازی شرکت داده شده باشد و اگر ماهیت مدل و خروجیهای آن را درک کند، احتمال اجرای قوی مدل افزایش می‌یابد [پریتسکر و پگدن، ۱۹۷۹]. به طریق وارون، اگر مدل و فرضهای اساسی آن به‌طور مناسبی شناسانیده نشود، گام اجرا احتمالاً صرفنظر از اعتبار مدل شبیه‌سازی، آسیب خواهد دید.

فرایند ساخت مدل شبیه‌سازی را که در شکل ۱-۴ نشان داده شد می‌توان به چهار مرحله تقسیم کرد. مرحلهٔ اول، متشکل از گامهای ۱ (صورتبندی مسأله) و ۲ (تعیین اهداف و طرح کلی پروژه)، دوره‌ای مربوط به اکتشاف یا تعیین جهت است. صورت اولیهٔ مسأله معمولاً بسیار مبهم است، اهداف اولیه معمولاً باید دوباره تعیین شوند و طرح اولیهٔ پروژه معمولاً باید تنظیم مجدد شود. این تنظیمهای مجدد و ابهام‌زداییها را می‌توان در این مرحله یا شاید در مرحله‌ای دیگر انجام داد. (یعنی ممکن است تحلیلگر فرایند را دوباره آغاز کند).

مرحلهٔ دوم مدلسازی و گردآوری داده‌ها و گامهای ۳ (مدلسازی)، ۴ (گردآوری داده‌ها)، ۵ (برنامه‌نویسی)، ۶ (وارسی برنامه)، ۷ (معتبرسازی) را در بر می‌گیرد. میان این گامها رابطهٔ متقابل همیشگی لازم است. کنارگذاشتن استفاده‌کننده از مدل در این مرحله ممکن است به هنگام اجرا پیامدهای فاجعه‌آمیزی در بر داشته باشد.

مرحلهٔ سوم به اجرای مدل مربوط است و گامهای ۸ (طرح آزمایشی)، ۹ (اجرای مدل و تحلیل نتایج)، و ۱۰ (اجراهای بیشتر) را در بر می‌گیرد. این مرحله باید برنامه‌ای به‌دقت طراحی شده برای اجرای تجربه با به‌کارگیری مدل شبیه‌سازی داشته باشد. هر شبیه‌سازی تصادفی مبتنی بر پیشامدهای گسسته، در واقع، تجربه‌ای آماری است. متغیرهای خروجی، براوردهایی در بردارندهٔ خطای تصادفی‌اند و بدین ترتیب، تحلیل مناسب آماری لزوم می‌یابد. این فلسفه با دید تحلیلگری در تضاد است که تنها به یک اجرا می‌پردازد و از تنها یک قلم داده نتیجه‌ای ارائه می‌دهد.

مرحلهٔ چهارم و آخر، یعنی به‌کارگیری، گامهای ۱۱ (مستندسازی برنامه و گزارش نتایج) و ۱۲ به‌کارگیری مدل را در بر دارد. اجرای موفقیت‌آمیز به شرکت دادن مداوم استفاده‌کننده از مدل و تکمیل موفقیت‌آمیز هر یک از گامهای فرایند بستگی دارد. شاید نقطهٔ تعیین‌کننده در سراسر فرایند، گام ۷ (معتبرسازی) باشد زیرا مدلی بی‌اعتبار به نتایجی غلط می‌انجامد که در صورت به‌کارگیری ممکن است خطرناک و یا پرهزینه باشد.

منابع

Adkins, Gerald, and Udo W. Pooch [1977], "Computer Simulation: A Tutorial," *Computer*, Vol. 10, No. 4, pp. 12-17.

www.matlabdl.com

Gordon, Geoffrey [1978], *System Simulation*, 2nd ed., Prentice-Hall, Englewood Cliffs, N.J.

Hillier, Frederick S., and Gerald J. Lieberman [1980], *Introduction to Operations Research*, 3rd ed., Holden-Day, San Francisco.

Law, Averill M., and W. David Kelton [1982], *Simulation Modeling and Analysis*, McGraw-Hill, New York.

Mihram, Danielle, and G. Arthur Mihram [1974], " Human Knowledge, The Role of Models, Metaphors and Analogy," *International Journal of General Systems*, Vol. 1, No. 1, pp. 41-60.

Morris, W. T. [1967], "On the Art of Modeling," *Management Science*, Vol. 13, No. 12.

Naylor, T. H., J. L. Balintfy, D. S. Burdick, and K. Chu [1966], *Computer Simulation Techniques*, Wiley, New York.

Pritsker, A. Alan B., and Claude D. Pegden [1979], *Introduction to Simulation and Slam*, Halsted Press, New York.

Schmidt, J. W., and R. E. Taylor [1970], *Simulation and Analysis of Industrial Systems*, Irwin, Homewood, Ill.

Shannon, Robert E. [1975], *Systems Simulation: The Art and Science*, Prentice-Hall, Englewood Cliffs, N.J.

تمرینها

۱-۱ در مورد سیستمهای زیر، چند نهاد، خصیصه، فعالیت، پیشامد، و متغیر حالت را نام ببرید:

الف) تعمیرگاه وسایل برقی خانگی

ب) کافه تریا

ج) فروشگاه مواد غذایی

د) لباسشویی عمومی

ه) غذاخوری سرپایی

و) اتاق اورژانس بیمارستان

ز) شرکت تاکسیرانی با ۱۰ دستگاه خودرو

ح) خط مونتاژ خودرو

۱-۲ فرایند شبیه‌سازی که در شکل ۱-۴ نشان داده شد را در نظر آورید.

الف) با ادغام کردن فعالیتهای همانند، دستکم دو گام را از تعداد گامها کم کنید و منطق خود را ارائه دهید.

ب) با جداکردن یا افزودن بر گامهای موجود، دستکم دو گام به تعداد گامها بیفزایید و منطق خود را ارائه دهید.

۱-۳ شبیه‌سازی ترافیک در تقاطعی مهم قرار است با این هدف انجام گیرد که جریان فعلی ترافیک را اصلاح کند. گامهای ۱ و ۲ی فرایند شبیه‌سازی شکل ۱-۴ را سه بار تکرار کنید، به طوری که هر بار تکرار از بار قبلی پیچیده‌تر باشد.

۱-۴ از کدام راهها و با چه گامهایی می‌توان برای اجرای فرایند شبیه‌سازی شکل ۱-۴ از کامپیوتر شخصی استفاده کرد؟

۱-۵ فهرست زمینه‌های به‌کارگیری شبیه‌سازی در بخش ۱-۳ را در نظر بگیرید. این زمینه‌ها را به درسهای موجود در برنامهٔ رسمی تحصیلی خود ارتباط دهید.

۱-۶ قرار است پخت اسپاگتی برای شام شبیه‌سازی و تعیین شود که برای حاضر بودن شام رأس ساعت ۷ شب بر روی میز، چه موقع باید کار را شروع کرد. دستور تهیهٔ اسپاگتی را بخوانید (یا آن را از یک دوست یا خویشاوند، یا ... بپرسید). به منظور اجرای شبیه‌سازی به‌نحوی که مدل همهٔ مراحل تهیهٔ غذا را دربر داشته باشد، آنچه را که فکر می‌کنید در قسمت گردآوری داده‌های فرایند شبیه‌سازی شکل ۱-۴ مورد نیاز است به بهترین وجه ممکن پیگیری کنید.

۱-۷ پیشامدها و فعالیتهای مربوط به عملیات دفترچه حساب جاری شما کدام‌اند؟

www.matlabdl.com

۲

مثالهایی از شبیه‌سازی

هدف این فصل ارائهٔ مثالهای متعددی از شبیه‌سازی است که مستقیماً، یعنی بدون استفاده از کامپیوتر قابل اجرا باشد. مثالهایی از این قبیل شناخت مناسبی از روش شبیه‌سازی سیستمهای گسسته و تجزیه و تحلیل آن در اختیار خواننده قرار می‌دهد. با ارائهٔ مثالهای شبیه‌سازی در این مرحله از شروع کتاب، خواننده ارزش بسیاری از نکات ظریفی را که در فصلهای بعد ارائه می‌شوند خواهد دانست. شبیه‌سازیهای این فصل با برداشتن سه گام زیر انجام می‌شود:

۱. ویژگیهای هر یک از ورودیهای شبیه‌سازی را تعیین کنید. در اکثر موارد، این‌گونه ویژگیها را می‌توان در قالب توزیعهای پیوسته یا گسستهٔ احتمال مدلسازی کرد.

۲. یک جدول شبیه‌سازی ایجاد کنید. هر مسألهٔ شبیه‌سازی جدول شبیه‌سازی خاص خود را دارد، زیرا هر جدول برای هدف خاصی ایجاد می‌شود. جدول ۲-۱ مثالی از جدول شبیه‌سازی است. در این مثال تعداد ورودیها، x_{ij}، مساوی p است، یعنی $j = ۱, \ldots, p$ و به ازای هر تکرار، $i = ۱, ۲, \ldots, n$ ، یک پاسخ، y_i، وجود دارد.

۳. در نوبت iام تکرار، مقداری برای هر یک از p ورودی تولید و تابع محاسبه‌کنندهٔ مقدار پاسخ y_i را ارزیابی کنید. این گام با نمونه‌گیری از توزیعهای تعیین شده در گام ۱ اجرا می‌شود.

شبیه‌سازی ابزاری نیرومند است که می‌توان آن را به منظور تحلیل بسیاری از مسائل پیچیده به‌کار برد. اما، پیش از آنکه شبیه‌سازی به‌عنوان راه حل برگزیده شود، باید هر کوشش ممکن برای حل ریاضی مسأله، احیاناً با مدلهای ریاضی موجود برای مسائل صف یا شاید با نظریهٔ کنترل موجودی و ... به‌عمل آید. ساختن مدل شبیه‌سازی ممکن است عملی وقتگیر باشد و اگر راه‌حل بسته‌ای موجود باشد، ممکن است بسیار کم‌هزینه‌تر از شبیه‌سازی باشد.

جدول ۲-۱ جدول شبیه‌سازی.

پاسخ (y_i)	ورودیها x_{ip}	...	x_{ij}	...	x_{i2}	x_{i1}	دفعات تکرار
							۱
							۲
							۳
							⋮
							n

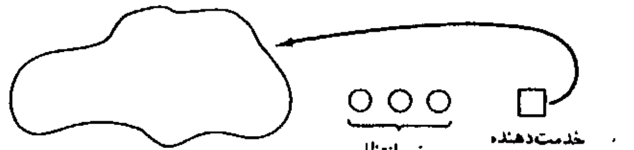
در این فصل، مثالهای بسیار دربارهٔ شبیه‌سازی ارائه می‌کنیم. دو زمینهٔ نخست کاربرد به مدلهای صف وکنترل موجودی مربوط‌اند. شبیه‌سازی در حل مسائل واقعی موجود در این دو زمینه بسیار سودمند واقع شده است. برای اینکه خواننده موفق به کسب شناختی از شرایط ایجادکنندهٔ این مسائل شود، توضیحی مقدماتی ارائه می‌کنیم. سپس، در فصلهای ۵ و ۶ مدلهای صف و سیستمهای مربوط به موجودی را با تفصیل بیشتری شرح می‌دهیم.

در این فصل، مثالهای جالب دیگری نیز عرضه می‌کنیم. اولین آنها مسألهٔ مربوط به پایایی است ـ زمینهٔ دیگری که کاربرد شبیه‌سازی در آن سودمند واقع شده است. مثال دیگری نیز وجود دارد که مفهوم اعداد تصادفی نرمال را عرضه می‌دارد. سرانجام، مثالی نیز در زمینهٔ تعیین تقاضا در مهلت تحویل ارائه می‌کنیم.

۲-۱ شبیه‌سازی سیستمهای صف

سیستم صف با جمعیت متقاضی، چگونگی ورود و خدمتدهی، ظرفیت سیستم و نظام صف مشخص می‌شود. این ویژگیهای سیستم صف را به تفصیل در فصل ۵ شرح داده‌ایم. یک سیستم سادهٔ صف در شکل ۲-۱ نمایش داده شده است.

در این سیستم، جمعیت متقاضی نامحدود است؛ یعنی، اگر یک نفر، جمعیت متقاضی را ترک



شکل ۲-۱ سیستم صف.

کند و به صف انتظار ملحق شود یا به محل دریافت خدمت برود، هیچ‌گونه تغییری در آهنگ ورود سایر متقاضیان نیازمند خدمت روی نخواهد داد. به‌علاوه، در این سیستم، ورودها هر بار یکی و آن نیز به‌صورت تصادفی رخ می‌دهد و اگر واردشدگان به صف انتظار ملحق شوند، سرانجام خدمت دریافت خواهند کرد. در ضمن، مدتهای خدمتدهی تصادفی است و در قالب توزیع احتمالی تعیین می‌شوند که با گذشت زمان بدون تغییر می‌ماند. ظرفیت سیستم نیز نامحدود است. (سیستم، واحد در حال دریافت خدمت و آنهایی که در صف انتظارند را دربرمی‌گیرد.) سرانجام، متقاضیان به‌ترتیب ورود (اغلب مشهور به FIFO) از یک خدمت‌دهنده یا مجرا خدمت می‌گیرند.

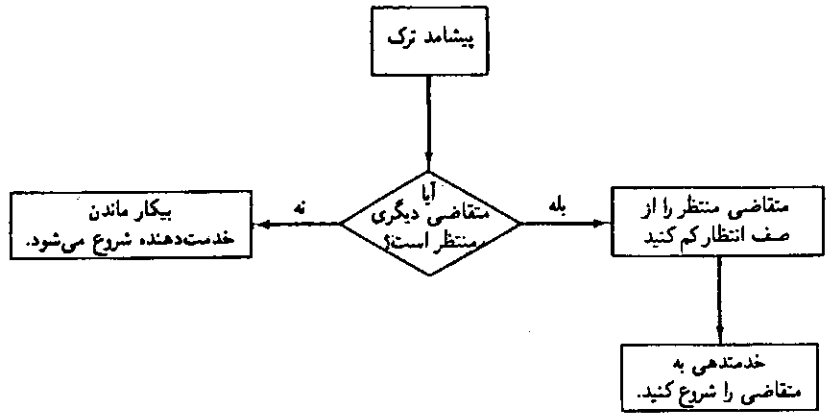
ورودها و خدمت‌دهیها با توزیعهای مدت بین دو ورود و مدتهای خدمتدهی مشخص می‌شوند. به‌طور کلی، آهنگ مؤثر ورود باید از ماکسیمم آهنگ خدمتدهی کمتر باشد وگرنه طول صف انتظار به‌طور نامحدود افزایش می‌یابد. هرگاه صفها به‌طور نامحدود رشد کنند، آنها را «انفجارآمیز» یا ناپایدار می‌نامند. وضعیتی استثنایی مربوط به آهنگهای ورودی است که در دوره‌های زمانی کوتاهی بیش از آهنگهای خدمتدهی باشد. اما، چنین وضعیتی پیچیده‌تر از آن است که در این فصل تشریح شد.

پیش از معرفی چند شبیه‌سازی از سیستمهای صف، درک مفاهیم حالت سیستم، پیشامدها، و ساعت شبیه‌سازی لازم است. حالت سیستم، تعداد حاضران در سیستم و وضعیت خدمت‌دهنده از لحاظ مشغول بودن یا بیکار بودن است. پیشامد مجموعهٔ شرایطی است که موجب تغییری لحظه‌ای در حالت سیستم می‌شود. در مسألهٔ تک مجرایی صف، تنها دو پیشامد ممکن است حالت سیستم را تغییر دهد. این دو پیشامد ورود یک واحد (پیشامد ورود) و پیشامد تکمیل خدمتدهی به یک واحد (پیشامد ترک) است. سیستم صف دربرگیرندهٔ خدمت‌دهنده، واحد در حال خدمتگیری (اگر واحدی در حال خدمتگیری باشد)، و آحاد حاضر در صف (اگر در صف واحدی باشد) است.

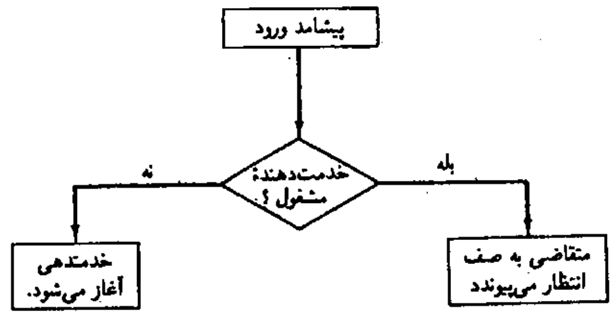
اگر خدمتدهی تازه کامل شده باشد، شبیه‌سازی مطابق دیاگرام جریان که در شکل ۲-۲ نشان داده شده است ادامه می‌یابد. توجه کنید که خدمت‌دهنده در شکل ۲-۲ تنها دو وضعیت دارد: یا مشغول یا بیکار است.

پیشامد دوم هنگامی روی می‌دهد که یک متقاضی به سیستم وارد شود. دیاگرام چنین موردی در شکل ۲-۳ نشان داده شده است. متقاضی واردشده ممکن است خدمت‌دهنده را بیکار یا مشغول بیابد. بنابراین، یا بر خدمت‌دهنده وارد می‌شود یا بدین منظور به صف ملحق می‌شود. اقدام مقتضی در مورد متقاضی مورد بحث به شرح شکل ۲-۴ اعمال می‌شود. اگر خدمت‌دهنده مشغول باشد، متقاضی به صف وارد می‌شود. اگر خدمت‌دهنده بیکار و صف خالی باشد، متقاضی به خدمت‌دهنده وارد می‌شود. این امر غیر ممکن است که خدمت‌دهنده بیکار و صف غیر خالی باشد.

با کامل کردن خدمتدهی ممکن است خدمت‌دهنده بیکار شود یا با خدمتدهی به متقاضی بعدی همچنان مشغول بماند. شکل ۲-۵ رابطهٔ این دو نتیجه با وضعیت صف را نشان می‌دهد. اگر



شکل ۲-۲ دیاگرام جریانِ مربوط به خدمتدهی تازه تکمیل شده.



شکل ۲-۳ دیاگرام جریان ورود به سیستم.

		وضعیت صف	
		خالی	غیرخالی
وضعیت خدمتدهنده	مشغول	ورود به صف	ورود به صف
	بیکار	شروع خدمتدهی	غیرممکن

شکل ۲-۴ عملیات متصور به هنگام ورود یک متقاضی.

		وضعیت صف	
		خالی	غیرخالی
وضعیت خدمتدهنده	مشغول	ناممکن	
	بیکار		ناممکن

شکل ۲-۵ وضعیت خدمت‌دهنده پس از تکمیل خدمتدهی.

صف خالی نباشد، متقاضی دیگری به خدمت‌دهنده می‌رسد و خدمت‌دهنده مشغول می‌ماند. اگر صف خالی باشد، پس از کامل کردن خدمتدهی ، خدمت‌دهنده بیکار خواهد شد. این دو امکان با بخشهای سایه‌خوردهٔ شکل ۲-۵ نشان داده شده است. با کامل شدن هر خدمتدهی، اگر صف خالی باشد، امکان ندارد که خدمت‌دهنده مشغول بماند. همچنین ، پس از کامل شدن خدمتدهی، اگر صف خالی نباشد، امکان ندارد که خدمت‌دهنده بیکار بماند.

اینک باید دید که پیشامدهای پیش گفته چگونه با گذشت زمان شبیه‌سازی رخ می‌دهد. به‌طورکلی، شبیه‌سازی سیستمهای صف ناظر به نگهداری فهرستی از پیشامدهاست تا آنچه را که در زمانهای بعد رخ می‌دهد تعیین کند. این فهرست معرف زمانهای رخ دادن انواع پیشامدهای گوناگون مربوط به هر یک از افراد حاضر در سیستم است. زمانگیری با «ساعتی» که مشخص‌کنندهٔ رخ دادن پیشامدها با گذشت زمان است انجام می‌شود. معمولاً در شبیه‌سازی، پیشامدها به‌طور تصادفی روی می‌دهند. تصادفی بودن تقلیدی از زندگی واقعی است که عدم اطمینان را نشان می‌دهد. مثلاً به‌طور قطع معلوم نیست که چه موقع مشتری بعدی برای ترک فروشگاه مواد غذایی، به صندوق فروشگاه مراجعه می‌کند، یا قطعاً معلوم نیست که چقدر طول می‌کشد تا کارمند باجهٔ بانک، ثبت یک نقل و انتقال مالی را به اتمام برساند.

معرفی عامل تصادف مورد نیاز برای تقلید زندگی واقعی، با استفاده از «اعداد تصادفی» میسر است. اعداد تصادفی به‌طور یکنواخت و مستقل در فاصلهٔ (۰,۱) توزیع می‌شود. ارقام تصادفی نیز به‌طور یکنواخت روی مجموعهٔ {۰, ۱, ... , ۹} توزیع می‌شود. در تولید اعداد تصادفی می‌توان با کنار هم قرار دادن ارقام تصادفی به تعداد مناسب و نوشتن ممیز در سمت چپ عدد به‌دست آمده به مقصود رسید. تعداد مناسب ارقام را دقتی تعیین می‌کند که داده‌های مصرفی به‌عنوان ورودی باید از آن برخوردار باشد. اگر توزیع ورودیها مقادیری با دو رقم اعشار داشته باشد، از جدول ارقام تصادفی (مثل جدول پ-۱) دو رقم می‌گیریم و برای ایجاد عدد تصادفی، در سمت چپ آن ممیز می‌گذاریم.

اعداد تصادفی را تولید هم می‌توان کرد. هرگاه اعداد با استفاده از شیوه‌ای از قبل تعیین شده تولید شوند، به آنها اعداد شبه تصادفی می‌گویند. چون روش تولید معلوم است، همواره می‌توان پیش از شبیه‌سازی دانست که دنبالهٔ این اعداد کدام است. روشهای مختلف تولید اعداد تصادفی را در فصل ۷ بررسی کرده‌ایم.

در مسألهٔ تک مجرایی صف، مدتهای بین دو ورود و مدتهای خدمتدهی بر اساس توزیعهای این متغیرهای تصادفی تعیین (تولید) می‌شوند. مثالهای زیر نشان می‌دهد که این مدتها چگونه تولید می‌شوند. برای رعایت سادگی فرض کنید که مدتهای بین ورودها با پنج بار ریختن یک تاس و ثبت عددی که بر وجه بالایی تاس نمایان شده است تولید شود. جدول ۲-۲ مجموعهٔ پنج مدت بین ورود تولید شده به این ترتیب را نشان می‌دهد. از این پنج مدت بین دو ورود برای محاسبهٔ زمانهای ورود شش مشتری به سیستم صف استفاده شده است. فرض بر این است که اولین مشتری در زمان صفر وارد می‌شود. با این رخداد، ساعت به‌کار می‌افتد. مشتری دوم، دو واحد زمان بعد از آن، یعنی در زمان ۲ وارد می‌شود. مشتری سوم، چهار واحد زمان بعد، در زمان ۶ وارد می‌شود، و . . . .

مدت مورد نیاز دیگر، مدت خدمتدهی است. جدول ۲-۳ مدتهای خدمتدهی را دربرمی‌گیرد که از توزیع تصادفی مدتهای خدمتدهی تولید شده است. تنها مقادیر ممکن خدمتدهی، یک، دو، سه و چهار واحد زمانی است. با پذیرش این فرض که این مقادیر چهارگانه دارای احتمال رخداد

جدول ۲-۲ مدتهای بین دو ورود و زمانهای ورود.

مشتری	مدت بین دو ورود	زمان ورود برحسب ساعت شبیه‌سازی
۱	-	۰
۲	۲	۲
۳	۴	۶
۴	۱	۷
۵	۲	۹
۶	۶	۱۵

جدول ۲-۳ مدتهای خدمتدهی.

مشتری	مدت خدمتدهی
۱	۲
۲	۱
۳	۳
۴	۲
۵	۱
۶	۴

یکسان‌اند، این مقادیر بدین صورت قابل تولیدند که اعداد یک تا چهار را بر مهره‌هایی می‌نویسیم و با جانشینی آنها را از کلاهی بیرون می‌آوریم و ثبت می‌کنیم. حال، برای شبیه‌سازی سیستم تک مجرایی صف باید مدتهای بین دو ورود و مدتهای خدمتدهی را به هم مرتبط کرد. همچنانکه جدول ۲-۴ نشان می‌دهد، اولین مشتری در زمان صفر وارد و خدمتدهی به او که نیازمند دو دقیقه وقت است، بلافاصله شروع می‌شود. خدمتدهی در زمان ۲ کامل می‌شود. مشتری دوم در زمان ۲ وارد و کار او در زمان ۳ تمام می‌شود. توجه کنید که مشتری چهارم در زمان ۷ وارد شده است ولی خدمتدهی را تا زمان ۹ نمی‌توان شروع کرد. زیرا خدمتدهی به مشتری ۳ تا زمان ۹ تمام نشده است.

جدول ۲-۴ مشخصاً برای مسألهٔ تک مجرایی صف طراحی شده است که به مشتریان بر اساس ترتیب ورود به سیستم خدمت می‌دهد. در این جدول بر اساس ساعت شبیه‌سازی، حساب زمان رخداد هر پیشامد ثبت شده است. در ستون دوم جدول ۲-۴ زمان هر پیشامد ورود ثبت شده است، در حالی‌که در ستون آخر، زمان هر پیشامد ترک ثبت شده است. رخ دادن این دو پیشامد با رعایت ترتیب زمانی در جدول ۲-۵ و شکل ۲-۶ نشان داده شده است.

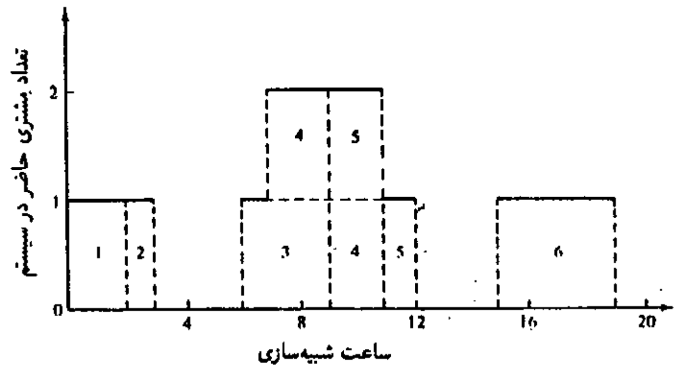
باید توجه داشت که جدول ۲-۵ بر اساس ساعت شبیه‌سازی تنظیم شده است و ممکن است پیشامدها در آن لزوماً برحسب شمارهٔ مشتری مرتب نشده باشد. مرتب کردن پیشامدها برحسب زمان که آن را در فصل ۳ تشریح کرده‌ایم، اساس شبیه‌سازی پیشامدهای گسسته را تشکیل می‌دهد.
شکل ۲-۶ تعداد مشتری حاضر در سیستم را در زمانهای مختلف شبیه‌سازی نشان می‌دهد. در واقع، این شکل نمایش تصویری فهرست مندرج در جدول ۲-۵ است. مشتری ۱ از زمان صفر تا ۲ در سیستم حاضر است. مشتری ۲ در زمان ۲ به سیستم وارد و در زمان ۳ از سیستم خارج می‌شود. از زمان ۳ تا ۶ مشتریانی در سیستم نیستند و در برخی دوره‌ها دو مشتری در سیستم حاضرند؛ مانند زمان ۸ که مشتریان ۳ و ۴ در سیستم حاضرند. زمانهایی نیز وجود دارد که پیشامدها با هم رخ می‌دهند؛ مثل زمان ۹ که مشتری ۳ سیستم را ترک می‌کند و مشتری ۵ به آن وارد می‌شود.

جدول ۲-۴ جدول شبیه‌سازی با تأکید بر اینکه زمانها بر اساس ساعت شبیه‌سازی باشد.

مشتری	زمان ورود	زمان شروع خدمت	مدت خدمتدهی	زمان پایان خدمتدهی
۱	۰	۰	۲	۲
۲	۲	۲	۱	۳
۳	۶	۶	۳	۹
۴	۷	۹	۲	۱۱
۵	۹	۱۱	۱	۱۲
۶	۱۵	۱۵	۴	۱۹

جدول ۲-۵ ترتیب زمانی پیشامدها.

ساعت شبیه‌سازی	مشتری	نوع پیشامد
۰	۱	ورود
۲	۱	ترک
۲	۲	ورود
۳	۲	ترک
۶	۳	ورود
۷	۴	ورود
۹	۳	ترک
۹	۵	ورود
۱۱	۴	ترک
۱۲	۵	ترک
۱۵	۶	ورود
۱۹	۶	ترک



شکل ۲-۶ تعداد مشتری حاضر در سیستم.

مثال ۲-۱ از منطق تشریح شدهٔ بالا پیروی می‌کند در عین اینکه حساب تعدادی از ویژگیهای سیستم را نیز نگه می‌دارد. مثال ۲-۲ به صف دو مجرایی مربوط است. دیاگرامهای جریان سیستم صف چند مجرایی اندکی از دیاگرامهای جریان مربوط به سیستم تک مجرایی متفاوت است. ایجاد و تعبیر این دیاگرامهای جریان را به‌صورت تمرین به عهدهٔ خواننده می‌گذاریم.

■ مثال ۲-۱ صف تک مجرایی

یک فروشگاه مواد غذایی تنها یک باجهٔ صندوق دارد. مشتریها به‌طور تصادفی با فواصل زمانی یک تا ۸ دقیقه به صندوق مراجعه می‌کنند. همان‌طور که جدول ۲-۶ نشان می‌دهد، هر مقدار ممکن برای مدت ورود احتمالی یکسان برای رخ‌دادن دارد. مدتهای خدمتدهی از یک تا شش دقیقه و طبق احتمالات نشان داده شده در جدول ۲-۷ تغییر می‌کند. دو ستون آخر جداول ۲-۶ و ۲-۷ را پس از این تشریح خواهیم کرد. مسأله ناظر به تحلیل سیستم از طریق شبیه‌سازی ورود ۲۰ مشتری و خدمتدهی به آنهاست.

در عمل، اجرایی که ۲۰ مشتری را در برگیرد برای نتیجه‌گیری نهایی بسیار کوچک است. به شرح مطالب فصل ۱۱، از طریق افزایش اندازهٔ نمونه، بر دقت نتایج افزوده می‌شود. اما، هدف این تمرین، تشریح چگونگی اجرای شبیه‌سازیهای دستی است و نه توصیهٔ انجام تغییراتی در فروشگاه.

جدول ۲-۶ توزیع مدتهای بین دو ورود.

مدتهای بین ورود (دقیقه)	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۱	۰٫۱۲۵	۰٫۱۲۵	۰۰۱-۱۲۵
۲	۰٫۱۲۵	۰٫۲۵۰	۱۲۶-۲۵۰
۳	۰٫۱۲۵	۰٫۳۷۵	۲۵۱-۳۷۵
۴	۰٫۱۲۵	۰٫۵۰۰	۳۷۶-۵۰۰
۵	۰٫۱۲۵	۰٫۶۲۵	۵۰۱-۶۲۵
۶	۰٫۱۲۵	۰٫۷۵۰	۶۲۶-۷۵۰
۷	۰٫۱۲۵	۰٫۸۷۵	۷۵۱-۸۷۵
۸	۰٫۱۲۵	۱٫۰۰۰	۸۷۶-۰۰۰

جدول ۲-۷ توزیع مدتهای خدمتدهی.

مدت خدمتدهی (دقیقه)	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۱	۰٫۱۰	۰٫۱۰	۰۱-۱۰
۲	۰٫۲۰	۰٫۳۰	۱۱-۳۰
۳	۰٫۳۰	۰٫۶۰	۳۱-۶۰
۴	۰٫۲۵	۰٫۸۵	۶۱-۸۵
۵	۰٫۱۰	۰٫۹۵	۸۶-۹۵
۶	۰٫۰۵	۱٫۰۰	۹۶-۰۰

www.matlabdl.com

مسألهٔ دیگری که در اینجا وجود دارد و در فصل ۱۱ کاملاً دربارهٔ آن بحث کرده‌ایم، مربوط به شرایط اولیه است. شبیه‌سازی مسألهٔ فروشگاه مواد غذایی اگر با سیستمی خالی شروع شود، واقع‌بینانه نیست، مگر اینکه تعمداً سیستم را از زمان شروع فعالیت مدلسازی کنیم یا اینکه مدلسازی را تا رسیدن فعالیت به حالت پایا ادامه دهیم. اما، برای تسهیل جنبهٔ آموزشی این مثال، از شرایط اولیه و دیگر نکات مورد علاقه در می‌گذریم.

به منظور تولید ورودها به باجهٔ صندوق، به مجموعه‌ای از اعداد تصادفی با توزیع یکنواخت نیاز داریم. اعداد تصادفی دارای خاصیتهای زیرند:

۱. مجموعهٔ اعداد تصادفی به‌طور یکنواخت بین صفر و یک توزیع می‌شوند.

۲. اعداد تصادفی متوالی مستقل‌اند.

قبلاً خاطرنشان کردیم که اعداد تصادفی با توزیع یکنواخت از راههای فراوانی قابل تولیدند که برخی از این راهها را در فصل ۷ تشریح کرده‌ایم. به‌علاوه، جدولهای ارقام تصادفی نیز در دسترس‌اند. برای تولید اعداد تصادفی در این مثال، از این جدولها استفاده کرده‌ایم.

با زدن ممیز در جای مناسب، ارقام تصادفی به اعداد تصادفی تبدیل می‌شوند. در این مثال، اعداد تصادفی با سه رقم اعشار کافی‌اند. برای تولید مدتهای بین دو ورود، تنها به ۱۹ عدد تصادفی نیازمندیم. چرا فقط به ۱۹ عدد؟ اولین ورود طبق فرض در زمان صفر رخ می‌دهد، پس برای ۲۰ نفر شدن تعداد مشتریها نیاز به تولید ۱۹ ورود دیگر داریم.

از دو ستون آخر جدولهای ۲-۶ و ۲-۷ برای تولید ورودها و مدتهای خدمتدهی تصادفی استفاده می‌شود. ستون ما قبل آخر هر جدول نمایشگر احتمال تجمعی مربوط به‌هر توزیع است. ستون آخر، تخصیص ارقام تصادفی را نشان می‌دهد. تنها جدول ۲-۶ را در نظر بگیرید. اولین ارقام تصادفی تخصیص‌یافته ۰۰۱-۱۲۵ است. شمار اعداد سه رقمی ممکن، ۱۰۰۰ است (۰۰۱ تا ۰۰۰). احتمال یک دقیقه شدن مدت بین ورود ۰٫۱۲۵ است و ۱۲۵ عدد سه رقمی از ۱۰۰۰ عدد ممکن به چنین رخدادی تخصیص می‌یابد. به اعداد سه رقمی نیاز داریم، زیرا توزیع احتمال با دقت سه رقم اعشار تعریف شده است. مثلاً، احتمال ۴ دقیقه شدن مدت بین دو ورود ۰٫۱۲۵ است. با تهیهٔ فهرست ۱۹ عدد سه رقمی از جدول پ-۱ و مقایسهٔ آنها با ارقام تصادفی تخصیص یافته در جدول ۲-۶، مدتهای بین ورود برای ۱۹ مشتری تولید می‌شود.

شروع در موقعیتی تصادفی در جدول ارقام تصادفی و پیشروی در جهتی منظم، بدون اینکه در مسألهٔ مفروض هرگز از دنبالهٔ واحدی از ارقام دو بار استفاده کنیم، شیوهٔ مناسبی است. استفادهٔ مکرر از الگوی واحد ممکن است موجب ایجاد اریبی شود زیرا الگویی واحد از پیشامدها تولید خواهد شد. تعیین مدت بین دو ورود را در جدول ۲-۸ نشان داده‌ایم. توجه کنید که اولین ارقام تصادفی ۹۱۳ است. برای یافتن مدت بین ورود مربوطه، به ستون چهارم جدول ۲-۶ وارد شوید و در ستون اول جدول، ۸ دقیقه را بخوانید.

** تعیین زمان ورود جهت هر مشتری:

مشتری ۴، دقیقه چندم را رسیده است؟ دقیقهٔ ۱۵ = ۱ + ۶ + ۸ + ۰

مشتری ۵، دقیقه چندم وارد شده است؟ دقیقهٔ ۲۳ = ۸ + ۱ + ۶ + ۸ + ۰

جدول ۲-۸ تعیین مدتهای بین دو ورود.

مشتری	ارقام تصادفی	مدت بین دو ورود (دقیقه)	مشتری	ارقام تصادفی	مدت بین دو ورود (دقیقه)
۱	-	-	۱۱	۱۰۹	۱
۲	۹۱۳	۸	۱۲	۰۹۳	۱
۳	۷۲۷	۶	۱۳	۶۰۷	۵
۴	۰۱۵	۱	۱۴	۷۳۸	۶
۵	۹۴۸	۸	۱۵	۳۵۹	۳
۶	۳۰۹	۳	۱۶	۸۸۸	۸
۷	۹۲۲	۸	۱۷	۱۰۶	۱
۸	۷۵۳	۷	۱۸	۲۱۲	۲
۹	۲۳۵	۲	۱۹	۴۹۳	۴
۱۰	۳۰۲	۳	۲۰	۵۳۵	۵

**

* جدول ۲-۹ مدتهای تولید شده برای خدمتدهی.

مشتری	ارقام تصادفی	مدت خدمتدهی (دقیقه)	مشتری	ارقام تصادفی	مدت خدمتدهی (دقیقه)
۱	۸۴	۴	۱۱	۳۲	۳
۲	۱۰	۱	۱۲	۹۴	۵
۳	۷۴	۴	۱۳	۷۹	۴
۴	۵۳	۳	۱۴	۰۵	۱
۵	۱۷	۲	۱۵	۷۹	۵
۶	۷۹	۴	۱۶	۸۴	۴
۷	۹۱	۵	۱۷	۵۲	۳
۸	۶۷	۴	۱۸	۵۵	۳
۹	۸۹	۵	۱۹	۳۰	۲
۱۰	۳۸	۳	۲۰	۵۰	۳

مدتهای خدمتدهی برای هر ۲۰ مشتری در جدول ۲-۹ نشان داده شده است. این مدتها بر اساس روش تشریح شده در فوق و با استفاده از جدول ۲-۷ تولید شده‌اند. مدت خدمتدهی به مشتری اول ۴ دقیقه است زیرا ارقام تصادفی ۸۴ در ردهٔ ۶۱-۸۵ قرار می‌گیرد.

جدول شبیه‌سازی خلاصهٔ شبیه‌سازی دستی است. این‌گونه جدولها برای وضعیت در دست

جدول ۲-۱۰ جدول شبیه‌سازی برای مسألهٔ صف.

مشتری	مدت سپری شده از آخرین ورود (دقیقه)	زمان ورود	مدت خدمتدهی (دقیقه)	زمان شروع خدمت	مدت ماندن مشتری در صف (دقیقه)	زمان پایان خدمت	مدت ماندن مشتری در سیستم (دقیقه)	مدت بیکاری خدمت‌دهنده (دقیقه)
۱	-	۰	۴	۰	۰	۴	۴	۰
۲	۸	۸	۱	۸	۰	۹	۱	۴
۳	۶	۱۴	۴	۱۴	۰	۱۸	۴	۵
۴	۱	۱۵	۳	۱۸	۳	۲۱	۶	۰
۵	۸	۲۳	۲	۲۳	۰	۲۵	۲	۲
۶	۳	۲۶	۴	۲۶	۰	۳۰	۴	۱
۷	۸	۳۴	۵	۳۴	۰	۳۹	۵	۴
۸	۷	۴۱	۴	۴۱	۰	۴۵	۴	۲
۹	۲	۴۳	۵	۴۵	۲	۵۰	۷	۰
۱۰	۳	۴۶	۳	۵۰	۴	۵۳	۷	۰
۱۱	۱	۴۷	۳	۵۳	۶	۵۶	۹	۰
۱۲	۱	۴۸	۵	۵۶	۸	۶۱	۱۳	۰
۱۳	۵	۵۳	۴	۶۱	۸	۶۵	۱۲	۰
۱۴	۶	۵۹	۱	۶۵	۶	۶۶	۷	۰
۱۵	۳	۶۲	۵	۶۶	۴	۷۱	۹	۰
۱۶	۸	۷۰	۴	۷۱	۱	۷۵	۵	۰
۱۷	۱	۷۱	۳	۷۵	۴	۷۸	۷	۰
۱۸	۲	۷۳	۳	۷۸	۵	۸۱	۸	۰
۱۹	۴	۷۷	۲	۸۱	۴	۸۳	۶	۰
۲۰	۵	۸۲	۳	۸۳	۱	۸۶	۴	۰
	۸۲		۶۸		۵۶		۱۲۴	۱۸

بررسی طراحی و چنان ساخته می‌شوند که پرسشهای مطروحه را پاسخ گویند. جدول شبیه‌سازی این مسأله را در جدول ۲-۱۰ نشان دادیم که متممی برای جدولی از نوع جدول ۲-۴ است که قبلاً دیده‌ایم. فرض می‌کنیم که مشتری اول در زمان صفر وارد شود. خدمتدهی بلافاصله شروع و در زمان ۴ تمام می‌شود. مشتری به مدت ۴ دقیقه در سیستم بوده است. مشتری دوم در زمان ۸ وارد می‌شود. بدین ترتیب، خدمت‌دهنده (دریافت‌کنندهٔ پول) به مدت ۴ دقیقه بیکار بوده است. با بررسی مشتری چهارم، دیده می‌شود که این مشتری در زمان ۱۵ وارد شده و شروع خدمتدهی به او تا زمان ۱۸ ممکن نبوده است. این مشتری ناچار از انتظارکشیدن در صف به مدت ۳ دقیقه شده

است. این فرایند در مورد هر ۲۰ مشتری اجرا می‌شود. همان‌طور که دیده می‌شود، مجموعها برای مدتهای خدمتدهی، مدتهای ماندن مشتریان در سیستم، مدت بیکاری خدمت‌دهنده و مدت انتظار مشتریان در صف تعیین می‌شود.

برخی از یافته‌های این شبیه‌سازی کوتاه‌مدت به شرح زیر است:

۱. متوسط مدت انتظار هر مشتری ۲٫۸ دقیقه است. این نتیجه به طریق زیر تعیین می‌شود:

$$\text{متوسط مدت انتظار (دقیقه)} = \frac{\text{مجموع مدت انتظار مشتریان در صف (دقیقه)}}{\text{مجموع تعداد مشتریان}}$$

$$= \frac{۵۶}{۲۰} = ۲٫۸ \quad \text{دقیقه}$$

۲. احتمال مجبور شدن هر مشتری به‌انتظارکشیدن در صف ۰٫۶۵ است. این نتیجه به طریق زیر تعیین می‌شود:

$$\text{احتمال (انتظار)} = \frac{\text{تعداد مشتریانی که در انتظار می‌مانند}}{\text{مجموع تعداد مشتریان}}$$

$$= \frac{۱۳}{۲۰} = ۰٫۶۵$$

۳. نسبت مدت بیکاری خدمت‌دهنده ۰٫۲۱ است. این نتیجه به طریق زیر تعیین می‌شود:

$$\text{احتمال بیکاری خدمت‌دهنده} = \frac{\text{مجموع مدت بیکاری خدمت‌دهنده (دقیقه)}}{\text{مجموع مدت اجرای شبیه‌سازی (دقیقه)}}$$

$$= \frac{۱۸}{۸۶} = ۰٫۲۱$$

احتمال مشغول بودن خدمت‌دهنده: مکمل ۰٫۲۱ یا ۰٫۷۹ است.

۴. متوسط مدت خدمتدهی ۳٫۴ دقیقه است. نتیجه به طریق زیر تعیین می‌شود:

$$\text{متوسط مدت خدمتدهی} = \frac{\text{مجموع مدت خدمتدهی (دقیقه)}}{\text{مجموع تعداد مشتریان}} = \frac{۶۸}{۲۰} = ۳٫۴ \quad \text{دقیقه}$$

می‌توان این نتیجه را با یافتن میانگین توزیع مدت خدمتدهی با بکارگیری معادلهٔ

$$E(S) = \sum_{s=0}^{\infty} s\,p(s)$$

با امید ریاضی مدت خدمتدهی مقایسه کرد. با بکارگیری معادلهٔ امید ریاضی در مورد توزیع

مندرج در جدول ۲-۷ به نتیجهٔ زیر می‌رسیم.

$$\text{امید ریاضی مدت خدمتدهی} = ۱(۰٫۱۰) + ۲(۰٫۲۰) + ۳(۰٫۳۰)$$
$$+ ۴(۰٫۲۵) + ۵(۰٫۱۰) + ۶(۰٫۰۵)$$
$$= ۳٫۲ \ \text{دقیقه}$$

امید ریاضی مدت خدمتدهی اندکی کمتر از متوسط مدت خدمتدهی در شبیه‌سازی است. هر چه شبیه‌سازی طولانیتر باشد، این متوسط به $E(S)$ نزدیکتر می‌شود.

۵. متوسط مدت بین دو ورود ۴٫۳ دقیقه است. این نتیجه به طریق زیر تعیین می‌شود:

$$\text{متوسط مدت بین دو ورود (دقیقه)} = \frac{\text{جمع تمام مدتهای بین دو ورود (دقیقه)}}{\text{تعداد ورودها منهای یک}}$$
$$= \frac{۸۲}{۱۹} = ۴٫۳ \ \text{دقیقه}$$

زمان بین دو ورود ۱۹ مشتری داریم

یک را از مخرج کم می‌کنیم، زیرا فرض بر این است که اولین ورود در زمان صفر روی می‌دهد. می‌توان این نتیجه را با یافتن میانگین توزیع یکنواخت گسسته‌ای که نقاط شروع و پایان آن $a = ۱$ و $b = ۸$ است، مقایسه کرد. میانگین از رابطهٔ

در بازه $[۱،۸]$ چون زمان بین دو ورود از ۱ دقیقه تا ۸ دقیقه است

$$E(A) = \frac{a+b}{۲} = \frac{۱+۸}{۲} = ۴٫۵ \ \text{دقیقه}$$

میانگین توزیع یکنواخت گسسته = $\frac{a+b}{۲}$ یا امید

به‌دست می‌آید. امید ریاضی مدت بین ورودها کمی بیش از مقدار متوسط است. اما، در شبیه‌سازیهای طولانیتر مقدار متوسط مدت بین ورودها باید به میانگین تئوریک، $E(A)$، میل کند.

۶. متوسط مدت انتظار آنهایی که به انتظار می‌مانند ۴٫۳ دقیقه است. این، به طریق زیر تعیین می‌شود:

$$\text{متوسط مدت انتظار آنهایی که به انتظار می‌مانند (دقیقه)} = \frac{\text{مجموع مدتی که مشتریان در صف به انتظار می‌مانند (دقیقه)}}{\text{مجموع تعداد مشتریانی که در صف به انتظار می‌مانند}}$$
$$= \frac{۵۶}{۱۳} = ۴٫۳ \ \text{دقیقه}$$

۷. متوسط مدتی که هر مشتری در سیستم می‌گذراند ۶٫۲ دقیقه است. این نتیجه را می‌توان

از دو طریق به‌دست آورد. اول اینکه، محاسبه را می‌توان با استفاده از رابطهٔ زیر انجام داد

$$\text{متوسط مدت ماندن مشتری در سیستم (دقیقه)} = \frac{\text{مجموع مدت ماندن مشتریان در سیستم (دقیقه)}}{\text{مجموع تعداد مشتریان}}$$
$$= \frac{۱۲۴}{۲۰} = ۶٫۲ \ \text{دقیقه}$$

راه دوم محاسبهٔ همین نتیجه، تشخیص این مطلب است که رابطهٔ زیر باید برقرار باشد:

$$\text{متوسط مدتی که مشتری در سیستم می‌ماند (دقیقه)} = \text{متوسط مدتی که مشتری در صف به انتظار می‌ماند (دقیقه)} + \text{متوسط مدتی که مشتری برای خدمتگیری صرف می‌کند (دقیقه)}$$

یافته‌های ۱ و ۴ فهرست بالا داده‌های لازم برای سمت راست این معادله را فراهم می‌آورند تا نتیجهٔ

$$\text{متوسط مدتی که مشتری در سیستم می‌ماند (دقیقه)} = ۲٫۸ + ۳٫۴ = ۶٫۲ \ \text{دقیقه}$$

به‌دست آید.

تصمیم‌گیرنده به چنین نتایجی علاقه‌مند است، ولی شبیه‌سازیهای طولانیتر دقت یافته‌ها را افزایش می‌دهد. اما، در این مرحله می‌توان به برخی نتیجه‌گیریهای عینی دست یافت. اکثر مشتریان ناچار به انتظار ماندن هستند؛ هر چند که متوسط مدت انتظار زیاده از حد نیست. برای خدمت‌دهنده مدت بیکاری نامناسبی پیش نمی‌آید. اظهارات عینی در مورد نتایج، به مقایسهٔ هزینهٔ انتظار با هزینهٔ خدمت‌دهنده‌های بیشتر بستگی دارد. (شبیه‌سازیهایی را که به شکلهای دیگری از توزیعهای ورود و خدمتدهی نیاز دارد، به‌عنوان تمرین برای خواننده ارائه کرده‌ایم.) ■

■ مثال ۲-۲ مسألهٔ اتو رستوران هابیل و خباز

هدف این مثال، ارائهٔ شیوهٔ شبیه‌سازی در موردی است که بیش از یک مجرا وجود داشته باشد. یک اتو رستوران را در نظر بگیرید که آورندگان غذا سفارشها را دریافت می‌کنند و غذا را به خودروها می‌آورند. خودروها به‌صورت مندرج در جدول ۲-۱۱ وارد می‌شوند. تعداد آورندگان غذا دو نفر است—هابیل و خباز. هابیل برای انجام این کار تواناتر است و کمی سریعتر از خباز کار می‌کند. توزیع مدتهای خدمتدهی در جدولهای ۲-۱۲ و ۲-۱۳ نشان داده شده است.

شبیه‌سازی به طریقی همانند مثال ۲-۱ انجام می‌گیرد، با این تفاوت که این‌بار پیچیده‌تر است. قاعدهٔ ساده‌کننده این است که اگر هر دو آورندهٔ غذا بیکار باشند، هابیل مشتری از راه رسیده را می‌گیرد؛ شاید هابیل با سابقه‌تر باشد. (اگر تصمیم در این مورد که هرگاه هر دو بیکارند چه کسی به خودروی وارد شده خدمت دهد بر پایهٔ تصادفی استوار می‌بود جواب متفاوتی به‌دست می‌آمد.)

مسأله این است که روش فعلی تا چه حد خوب کار می‌کند. برای براورد معیارهای عملکرد

جدول ۲-۱۱ توزیع مدتهای بین ورود خودروها.

مدتهای بین دو ورود(دقیقه)	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۱	۰٫۲۵	۰٫۲۵	۰۱-۲۵
۲	۰٫۴۰	۰٫۶۵	۲۶-۶۵
۳	۰٫۲۰	۰٫۸۵	۶۶-۸۵
۴	۰٫۱۵	۱٫۰۰	۸۶-۰۰

جدول ۲-۱۲ توزیع خدمتدهی هابیل.

مدت خدمتدهی (دقیقه)	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۲	۰٫۳۰	۰٫۳۰	۰۱-۳۰
۳	۰٫۲۸	۰٫۵۸	۳۱-۵۸
۴	۰٫۲۵	۰٫۸۳	۵۹-۸۳
۵	۰٫۱۷	۱٫۰۰	۸۴-۰۰

جدول ۲-۱۳ توزیع خدمتدهی خباز.

مدت خدمتدهی (دقیقه)	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۳	۰٫۳۵	۰٫۳۵	۰۱-۳۵
۴	۰٫۲۵	۰٫۶۰	۳۶-۶۰
۵	۰٫۲۰	۰٫۸۰	۶۱-۸۰
۶	۰٫۲۰	۱٫۰۰	۸۱-۰۰

سیستم، از شبیه‌سازی یک ساعتهٔ عملیات سیستم استفاده می‌کنیم. شبیه‌سازی طولانیتر بسیار مطمئنتر خواهد بود، ولی بنا به مقاصد توضیحی دورهٔ یک ساعته را انتخاب کرده‌ایم.

شبیه‌سازی بیشتر به شیوهٔ مثال ۲-۱ انجام می‌شود. در این مثال شبیه‌سازی پیشامدهای بیشتری وجود دارد، زیرا خدمت‌دهندهٔ دومی نیز داریم. پیشامدهای ممکن به شرح زیر است: مشتریان وارد می‌شوند، خدمتدهی به مشتریان توسط هابیل شروع می‌شود. خدمتدهی به مشتریان توسط

هابیل کامل می‌شود. خدمتدهی به مشتریان توسط خباز شروع می‌شود. خدمتدهی به مشتریان توسط خباز کامل می‌شود. جدول ۲-۱۴ جدول شبیه‌سازی را نشان می‌دهد.

جدول ۲-۱۴ جدول شبیه‌سازی برای مثال رستوران.

مشتری	ارقام تصادفی ورود	مدتهای بین دو ورود	زمانهای ورود برحسب ساعت شبیه‌سازی	ارقام تصادفی خدمتدهی	هابیل: زمانهای شروع خدمت	هابیل: مدتهای خدمتدهی	هابیل: زمانهای پایان خدمت	خباز: زمانهای شروع خدمت	خباز: مدتهای خدمتدهی	خباز: زمانهای پایان خدمت	مدت انتظار در صف
۱	-	-	۰	۹۵	۰	۵	۵				۰
۲	۲۶	۲	۲	۲۱				۲	۳	۵	۰
۳	۹۸	۴	۶	۵۱	۶	۳	۹				۰
۴	۹۰	۴	۱۰	۹۲	۱۰	۵	۱۵				۰
۵	۲۶	۲	۱۲	۸۹				۱۲	۶	۱۸	۰
۶	۴۲	۲	۱۴	۳۸	۱۵	۳	۱۸				۱
۷	۷۴	۳	۱۷	۱۳	۱۸	۲	۲۰				۱
۸	۸۰	۳	۲۰	۶۱	۲۰	۴	۲۴				۰
۹	۶۸	۳	۲۳	۵۰				۲۳	۴	۲۷	۰
۱۰	۲۲	۱	۲۴	۴۹	۲۴	۳	۲۷				۰
۱۱	۴۸	۲	۲۶	۳۹	۲۷	۳	۳۰				۱
۱۲	۳۴	۲	۲۸	۵۳				۲۸	۴	۳۲	۰
۱۳	۴۵	۲	۳۰	۸۸	۳۰	۵	۳۵				۰
۱۴	۲۴	۱	۳۱	۰۱				۳۲	۳	۳۵	۱
۱۵	۳۴	۲	۳۳	۸۱	۳۵	۴	۳۹				۲
۱۶	۶۳	۲	۳۵	۵۳				۳۵	۴	۳۹	۰
۱۷	۳۸	۲	۳۷	۸۱	۳۹	۴	۴۳				۲
۱۸	۸۰	۳	۴۰	۶۴				۴۰	۵	۴۵	۰
۱۹	۴۲	۲	۴۲	۰۱	۴۳	۲	۴۵				۱
۲۰	۵۶	۲	۴۴	۶۷	۴۵	۴	۴۹				۱
۲۱	۸۹	۴	۴۸	۰۱				۴۸	۳	۵۱	۰
۲۲	۱۸	۱	۴۹	۴۷	۴۹	۳	۵۲				۰
۲۳	۵۱	۲	۵۱	۷۵				۵۱	~~۵~~	۵۶	۰
۲۴	۷۱	۳	۵۴	۵۷	۵۴	۳	۵۷				۰
۲۵	۱۶	۱	۵۵	۸۷				۵۶	۶	۶۲	۱
۲۶	۹۲	۴	۵۹	۴۷	۵۹	۳	۶۲				۰
						۵۶			۴۳		۱۱

تجزیه و تحلیل جدول ۲-۱۴ به یافته‌های زیر می‌انجامد:

۱. در دورهٔ ۶۲ دقیقه‌ای، هابیل در ۹۰ درصد از وقت به خدمتدهی مشغول است.

۲. خباز تنها ۶۹ درصد از وقت را به خدمتدهی مشغول بوده است. قاعدهٔ حق تقدم موجب شده است که خباز کمتر مشغول باشد.

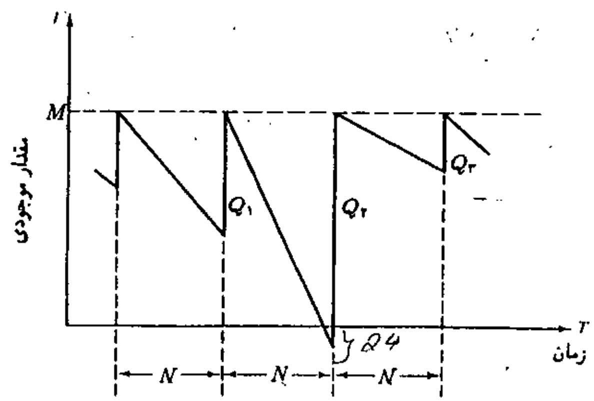
۳. ۹ نفر از ۲۶ نفر افراد واردشده یا در حدود ۳۵ درصد از آنها ناچار از انتظارکشیدن بوده‌اند. متوسط مدت انتظار برای تمام مشتریان تنها ۰٫۴۲ دقیقه یا ۲۵ ثانیه بوده است که مقداری بسیار کم است.

۴. ۹ موردی که مجبور بوده‌اند منتظر بمانند، به‌طور متوسط تنها ۱٫۲۲ دقیقه انتظار کشیده‌اند که مقداری بسیار کم است.

۵. خلاصه، این سیستم متوازن به نظر می‌رسد. یک خدمت‌دهنده نمی‌تواند از عهدهٔ خدمتدهی به همهٔ متقاضیان غذا برآید و سه‌خدمت‌دهنده نیز احتمالاً بسیار زیاد است. افزودن یک خدمت‌دهندهٔ دیگر، مطمئناً مدت انتظار را تقریباً به صفر می‌رساند. اما برای توجیه به‌کارگیری خدمت‌دهندهٔ دیگر، هزینهٔ انتظار باید بسیار زیاد باشد. ■

۲-۲ شبیه‌سازی سیستمهای موجودی

ردهٔ مهمی از مسائل شبیه‌سازی به سیستمهای موجودی مربوط است. سیستم سادهٔ موجودی در شکل ۲-۷ نشان داده شده است. این سیستم موجودی، مروری دوره‌ای به طول N دارد که در آن، سطح موجودی وارسی می‌شود و سفارشی صادر می‌شود که موجودی را به سطح M بالا خواهد آورد. در پایان اولین دورهٔ بررسی، سفارشی به مقدار Q_1 صادر می‌شود. مدت تحویل



شکل ۲-۷ سیستم موجودی احتمالی سطح سفارش.

(یعنی بین صدور و دریافت سفارش) در این سیستم موجودی صفر است. چون مقادیر تقاضا معمولاً با اطمینان مشخص نیست، مقادیر سفارش احتمالی‌اند. در شکل ۲-۷ تقاضا به‌صورت یکنواخت در دورهٔ زمانی نشان داده شده است. مقادیر تقاضا در عمل یکنواخت نیست و با گذشت زمان دستخوش نوسان می‌شود. یک امکان این است که همهٔ تقاضا در شروع دوره برسد. امکان واقع‌بینانهٔ دیگر نیز این است که مهلت تحویل غیرصفر و احتمالی باشد.

توجه کنید که مقدار موجودی در دور دوم به زیر صفر کاهش می‌یابد که این موضوع معرف کمبود است. در شکل ۲-۷، این واحدها سفارش تحویل نشده را تشکیل می‌دهد. هرگاه سفارشی برسد، ابتدا به تقاضای مربوط به اقلام سفارش تحویل نشده پاسخ داده می‌شود. برای پرهیز از کمبود، نیاز به نگهداری یک ذخیره یا موجودی اطمینان وجود دارد.

برای نگهداری موجودی در انبار، هزینه‌ای وجود دارد که آن را به بهرهٔ پرداختی برای تهیهٔ منابع مالی قرض‌شده جهت خرید اقلام نسبت می‌دهند. (این هزینه را می‌توان به‌عنوان زیان ناشی از نبود منابع مالی برای سایر مقاصد سرمایه‌گذاری نیز تعبیر کرد). سایر هزینه‌ها از قبیل اجارهٔ فضای انبار، استخدام نگهبان، و ... را می‌توان در ستون هزینه‌های نگهداری موجودی قرار داد. راه دیگری که می‌توان به‌جای نگهداری موجودی فراوان در انبار از آن بهره گرفت، بررسی وضعیت انبار به دفعات بیشتر و در نتیجه، خریدها یا بازسازیهای موجودی به دفعات بیشتر است. این شیوه نیز هزینه‌ای دارد به نام هزینهٔ سفارشدهی. کم بودن موجودی نیز هزینه‌ای دارد. ممکن است مشتریان ناخرسند شوند و شهرت تجاری بدین‌گونه از دست برود. داشتن موجودیهای زیاد از امکان کمبود می‌کاهد. این هزینه‌ها باید چنان متوازن شوند که هزینهٔ کل سیستم موجودی مینیمم شود.

هزینهٔ کل (یا سود کل) سیستم موجودی، معیار سنجش عملکرد سیستم است که می‌تواند تحت تأثیر خط‌مشی‌های متفاوت قرار گیرد. مثلاً، در شکل ۲-۷ تصمیم‌گیرنده می‌تواند ماکسیمم سطح موجودی، M، و طول دوره، N، راکنترل کند. تغییر دادن N چه تأثیری بر هزینه‌های مختلف می‌گذارد؟

پیشامدهایی که ممکن است در سیستم موجودی (M, N) رخ دهند عبارت‌اند از: تقاضا برای اقلام موجود در انبار، بررسی وضعیت موجودی و دریافت سفارش در پایان هر دورهٔ بررسی. هرگاه مانند شکل ۲-۷ مهلت تحویل صفر باشد، دو پیشامد آخر همزمان رخ می‌دهند.

در مورد سیستمهای موجودی بسیار نوشته شده است. در فصل ۶ خلاصه‌ای از برخی مدلهای متداول موجودی ارائه داده‌ایم. اما، هدف این فصل، تشریح این مطلب است که شبیه‌سازی سیستمهای موجودی چگونه صورت می‌گیرد.

■ **مثال ۲-۳ مسألهٔ روزنامه‌فروش**

(این مثال، تنها به یک دورهٔ محدود زمانی مربوط است و تهیهٔ موجودی در آن تنها یک بار صورت می‌گیرد، موجودی باقیمانده در پایان یک دوره به‌عنوان باطله فروخته یا به‌دور ریخته می‌شود، انواعی

بسیار از مسائل واقعی، از جمله در باب ذخیره‌سازی لوازم یدکی، اقلام فاسد شدنی، کالاهای مربوط به مد و برخی اقلام فصلی از این جمله‌اند [هدلی و ویتین، ۱۹۶۳].)

یک مسألهٔ قدیمی و مهم موجودی، به خرید و فروش روزنامه مربوط است. روزنامه‌فروش، هر نسخه روزنامه را به ۱۳ واحد پول می‌خرد و به ۲۰ واحد پول می‌فروشد. روزنامه‌های فروش‌نرفته در انتهای روز به‌عنوان باطله و هر نسخه به ۲ واحد پول فروخته می‌شود. روزنامه در بسته‌های ده تایی قابل خریدن است و روزنامه‌فروش تنها می‌تواند ۵۰، ۶۰، و ... روزنامه بخرد. از لحاظ چگونگی اخبار سه نوع روز «خوب»، «متوسط»، و «بد» با احتمالات، به‌ترتیب، ۰٫۳۵، ۰٫۴۵ و ۰٫۲۰ وجود دارد. توزیع روزنامهٔ مورد تقاضا در هر یک از این روزها در جدول ۲-۱۵ ارائه شده است. هدف مسأله تعیین تعداد بهینهٔ روزنامه‌هایی است که روزنامه‌فروش باید بخرد. با شبیه‌سازی تقاضا برای ۲۰ روز و ثبت سود ناشی از فروش روزانه، این خواسته را تأمین می‌کنیم. با استفاده از نظریهٔ موجودی در فصل ۶، این مسأله به‌راحتی حل می‌شود. سود طبق رابطهٔ زیر تعیین می‌شود:

سود از دست رفته به خاطر فزونی تقاضا − هزینهٔ خرید روزنامه − درامد ناشی از فروش = سود

درامد ناشی از فروش روزنامه‌های باطله +

به‌موجب صورت مسأله، درامد ناشی از فروش هر نسخه روزنامه ۲۰ واحد پول است. هزینهٔ خرید روزنامه نیز ۱۳ واحد پول به ازای هر نسخه است. سود از دست رفته به‌خاطر فزونی تقاضا به ازای هر نسخه روزنامهٔ مورد تقاضا و غیرموجود، ۷ واحد پول است. هزینهٔ کمبود بدین‌گونه تا حدی قابل بحث است ولی مسأله را بسیار جالب می‌کند. درامد ناشی از فروش روزنامه‌های باطله ۲ واحد پول به ازای هر نسخه است.

جدول ۲-۱۵ توزیع روزنامه‌های مورد تقاضا.

تقاضا	توزیع احتمال تقاضا: خوب	متوسط	بد
۴۰	۰٫۰۳	۰٫۱۰	۰٫۴۴
۵۰	۰٫۰۵	۰٫۱۸	۰٫۲۲
۶۰	۰٫۱۵	۰٫۴۰	۰٫۱۶
۷۰	۰٫۲۰	۰٫۲۰	۰٫۱۲
۸۰	۰٫۳۵	۰٫۰۸	۰٫۰۶
۹۰	۰٫۱۵	۰٫۰۴	۰٫۰۰
۱۰۰	۰٫۰۷	۰٫۰۰	۰٫۰۰

جدولهای ۲-۱۶ و ۲-۱۷ تخصیص ارقام تصادفی برای تعیین نوع روزها و تقاضای مربوط به آن روزها را ارائه می‌کنند. حل این مسأله از راه شبیه‌سازی، نیازمند تعیین خط‌مشی خرید تعداد مشخصی روزنامه در هر روز و سپس شبیه‌سازی تقاضا برای روزنامه طی یک دورهٔ زمانی ۲۰ روزه به منظور تعیین سود است. خط‌مشی (تعداد روزنامه‌های خریداری شده) با انتخاب مقادیر دیگر تغییر داده می‌شود تا جایی که سود در سطوح پیش و پس از آن کاهش یابد. مقدار میانی، تعداد بهینهٔ روزنامه‌هایی است که روزنامه‌فروش باید خریداری کند.

جدول شبیه‌سازی برای خرید ۷۰ روزنامه در جدول ۲-۱۸ نشان داده شده است. در روز ۱ تقاضا برای روزنامه ۶۰ و درامد ناشی از فروش ۶۰ روزنامه ۱۲۰۰ واحد پول است. در پایان روز، ۱۰ روزنامه باقی می‌ماند. درامد ناشی از فروش به قیمت باطله ۲۰ واحد پول و از قرار هر نسخه ۲ واحد پول است. سود روز اول به شرح زیر تعیین می‌شود: ۷۰×۱۳

سود روز اول

$$\text{سود} = ۱۲۰۰ - ۹۱۰ - ۰ + ۲۰ = ۳۱۰$$

در روز پنجم تقاضا بیش از عرضه است. درامد ناشی از فروش ۱۴۰۰ واحد پول است زیرا با خط‌مشی جاری تنها ۷۰ نسخه روزنامه وجود دارد. چون ۲۰ روزنامهٔ دیگر هم می‌توانست فروخته

جدول ۲-۱۶ تخصیص ارقام تصادفی برای تعیین نوع روز.

نوع روز	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
خوب	۰٫۳۵	۰٫۳۵	۰۱-۳۵
متوسط	۰٫۴۵	۰٫۸۰	۳۶-۸۰
بد	۰٫۲۰	۱٫۰۰	۸۱-۰۰

جدول ۲-۱۷ تخصیص ارقام تصادفی برای روزنامه‌های مورد تقاضا.

تقاضا	توزیع تجمعی: خوب	متوسط	بد	تخصیص اعداد تصادفی: خوب	متوسط	بد
۴۰	۰٫۰۳	۰٫۱۰	۰٫۴۴	۰۱-۰۳	۰۱-۱۰	۰۱-۴۴
۵۰	۰٫۰۸	۰٫۲۸	۰٫۶۶	۰۴-۰۸	۱۱-۲۸	۴۵-۶۶
۶۰	۰٫۲۳	۰٫۶۸	۰٫۸۲	۰۹-۲۳	۲۹-۶۸	۶۷-۸۲
۷۰	۰٫۴۳	۰٫۸۸	۰٫۹۴	۲۴-۴۳	۶۹-۸۸	۸۳-۹۴
۸۰	۰٫۷۸	۰٫۹۶	۱٫۰۰	۴۴-۷۸	۸۹-۹۶	۹۵-۰۰
۹۰	۰٫۹۳	۱٫۰۰	۱٫۰۰	۷۹-۹۳	۹۷-۰۰	
۱۰۰	۱٫۰۰	۱٫۰۰	۱٫۰۰	۹۴-۰۰		

جدول ۲-۱۸ شبیه‌سازی برای خرید ۷۰ روزنامه.

روز	ارقام تصادفی برای تعیین نوع روز	نوع روز	ارقام تصادفی برای تقاضا	تقاضا	درامد حاصل از فروش	سود از دست رفته به‌خاطر فزونی تقاضا	درامد ناشی از فروش به قیمت باطله	سود روزانه
۱	۹۴	بد	۸۰	۶۰+۲۰ =	۱۲۰۰	–	۲۰=(۷۰-۶۰)×۲	۳۱۰
۲	۷۷	متوسط	۲۰	۵۰	۱۰۰۰	–	۴۰	۱۳۰
۳	۴۹	متوسط	۱۵	۵۰	۱۰۰۰	–	۴۰	۱۳۰
۴	۴۵	متوسط	۸۸	۷۰	۱۴۰۰	–	–	۴۹۰
۵	۴۳	متوسط	۹۸	۹۰	۱۴۰۰	۱۴۰	–	۳۵۰
۶	۳۲	خوب	۶۵	۸۰	۱۴۰۰	۷۰	–	۴۲۰
۷	۴۹	متوسط	۸۶	۷۰	۱۴۰۰	–	–	۴۹۰
۸	۰۰	بد	۷۳	۶۰	۱۲۰۰	–	۲۰	۳۱۰
۹	۱۶	خوب	۲۴	۷۰	۱۴۰۰	–	–	۴۹۰
۱۰	۲۴	خوب	۶۰	۸۰	۱۴۰۰	۷۰	–	۴۲۰
۱۱	۳۱	خوب	۶۰	۸۰	۱۴۰۰	۷۰	–	۴۲۰
۱۲	۱۴	خوب	۲۹	۷۰	۱۴۰۰	–	–	۴۹۰
۱۳	۴۱	متوسط	۱۸	۵۰	۱۰۰۰	–	۴۰	۱۳۰
۱۴	۶۱	متوسط	۹۰	۸۰	۱۴۰۰	۷۰	–	۴۲۰
۱۵	۸۵	بد	۹۳	۷۰	۱۴۰۰	–	–	۴۹۰
۱۶	۰۸	خوب	۷۳	۸۰	۱۴۰۰	۷۰	–	۴۲۰
۱۷	۱۵	خوب	۲۱	۶۰	۱۲۰۰	–	۲۰	۳۱۰
۱۸	۹۷	بد	۴۵	۵۰	۱۰۰۰	–	۴۰	۱۳۰
۱۹	۵۲	متوسط	۷۶	۷۰	۱۴۰۰	–	–	۴۹۰
۲۰	۷۸	متوسط	۹۶	۸۰	۱۴۰۰	۷۰	–	۴۲۰
					۲۵۸۰۰	۵۶۰	۲۲۰	۷۲۶۰

شود، سود ازدست‌رفته معادل ۱۴۰ واحد پول (۷ × ۲۰) ارزیابی می‌شود. سود روزانه به شرح زیر تعیین می‌شود:

سود = ۱۴۰۰ − ۹۱۰ − ۱۴۰ + ۰ = ۳۵۰ ← سود روز پنجم

سود دورهٔ ۲۰ روزه برابر با جمع سودهای روزانه، یعنی ۷۲۶۰ واحد پول است. این مبلغ را به شرح زیر و بر اساس جمعهای به‌دست آمده از ۲۰ روز شبیه‌سازی نیز می‌توان یافت:

سود کل = ۲۵۸۰۰ − ۱۸۲۰۰ − ۵۶۰ + ۲۲۰ = ۷۲۶۰

تعیین تعداد بهینهٔ روزنامه‌هایی که باید خرید به‌عنوان تمرین به خواننده واگذار می‌شود. ■

■ مثال ۲-۴ شبیه‌سازی سیستم موجودی (M, N)

این مثال از الگوی سیستم موجودی احتمالی سطح سفارش پیروی می‌کند که در شکل ۲-۷ نشان داده شده است. فرض کنید که بالاترین سطح موجودی، M، ۱۱ واحد و دورهٔ بررسی، N، ۵ روز باشد. مسأله در مورد براورد متوسط واحدهای مانده در انبار در پایان روز و تعداد روزهایی که شرایط کمبود در آنها وجود داشته، از طریق شبیه‌سازی است. توزیع تعداد واحدهای مورد تقاضا در روز، در جدول ۲-۱۹ نشان داده شده است. در این مثال، مهلت تحویل متغیری تصادفی است که در جدول ۲-۲۰ نشان داده شده است. فرض کنید که سفارشها در پایان روز صادر و در چارچوب تعیین شده توسط مهلت تحویل، در ابتدای روز وارد می‌شوند.

برای براورد میانگین واحدهای مانده در انبار در پایان روز، دوره‌های بسیاری باید شبیه‌سازی شود. برای این مثال تنها پنج دوره نشان داده خواهد شد. در پایان این فصل، به‌عنوان تمرین از دانشجو خواسته می‌شود که مثال را ادامه دهد.

تخصیص ارقام تصادفی برای تقاضای روزانه و مهلت تحویل در آخرین ستون سمت چپ جدولهای ۲-۱۹ و ۲-۲۰ و جدول شبیه‌سازی به‌دست آمده نیز در جدول ۲-۲۱ نشان داده شده است. شبیه‌سازی تحت شرایطی آغاز شده که سطح موجودی ۳ واحد بوده و ورود یک سفارش ۸ واحدی در مدت دو روز برنامه‌ریزی شده بوده است.

جدول ۲-۱۹ تخصیص ارقام تصادفی برای تقاضای روزانه.

تقاضا	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۰	۰٫۱۰	۰٫۱۰	۰۱-۱۰
۱	۰٫۲۵	۰٫۳۵	۱۱-۳۵
۲	۰٫۳۵	۰٫۷۰	۳۶-۷۰
۳	۰٫۲۱	۰٫۹۱	۷۱-۹۱
۴	۰٫۰۹	۱٫۰۰	۹۲-۰۰

جدول ۲-۲۰ تخصیص ارقام تصادفی برای مهلت تحویل.

مهلت تحویل (روز)	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۱	۰٫۶	۰٫۶	۱-۶
۲	۰٫۳	۰٫۹	۷-۹
۳	۰٫۱	۱٫۰	۰

جدول ۲-۲۱ جدول شبیه‌سازی برای سیستم موجودی (M, N).

دور	روز	موجودی در ابتدای روز	ارقام تصادفی برای تقاضا	تقاضا	موجودی در انتهای روز	مقدار کمبود	مقدار سفارش	ارقام تصادفی برای مهلت تحویل	روزهای مانده تا ورود سفارش
۱	۱	۳	۲۴	۱	۲	۰	-	-	۱
	۲	۲	۳۵	۱	۱	۰	-	-	۰
	۳	۹	۶۵	۲	۷	۰	-	-	-
	۴	۷	۸۱	۳	۴	۰	-	-	-
	۵	۴	۵۴	۲	۲	۰	۹	۵	-
۲	۱	۲	۰۳	۰	۲	۰	-	-	۰
	۲	۱۱	۸۷	۳	۸	۰	-	-	-
	۳	۸	۲۷	۱	۷	۰	-	-	-
	۴	۷	۷۳	۳	۴	۰	-	-	-
	۵	۴	۷۰	۲	۲	۰	۹	۰	۳
۳	۱	۲	۴۷	۲	۰	۰	-	-	۲
	۲	۰	۴۵	۲	۰	۲	-	-	۱
	۳	۰	۴۸	۲	۰	۴	-	-	۰
	۴	۹	۱۷	۱	۴	۰	-	-	-
	۵	۴	۰۹	۰	۴	۰	۷	۳	۱
۴	۱	۴	۴۲	۲	۲	۰	-	-	۰
	۲	۹	۸۷	۳	۶	۰	-	-	-
	۳	۶	۲۶	۱	۵	۰	-	-	-
	۴	۵	۳۶	۲	۳	۰	-	-	-
	۵	۳	۴۰	۲	۱	۰	۱۰	۴	۱
۵	۱	۱	۰۷	۰	۱	۰	-	-	۰
	۲	۱۱	۶۳	۲	۹	۰	-	-	-
	۳	۹	۱۹	۱	۸	۰	-	-	-
	۴	۸	۸۸	۳	۵	۰	-	-	-
	۵	۵	۹۳	۴	۱	۰	۱۰	۸	۲
					۸۷				

بررسی جدول شبیه‌سازی برای چند روز منتخب، نحوهٔ عمل فرایند را نشان می‌دهد. سفارش مربوط به ۸ واحد، در صبح روز سوم از دورهٔ اول وارد می‌شود و سطح موجودی را از ۱ به ۹ واحد افزایش می‌دهد. تقاضا در خلال بقیهٔ دورهٔ اول، موجودی پایان روز را به ۲ واحد در روز پنجم

کاهش می‌دهد. بنابراین، سفارشی برای ۹ واحد صادر می‌شود. مهلت تحویل برای این سفارش، یک روز است. سفارش ۹ واحدی در صبح روز ۲ از دور ۲ به موجودی افزوده می‌شود.

توجه کنید که موجودی در ابتدای روز دوم از دور سوم صفر است. تقاضایی به میزان ۲ واحد در این روز، به ایجاد کمبود می‌انجامد که این واحدها در این روز و روز پس از آن نیز در شمار واحدهای سفارش تحویل‌نشده قرار می‌گیرند. در صبح روز ۴ از دور ۳، میزان موجودی در آغاز روز ۹ واحد است. ۴ واحد سفارش تحویل‌نشده و یک واحد مورد تقاضا در این روز، موجودی در پایان روز را به ۴ واحد کاهش می‌دهد.

بر اساس پنج دور شبیه‌سازی، متوسط موجودی در پایان روز، تقریباً ۳٫۵ واحد (۲۵ ÷ ۸۷) است. در دو روز از ۲۵ روز، شرایط کمبود وجود دارد. ■

۲-۳ دیگر مثالهای شبیه‌سازی

این بخش شامل مثالهایی از شبیه‌سازی در زمینهٔ پایایی، یک مأموریت بمباران و تولید توزیع احتمال تقاضا در مهلت تحویل بر اساس توزیعهای تقاضا و مهلت تحویل است.

■ **مثال ۲-۵ مسألهٔ پایایی**

یک ماشین فرز بزرگ، سه برینگ مختلف دارد که در جریان کار دچار خرابی می‌شوند. تابع توزیع تجمعی عمر برینگها یکسان و به شرح جدول ۲-۲۲ است. هرگاه برینگی خراب شود، فرز از کار می‌افتد،

جدول ۲-۲۲ توزیع عمر برینگ.

عمر برینگ (ساعت)	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۱۰۰۰	۰٫۱۰	۰٫۱۰	۰۱-۱۰
۱۱۰۰	۰٫۱۳	۰٫۲۳	۱۱-۲۳
۱۲۰۰	۰٫۲۵	۰٫۴۸	۲۴-۴۸
۱۳۰۰	۰٫۱۳	۰٫۶۱	۴۹-۶۱
۱۴۰۰	۰٫۰۹	۰٫۷۰	۶۲-۷۰
۱۵۰۰	۰٫۱۲	۰٫۸۲	۷۱-۸۲
۱۶۰۰	۰٫۰۲	۰٫۸۴	۸۳-۸۴
۱۷۰۰	۰٫۰۶	۰٫۹۰	۸۵-۹۰
۱۸۰۰	۰٫۰۵	۰٫۹۵	۹۱-۹۵
۱۹۰۰	۰٫۰۵	۱٫۰۰	۹۶-۰۰

www.matlabdl.com

جدول ۲-۲۳ توزیع مدت تأخیر.

مدت تأخیر (دقیقه)	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۵	۰٫۶	۰٫۶	۶-۱
۱۰	۰٫۳	۰٫۹	۹-۷
۱۵	۰٫۱	۱٫۰	۰

تعمیرکاری احضار می‌شود و برینگ تازه‌ای نصب می‌شود. مدت تأخیر تعمیرکار در ورود به محل ماشین فرز نیز متغیری تصادفی با توزیع ارائه شده در جدول ۲-۲۳ است. هزینهٔ مدت از کارماندگی فرز از قرار دقیقه‌ای ۵ واحد پول براورد می‌شود. هزینهٔ مستقیم دستمزد تعمیرکار، ساعتی ۱۲ واحد پول است. تعویض یک برینگ به مصرف ۲۰ دقیقه وقت، تعویض دو برینگ به مصرف ۳۰ دقیقه وقت و تعویض سه برینگ به مصرف ۴۰ دقیقه وقت نیاز دارد. هزینهٔ هر برینگ نیز ۱۶ واحد پول است. پیشنهادی ارائه شده است مبنی بر اینکه هرگاه یک برینگ خراب شود، هر سه برینگ تعویض شوند. مدیریت نیاز به ارزیابی این پیشنهاد دارد.

جدول ۲-۲۴ نتیجهٔ شبیه‌سازی ۲۰٫۰۰۰ ساعت کار سیستم طبق خط‌مشی جاری را ارائه می‌کند. توجه کنید که مواردی وجود دارد که در یک زمان بیش از یک برینگ خراب می‌شود. رخداد چنین حالتی در عمل نامحتمل است و در اینجا ناشی از فاصله‌های نسبتاً زیاد ۱۰۰ ساعته است. در این مثال فرض بر آن است که مدتها هیچ‌گاه یکسان نیستند و به این ترتیب، در هر خرابی، بیش از یک برینگ تعویض نمی‌شود. برای برینگهای ۱ و ۲ شانزده تعویض ولی برای برینگ ۳ تنها ۱۴ تعویض صورت گرفته است. هزینهٔ سیستم فعلی به شرح زیر براورد می‌شود

هزینهٔ برینگها = ۴۶ برینگ × ۱۶ واحد پول برای هر برینگ = ۷۳۶

هزینهٔ مدت تأخیر = (۹۵ + ۱۲۵ + ۱۱۰)دقیقه × ۵ واحد پول در دقیقه = ۱۶۵۰

هزینهٔ مدت از کارماندگی حین تعمیر = ۴۶ برینگ × ۲۰ دقیقه برای هر برینگ × ۵ واحد پول در دقیقه

= ۴۶۰۰

هزینهٔ تعمیرکار = ۴۶ برینگ × ۲۰ دقیقه برای هر برینگ × ۱۲ واحد پول در ۶۰ دقیقه = ۱۸۴

هزینهٔ کل = ۷۳۶ + ۱۶۵۰ + ۴۶۰۰ + ۱۸۴ = ۷۱۷۰

جدول ۲-۲۵ شبیه‌سازی با استفاده از روش پیشنهادی است. توجه کنید که تا جایی که ممکن است عمر یکسانی برای هر سه برینگ ظاهر می‌شود. فرض بر این است که برینگها به‌ترتیب در قفسه‌ای قرار دارند و به دنبال یکدیگر برداشته و روی فرز نصب می‌شوند. (تأثیر استفاده از اعداد تصادفی مختلف در برابر استفاده از اعداد تصادفی مشترک، در فصل ۱۲ شرح داده شده است.)

جدول ۲-۲۴ تعویض برینگ با استفاده از روش جاری.

	برینگ ۱					برینگ ۲					برینگ ۳				
	ارقام تصادفی	عمر (ساعت)	عمر تجمعی (ساعت)	ارقام تصادفی	تأخیر (دقیقه)	ارقام تصادفی	عمر (ساعت)	عمر تجمعی (ساعت)	ارقام تصادفی	تأخیر (دقیقه)	ارقام تصادفی	عمر (ساعت)	عمر تجمعی (ساعت)	ارقام تصادفی	تأخیر (دقیقه)
۱	۶۷	۱۴۰۰	۱۴۰۰	۲	۵	۷۱	۱۵۰۰	۱۵۰۰	۰	۱۵	۷۶	۱۵۰۰	۱۵۰۰	۰	۱۵
۲	۰۸	۱۰۰۰	۲۴۰۰	۳	۵	۲۳	۱۲۰۰	۲۷۰۰	۷	۱۰	۶۵	۱۴۰۰	۲۹۰۰	۲	۵
۳	۴۹	۱۳۰۰	۳۷۰۰	۱	۵	۸۶	۱۷۰۰	۴۴۰۰	۳	۵	۶۱	۱۴۰۰	۴۳۰۰	۷	۱۰
۴	۸۴	۱۶۰۰	۵۳۰۰	۷	۱۰	۹۳	۱۸۰۰	۶۲۰۰	۱	۵	۹۶	۱۹۰۰	۶۲۰۰	۱	۵
۵	۴۴	۱۲۰۰	۶۵۰۰	۸	۱۰	۸۱	۱۶۰۰	۷۸۰۰	۲	۵	۶۵	۱۴۰۰	۷۶۰۰	۳	۵
۶	۳۰	۱۲۰۰	۷۷۰۰	۱	۵	۴۴	۱۲۰۰	۹۰۰۰	۸	۱۰	۵۶	۱۳۰۰	۸۹۰۰	۳	۵
۷	۱۰	۱۰۰۰	۸۷۰۰	۲	۵	۱۹	۱۱۰۰	۱۰۱۰۰	۱	۵	۱۱	۱۱۰۰	۱۰۰۰۰	۶	۵
۸	۶۳	۱۴۰۰	۱۰۱۰۰	۸	۱۰	۵۱	۱۳۰۰	۱۱۴۰۰	۱	۵	۸۶	۱۷۰۰	۱۱۷۰۰	۳	۵
۹	۰۲	۱۰۰۰	۱۱۱۰۰	۳	۵	۴۵	۱۳۰۰	۱۲۷۰۰	۷	۱۰	۵۷	۱۳۰۰	۱۳۰۰۰	۱	۵
۱۰	۰۲	۱۰۰۰	۱۲۱۰۰	۸	۱۰	۱۲	۱۱۰۰	۱۳۸۰۰	۸	۵	۴۹	۱۳۰۰	۱۴۳۰۰	۲	۵
۱۱	۷۷	۱۵۰۰	۱۳۶۰۰	۷	۱۰	۴۸	۱۳۰۰	۱۵۱۰۰	۰	۱۵	۳۶	۱۲۰۰	۱۵۵۰۰	۸	۱۰
۱۲	۵۹	۱۳۰۰	۱۴۹۰۰	۵	۵	۰۹	۱۰۰۰	۱۶۱۰۰	۸	۱۰	۴۴	۱۲۰۰	۱۶۷۰۰	۲	۵
۱۳	۲۳	۱۱۰۰	۱۶۰۰۰	۵	۵	۴۴	۱۲۰۰	۱۷۳۰۰	۱	۵	۹۴	۱۸۰۰	۱۸۵۰۰	۱	۵
۱۴	۵۳	۱۳۰۰	۱۷۳۰۰	۹	۱۰	۴۶	۱۲۰۰	۱۸۵۰۰	۲	۵	۸۷	۱۵۰۰	۲۰۰۰۰	۷	۱۰
۱۵	۸۵	۱۷۰۰	۱۹۰۰۰	۶	۵	۴۰	۱۲۰۰	۱۹۷۰۰	۸	۱۰					
۱۶	۷۵	۱۵۰۰	۲۰۵۰۰	۴	۵	۵۲	۱۳۰۰	۲۱۰۰۰	۵	۵					
					۱۱۰					۱۲۵					۹۵

۱۶ بار برینگ ۱ عوض شده

۱۶ بار برینگ ۲ عوض شد

۱۴ بار برینگ ۳ عوض شده

جدول ۲-۲۵ تعویض برینگ با استفاده از روش پیشنهادی.

	عمر برینگ ۱ (ساعت)	عمر برینگ ۲ (ساعت)	عمر برینگ ۳ (ساعت)	اولین خرابی (ساعت)	عمر تجمعی (ساعت)	ارقام تصادفی	تأخیر (دقیقه)
۱	۱۴۰۰	۱۵۰۰	۱۵۰۰	۱۴۰۰	۱۴۰۰	۳	۵
۲	۱۰۰۰	۱۲۰۰	۱۴۰۰	۱۰۰۰	۲۴۰۰	۷	۱۰
۳	۱۳۰۰	۱۷۰۰	۱۴۰۰	۱۳۰۰	۳۷۰۰	۵	۵
۴	۱۶۰۰	۱۸۰۰	۱۹۰۰	۱۶۰۰	۵۳۰۰	۱	۵
۵	۱۲۰۰	۱۶۰۰	۱۴۰۰	۱۲۰۰	۶۵۰۰	۴	۵
۶	۱۲۰۰	۱۲۰۰	۱۳۰۰	۱۲۰۰	۷۷۰۰	۳	۵
۷	۱۰۰۰	۱۱۰۰	۱۱۰۰	۱۰۰۰	۸۷۰۰	۷	۱۰
۸	۱۴۰۰	۱۳۰۰	۱۷۰۰	۱۳۰۰	۱۰۰۰۰	۸	۱۰
۹	۱۰۰۰	۱۳۰۰	۱۳۰۰	۱۰۰۰	۱۱۰۰۰	۸	۱۰
۱۰	۱۰۰۰	۱۱۰۰	۱۳۰۰	۱۰۰۰	۱۲۰۰۰	۳	۵
۱۱	۱۵۰۰	۱۳۰۰	۱۲۰۰	۱۲۰۰	۱۳۲۰۰	۲	۵
۱۲	۱۳۰۰	۱۰۰۰	۱۲۰۰	۱۰۰۰	۱۴۲۰۰	۴	۵
۱۳	۱۱۰۰	۱۲۰۰	۱۸۰۰	۱۱۰۰	۱۵۳۰۰	۱	۵
۱۴	۱۳۰۰	۱۲۰۰	۱۵۰۰	۱۲۰۰	۱۶۵۰۰	۶	۵
۱۵	۱۷۰۰	۱۲۰۰	(۶۳) ۱۴۰۰	۱۲۰۰	۱۷۷۰۰	۲	۵
۱۶	۱۵۰۰	۱۳۰۰	(۲۱) ۱۱۰۰	۱۱۰۰	۱۸۸۰۰	۷	۱۰
۱۷	(۸۵) ۱۷۰۰	(۵۳) ۱۳۰۰	(۲۳) ۱۱۰۰	۱۱۰۰	۱۹۹۰۰	۰	۱۵
۱۸	(۰۵) ۱۰۰۰	(۲۹) ۱۲۰۰	(۵۱) ۱۳۰۰	۱۰۰۰	۲۰۹۰۰	۵	۵
							۱۲۵

ارقام تصادفی که به تعیین عمر برینگهای افزوده انجامیده‌اند از پانزدهمین تعویض برینگ ۳ به بعد در داخل پرانتز نشان داده شده‌اند. هرگاه از خط‌مشی تازه استفاده شده، به ۱۸ دست برینگ نیاز بوده است. در دو شبیه‌سازی، تأخیرهای تعمیرکار دوباره به‌کار نرفته بلکه به‌طور مستقل دوباره تولید شده است. هزینهٔ کل خط‌مشی تازه به شرح زیر محاسبه می‌شود

هزینهٔ برینگها = ۵۴ برینگ × ۱۶ واحد پول به ازای هر برینگ = ۸۶۴

هزینهٔ مدت تأخیر = ۱۲۵ دقیقه × ۵ واحد پول در دقیقه = ۶۲۵

هزینهٔ مدت ازکارماندگی حین تعمیر = ۱۸ دست × ۴۰ دقیقه برای هر دست × ۵ واحد پول در دقیقه = ۳۶۰۰

هزینهٔ تعمیرکار = ۱۸ دست × ۴۰ دقیقه برای هر دست × ۱۲ واحد پول در ۶۰ دقیقه = ۱۴۴

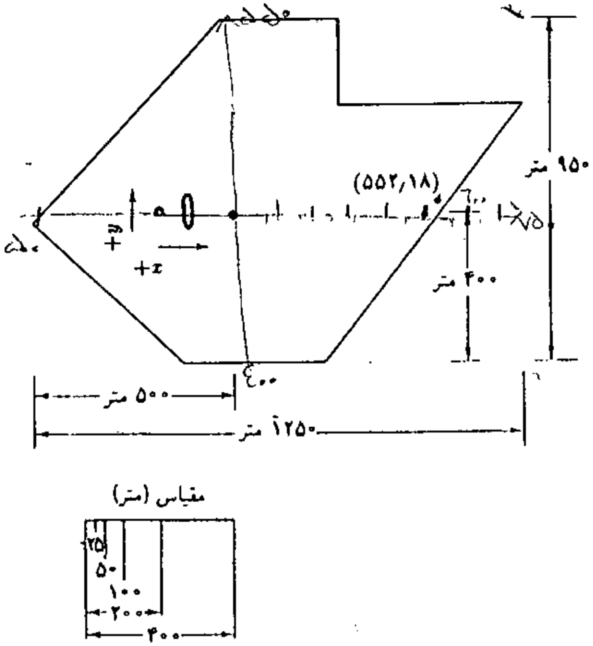
هزینهٔ کل = ۸۶۴ + ۶۲۵ + ۳۶۰۰ + ۱۴۴ = ۵۲۳۳

۷۱۷۰ − ۵۲۳۳ = ۱۹۳۷

در شبیه‌سازی ۲۰۰۰۰ ساعته، با به‌کارگیری خط‌مشی تازه، معادل ۱۹۳۷ واحد پول صرفه‌جویی می‌شود. اگر ماشین به‌طور مداوم کار کند شبیه‌سازی تقریباً $2\frac{1}{4}$ سال را دربر می‌گیرد. بنابراین، صرفه‌جویی سالانه حدود ۸۶۰ واحد پول می‌شود. ■

■ **مثال ۲-۶ اعداد تصادفی نرمال**

یک مسأله قدیمی شبیه‌سازی، مربوط به یک اسکادران بمب‌افکن است که سعی در نابودسازی یک زاغهٔ مهمات دارد که به‌صورت نشان داده شده در شکل ۲-۸ است. اگر بمبی در هر نقطهٔ زاغه فرود آید موفقیت و در غیر این‌صورت شکست حاصل شده است. هواپیما در جهت افقی پرواز می‌کند. بمبهای پرتاب شده در جهت افقی، با انحراف معیار ۶۰۰ متر به زمین می‌افتند. در جهت عمودی، نیز با انحراف معیار ۳۰۰ متر می‌افتند. هر اسکادران ده بمب‌افکن دارد. نقطه‌ای که نشانه‌روی می‌شود، نقطه‌ای واقع در قلب زاغهٔ مهمات است. هدف مسألهٔ شبیه‌سازی عملیات و اظهارنظر در مورد تعداد بمبهای به هدف نشسته است.



شکل ۲-۸ زاغهٔ مهمات.

به یاد دارید که مقدار تصادفی نرمال استاندارد، Z، به‌صورت

$$Z = \frac{X - \mu}{\sigma}$$

مقدار تصادفی نرمال استاندارد — میانگین واقعی توزیع X — انحراف معیار X

توزیع می‌شود که در آن X یک متغیر تصادفی نرمال، μ میانگین واقعی توزیع X، و σ انحراف معیار X است. بنابراین،

$$X = Z\sigma + \mu$$

متغیر تصادفی نرمال — نقطهٔ هدفگیری

نقطهٔ هدفگیری شده می‌تواند در این مثال نقطهٔ $(\circ, \circ)$ در نظر گرفته شود، یعنی مقدار μ در جهت افقی و مقدار μ در جهت عمودی صفر است. بنابراین،

$$X = Z\sigma_X$$
$$Y = Z\sigma_Y$$

مختصات شبیه‌سازی محل اصابت بمب

که (X, Y) مختصات شبیه‌سازی شدهٔ محل اصابت بمب است. حالا، $\sigma_X = ۶۰۰$ و $\sigma_Y = ۳۰۰$، پس

$$X = ۶۰۰ Z_i$$
$$Y = ۳۰۰ Z_j$$

اعداد تصادفی نرمال

اندیسهای i و j به این دلیل افزوده شده است که متفاوت بودن مقادیر Z را نشان دهد. این مقادیر Z چیستند و کجا می‌توان آنها را یافت؟ مقادیر Z اعداد تصادفی نرمال‌اند. می‌توان از اعداد تصادفی با توزیع یکنواخت، این مقادیر را به شرح مطالب فصل ۸ تولید کرد. راهی دیگر، به‌کارگیری جدولهای اعداد تصادفی نرمال تولید شده است که نمونهٔ کوچکی از آن در جدول پ-۲ ارائه شده است.

به منظور درک اینکه در این مأموریتهای بمباران چه می‌گذرد، یک شبیه‌سازی با شاید ۱۰ یا ۲۰ مأموریت می‌توان انجام داد. اما، محدودیت جا اجازهٔ چنین شبیه‌سازی گسترده‌ای را نمی‌دهد. مثالی در مورد یک مأموریت نشان می‌دهد که چنین شبیه‌سازیهایی چگونه انجام می‌شود. جدول اعداد تصادفی نرمال را همانند جدول اعداد تصادفی بکار می‌گیریم. یعنی کاملاً از جایی تصادفی در جدول شروع می‌کنیم و بدون استفادهٔ دوباره از اعداد به‌طور منظم در مسیری جلو می‌رویم. جدول ۲-۲۶ نتایج شبیه‌سازی یک مأموریت را نشان می‌دهد. علامت اختصاری RNN_x معرف «عدد تصادفی نرمال برای محاسبهٔ مختصهٔ x» و نظیر Z_i است. اولین عدد تصادفی نرمال مورد استفاده، ۰٫۸۴- بود که مختصهٔ x مساوی با $-۵۰۴ = (-۰٫۸۴)۶۰۰$ تولید کرد. عدد تصادفی نرمال برای تولید مختصهٔ y، ۰٫۶۶ بود که به مختصهٔ y برابر ۱۹۸ انجامید. در نظر گرفتن این دو عدد با هم، یعنی $(-۵۰۴, ۱۹۸)$، معرف شکست است. زیرا نقطه‌ای بیرون از هدف را

مثال از مونت کارلو یا شبیه‌سازی ایستا زیرا زمان عنصری موثر در حل مساله نیست

عدد تصادفی نرمال برای محاسبه‌ی مختصه‌ی x و نظیر Z_i

عدد تصادفی نرمال برای مختصه‌ی y و نظیر Z_j

جدول ۲-۲۶ شبیه‌سازی مأموریت بمباران.

بمب‌افکن	RNN_x (Z_i)	مختصهٔ x $(۶۰۰ RNN_x)$	RNN_y (Z_j)	مختصهٔ y $(۳۰۰ RNN_y)$	نتیجه[الف]
۱	−۰٫۸۴	−۵۰۴	۰٫۶۶	۱۹۸	به خطا رفته
۲	۱٫۰۳	۶۱۸	−۰٫۱۳	−۳۹	به خطا رفته
۳	۰٫۹۲	۵۵۲	۰٫۰۶	۱۸	اصابت کرده
۴	−۱٫۸۲	−۱۰۹۲	−۱٫۴۰	−۴۲۰	به خطا رفته
۵	−۰٫۱۶	−۹۶	۰٫۲۳	۶۹	اصابت کرده
۶	−۱٫۷۸	−۱۰۶۸	۱٫۳۳	۳۹۹	به خطا رفته
۷	۲٫۰۴	۱۲۲۴	۰٫۶۹	۲۰۷	به خطا رفته
۸	۱٫۰۸	۶۴۸	−۱٫۱۰	−۳۳۰	به خطا رفته
۹	−۱٫۵۰	−۹۰۰	−۰٫۷۲	−۲۱۶	به خطا رفته
۱۰	−۰٫۴۲	−۲۵۲	−۰٫۶۰	−۱۸۰	اصابت کرده

الف) ۳ بمب اصابت کرده، ۷ بمب به خطا رفته.

نشان می‌دهد. این نقطه و نقطهٔ مربوط به بمب‌افکن سوم را روی شکل ۲-۸ مشخص کرده‌ایم. ده بمب‌افکن، سه موفقیت و هفت شکست داشته‌اند. به منظور ارزیابی توان نابودسازی زاغه به مأموریتهای بسیار بیشتری نیاز است. مأموریتهای اضافی را به‌صورت تمرینی برای خواننده در نظر گرفته‌ایم. این، مثالی از مونت‌کارلو، یا شبیه‌سازی ایستاست زیرا زمان عنصری مؤثر در حل مسأله نیست. ■

■ مثال ۲-۷ تقاضا در مهلت تحویل

تقاضا در مهلت تحویل ممکن است در سیستم موجودی‌ای مطرح شود که مهلت تحویل در آن غیر آنی باشد. مهلت تحویل مدتی است که از صدور یک سفارش تا دریافت آن به درازا می‌کشد. در وضعیتهای واقعی، مهلت تحویل متغیری تصادفی است. در مهلت تحویل، تقاضا نیز به‌صورت تصادفی رخ می‌نماید. بدین ترتیب، تقاضا در مهلت تحویل، متغیری است تصادفی که به‌صورت جمع تقاضا در جریان مهلت تحویل، با $\sum_{i=0}^{T} D_i$ تعریف می‌شود؛ که i دورهٔ زمانی، $i = ۰, ۱, ۲, \ldots$؛ D_i تقاضا در iامین دورهٔ زمانی؛ و T مهلت تحویل است. توزیع تقاضا در مهلت تحویل، با شبیه‌سازی دوره‌های فراوانی از مهلت تحویل و ایجاد هیستوگرمی از نتایج تعیین می‌شود.

جمع تقاضا در جریان مهلت تحویل

شرکتی در کار فروش کاغذ روزنامه به‌صورت تویی است. تقاضای روزانه با توزیع احتمال زیر

ارائه می‌شود:

تقاضای روزانه (توپ)	۳	۴	۵	۶
احتمال	۰/۲۰	۰/۳۵	۰/۳۰	۰/۱۵

مهلت تحویل عبارت از تعداد روزهای از صدور سفارش تا دریافت آن از فروشنده توسط شرکت است. در این مورد، مهلت تحویل متغیری تصادفی است که با توزیع زیر ارائه می‌شود:

مدت تحویل (روز)	۱	۲	۳
احتمال	۰/۳۶	۰/۴۲	۰/۲۲

جدول ۲-۲۷ تخصیص ارقام تصادفی برای تقاضا و جدول ۲-۲۸ نیز تخصیص ارقام تصادفی برای مهلت تحویل را نشان می‌دهد. جدول ناتمام شبیه‌سازی را در جدول ۲-۲۹ نشان داده‌ایم. ارقام تصادفی برای دورهٔ اول ۵۷ بودند که مهلت تحویل ۲ روزه را تولید می‌کنند. بدین ترتیب، برای تقاضای روزانه باید دو زوج از ارقام تصادفی تولید کرد. زوج اول از این دو ۸۷ است و به تقاضایی معادل ۶ می‌انجامد و یک تقاضای ۴ به‌دنبال آن می‌آید. مهلت تحویل برای دورهٔ اول ۱۰ است. پس از آنکه دوره‌های بسیاری شبیه‌سازی شوند، هیستوگرمی تعریف می‌شود که ممکن است همانند شکل ۲-۹ نمایان شود. این مثال نشان می‌دهد که چگونه شبیه‌سازی را می‌توان با تولید یک نمونهٔ تصادفی از یک توزیع نامعلوم، در بررسی توزیع مورد استفاده قرار داد.

جدول ۲-۲۷ تخصیص ارقام تصادفی برای تقاضا.

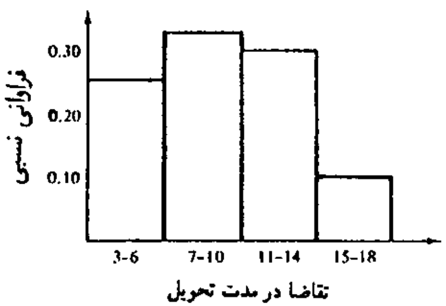
تقاضای روزانه	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۳	۰/۲۰	۰/۲۰	۰۱-۲۰
۴	۰/۳۵	۰/۵۵	۲۱-۵۵
۵	۰/۳۰	۰/۸۵	۵۶-۸۵
۶	۰/۱۵	۱/۰۰	۸۶-۰۰

جدول ۲-۲۸ تخصیص ارقام تصادفی برای مهلت تحویل.

مهلت تحویل (روز)	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۱	۰/۳۶	۰/۳۶	۰۱-۳۶
۲	۰/۴۲	۰/۷۸	۳۷-۷۸
۳	۰/۲۲	۱/۰۰	۷۹-۰۰

جدول ۲-۲۹ جدول شبیه‌سازی برای تقاضا در مهلت تحویل.

دور	ارقام تصادفی برای مهلت تحویل	مهلت تحویل (روز)	ارقام تصادفی برای تقاضا	تقاضا	تقاضا در مهلت تحویل
۱	۵۷	۲	۸۷	۶	
			۳۴	۴	۱۰
۲	۳۳	۱	۸۲	۵	۵
۳	۹۳	۳	۲۸	۴	
			۱۹	۳	
			۶۳	۵	۱۲
۴	۵۵	۲	۹۱	۶	
			۲۶	۴	۱۰
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.



شکل ۲-۹ هیستوگرام برای تقاضا در مهلت تحویل.

۲-۴ خلاصه

مقصود از این فصل، معرفی مفاهیم شبیه‌سازی از طریق مثالهایی به منظور نمایش زمینه‌های عمومی کاربرد و ایجاد انگیزه برای مطالب فصلهای دیگر است. با قرار دادن مثالها در اوایل کتاب، آگاهی خواننده از مفاهیم، روشها و راههای تحلیل در فصلهای آینده تقویت خواهد شد.

از جدولهـای شبیه‌سازی خلق‌الساعـه برای کامل کردن هر مثال استفاده کردیم. پیشامدهـای

www.matlabdl.com

مندرج در هر جدول با استفاده از اعداد تصادفی دارای توزیع یکنواخت و در یک مورد نیز با استفاده از اعداد تصادفی نرمال تولید شدند. نیاز به تعیین ویژگیهای داده‌های ورودی، تولید متغیرهای تصادفی از مدلهای ورودی، و تجزیه و تحلیل پاسخ به‌دست آمده را با مثالها به نمایش گذاشتند. این مطالب را با تفصیل بیشتری در بقیهٔ فصلهای کتاب بررسی کرده‌ایم. بدین ترتیب، این فصل به‌صورت توجیه برای نیاز به ۱۰ فصل دیگر به‌کار می‌آید.

مثالها اساساً از سیستمهای صف و موجودی گرفته شده‌اند زیرا بسیاری از شبیه‌سازیها به مسائلی در این زمینه‌ها مربوط می‌شوند. مثالهای بیشتری در زمینه‌های پایایی، شبیه‌سازی ایستا و تولید نمونه‌ای تصادفی از یک توزیع نامعلوم ارائه کرده‌ایم.

منابع

Graybeal, W.J., and U.W. Pooch [1980], *Simulation: Principles and Methods*, Winthrop, Cambridge, Mass.

Hadley, G., and T.M. Whitin [1963] *Analysis of Inventory Systems*, Prentice-Hall, Englewood Cliffs, N.J.

Hines, W.W., and D.C. Montgomery [1980], *Probability and Statistics in Engineering and Management Science*, 2nd. ed., Wiley, New York.

Meyer, Paul L. [1965], *Introductory Probability and Statistical Applications*, Addison-Wesley, Reading, Mass.

Ross, Sheldon [1976], *A First Course in Probability*, Macmillan, New York.

Ruie-Pala, E., C. Avila-Beloso, and W. W. Hines [1967], *Waiting Line Models, An Introduction to Their Theory and Application*, Reinhold, New York.

تمرینها

در مورد تمام مسائلی که به مثالهای کتاب اشاره دارند از ارقام تصادفی متفاوتی از جدول پ-۱ استفاده کنید.

۲-۱ در مثال ۲-۱ فرض کنید که توزیع ورود بین ۱ و ۱۰ دقیقه یکنواخت باشد. برای ۲۰ مشتری جدول شبیه‌سازی ایجاد کنید و تجزیه و تحلیل لازم را انجام دهید. تأثیر تغییر دادن توزیع مدت ورود چیست؟

۲-۲ در مثال ۲-۱ فرض کنید که توزیع خدمتدهی به‌صورت زیر تغییر کند:

مدت خدمتدهی (دقیقه)	۱	۲	۳	۴	۵	۶
احتمال	۰٫۰۵	۰٫۱۰	۰٫۲۰	۰٫۳۰	۰٫۲۵	۰٫۱۰

برای ۲۰ مشتری جدول شبیه‌سازی ایجاد کنید و تجزیه و تحلیل لازم را انجام دهید. تأثیر تغییر دادن توزیع مدت خدمتدهی چیست؟

۲-۳ شبیه‌سازی مثال ۲-۱ را در مورد ۲۰ مشتری دیگر (مشتریهای ۲۱ تا ۴۰) اجرا کنید. نتایج مثال ۲-۱ را با نتایج خود مقایسه کنید.

۲-۴ در مثال ۲-۱ میانگین وزندار زمانی تعداد مشتری در سیستم و میانگین وزندار زمانی تعداد مشتری در صف انتظار را تعیین کنید. (*راهنمایی*: شکل ۲-۶ را به‌کار ببرید.)

۲-۵ توزیع ورود خودروها در مثال ۲-۲ را به شرح زیر تغییر دهید:

مدت بین ورود (دقیقه)	۰	۱	۲	۳	۴
احتمال	۰٫۱۰	۰٫۲۰	۰٫۳۵	۰٫۲۰	۰٫۱۵

شبیه‌سازی و تحلیل پس از آن را برای یک دورهٔ یک ساعته انجام دهید. تأثیر تغییردادن توزیع مدت بین دو ورود چیست؟

۲-۶ مجدداً در مورد مثال ۲-۲، فرض کنید زانوی هابیل مجروح است و به سرعت قبل نمی‌تواند حرکت کند. در نتیجه، دو تغییر رخ می‌دهد. توزیع خدمتدهی هابیل تغییر می‌کند و اگر هر دو آورندهٔ غذا بیکار باشند، خباز در گرفتن مشتری حق تقدم دارد. توزیع جدید خدمتدهی هابیل به شرح زیر است:

مدت خدمتدهی (دقیقه)	۳	۴	۵	۶
احتمال	۰٫۳۰	۰٫۳۰	۰٫۲۵	۰٫۱۵

الف) شبیه‌سازی و تحلیل پس از آن را برای ۳۰ خدمتدهی کامل‌شده انجام دهید. تأثیر آسیب‌دیدگی هابیل و قاعدهٔ جدید چیست؟

ب) تأثیر افزودن کارمند تازه‌ای که به سرعت خباز کار می‌کند چیست؟ کارمند جدید تمام کار زیاد آمده بعد از خباز و هابیل را انجام می‌دهد.

۲-۷ مثال ۲-۲ را چنان اصلاح کنید که هرگاه هر دو خدمت‌دهنده بیکار باشند، هابیل با احتمال ۰٫۴۵ مشتری را بگیرد. تأثیر چنین تغییری چیست؟

۲-۸ تعداد بهینهٔ روزنامه‌هایی که باید در هر روز در مثال ۲-۳ خریداری شود را تعیین کنید. آیا باید از همان ارقام تصادفی برای هر میزان از روزنامه‌های خریداری شده توسط فروشنده استفاده کرد؟ چرا بله یا نه؟ استفاده از اعداد تصادفی جدید چه تأثیری دارد؟

۲-۹ نانوایی سعی دارد تعیین کند که هر روز باید چند دوجین از نان خاصی بپزد. توزیع احتمال تعداد مشتریان این نوع نان به شرح زیر است:

تعداد مشتری در روز	۸	۱۰	۱۲	۱۴
احتمال	۰/۳۵	۰/۳۰	۰/۲۵	۰/۱۰

مشتریها به موجب توزیع احتمال زیر، ۱، ۲، ۳، یا ۴ دوجین از این نان را سفارش می‌دهند.

تعداد دوجین نان سفارش داده شده توسط هر مشتری	۱	۲	۳	۴
احتمال	۰/۴	۰/۳	۰/۲	۰/۱

قیمت فروش هر دوجین از این نان ۲/۲۵ واحد پول و هزینهٔ پختن هر دوجین آن ۱/۵۸ واحد پول است. هر مقدار از این نان که در پایان روز فروش نرفته باشد به نصف قیمت به یک فروشگاه مواد غذایی فروخته می‌شود. بر اساس پنج روز شبیه‌سازی، چند دوجین (به نزدیکترین مضرب ۱۰ دوجین گرد کنید) از این نان باید هر روز پخته شود؟

۲-۱۰ تقاضا برای نوعی ابزار از توزیع احتمال نشان داده شده در زیر پیروی می‌کند:

تقاضای روزانه	۰	۱	۲	۳	۴
احتمال	۰/۳۳	۰/۲۵	۰/۲۰	۰/۱۲	۰/۱۰

موجودی انبار هر ۷ روز یک بار بررسی می‌شود (کارخانه هر روز کار می‌کند) و اگر سطح موجودی به ۶ واحد یا کمتر رسیده باشد، ۱۰ عدد از این ابزار سفارش داده می‌شود. مهلت تحویل (تعداد روزها تا تحویل) احتمالی است و طبق توزیع زیر تعریف می‌شود:

مهلت تحویل (روز)	۱	۲	۳
احتمال	۰/۳	۰/۵	۰/۲

شبیه‌سازی وقتی شروع می‌شود که آغاز هفته است، ۱۲ عدد از ابزار فوق موجود است، و هیچ سفارشی با تأخیر روبه‌رو نیست (وجود سفارش تحویل‌نشده مجاز است). ۶ هفته از عملکرد این سیستم را شبیه‌سازی و سیستم را تجزیه و تحلیل کنید. به منظور تعیین

تأثیر افزایش یا کاهش ۱) دورهٔ بررسی، ۲) مقدار سفارش مجدد و ۳) نقطهٔ زمان سفارش مجدد بر کمبودها، شبیه‌سازیهای بیشتری انجام دهید.

۲-۱۱ یک شرکت تأمین‌کنندهٔ وسایل لوله‌کشی به تعیین توزیع تقاضا در مهلت تحویل آبگیرهای صنعتی علاقه‌مند است. تابع فراوانی تقاضای روزانه معلوم و به شرح زیر است:

تقاضای روزانه	۰	۱	۲	۳	۴
احتمال	۰/۱۸	۰/۳۹	۰/۲۹	۰/۰۹	۰/۰۵

توزیع مهلت تحویل بر اساس سوابق موجود ایجاد شده و به شرح زیر است:

مهلت تحویل (روز)	۰	۱	۲	۳	۴	۵
احتمال	۰/۱۳۵	۰/۲۲۳	۰/۲۸۸	۰/۲۱۳	۰/۱۱۸	۰/۰۲۳

بر اساس ۲۰ دوره مهلت تحویل، توزیع تقاضا در مهلتهای تحویل را ایجاد کنید. با استفاده از فواصل ۰-۲، ۳-۵، ... هیستوگرام این توزیع را تهیه کنید و سپس با استفاده از فواصل ۰-۱، ۲-۳ و ۳-۴، ... ، هیستوگرام دیگری ایجاد کنید. آیا تغییر عرض فاصله تأثیر عمده‌ای بر شکل هیستوگرام توزیع تقاضا در مهلت تحویل می‌گذارد؟

۲-۱۲ دیاگرامهای جریان نظیر شکلهای ۲-۲ و ۲-۳ را برای یک سیستم صف با c مجرا ایجاد کنید.

۲-۱۳ شبیه‌سازی روش پیشنهادی مثال ۲-۵ را با استفاده از ارقام جدید تصادفی دو مرتبه انجام دهید. تأثیر تولید مجموعهٔ تازه‌ای از پیشامدها بر هزینهٔ کل چیست؟ چه موقع استفاده از همان ارقام تصادفی (و پیشامدها) برای پیشنهادهای رقابتی مناسب است؟

۲-۱۴ یک شرکت تاکسیرانی بین ساعت ۹ صبح تا ۵ بعد از ظهر با یک خودرو فعالیت می‌کند. در حال حاضر، افزودن خودرو دومی به این اتومبیل در دست بررسی است. تقاضا برای تاکسی از توزیع نشان داده شده در زیر پیروی می‌کند:

مدت بین تقاضاهای تلفنی (دقیقه)	۱۵	۲۰	۲۵	۳۰	۳۵
احتمال	۰/۱۴	۰/۲۲	۰/۴۳	۰/۱۷	۰/۰۴

توزیع مدت کامل کردن هر خدمتدهی به شرح زیر است:

مدت خدمتدهی (دقیقه)	۵	۱۵	۲۵	۳۵	۴۵
احتمال	۰/۱۲	۰/۳۵	۰/۴۳	۰/۰۶	۰/۰۴

پنج روزکار سیستم فعلی و سیستمی با یک تاکسی اضافه را شبیه‌سازی کنید. دو سیستم را برحسب مدتهای انتظار مشتریان و هر معیار روشنگر دیگری مقایسه کنید.

۲-۱۵ مثال ۲-۶ را با افزودن نُه مأموریت ادامه دهید و براوردی از کیفیت کار بمب‌افکنهای شبیه‌سازی شده در حمله به زاغهٔ مهمات ارائه کنید.

۲-۱۶ متغیرهای تصادفی X، Y، و Z به شرح زیر توزیع می‌شوند:

$$X \sim N(\mu = 100, \sigma^2 = 100)$$
$$Y \sim N(\mu = 300, \sigma^2 = 225)$$
$$Z \sim N(\mu = 40, \sigma^2 = 64)$$

۵۰ مقدار از مقادیر متغیر تصادفی $W = \frac{X+Y}{Z}$ را شبیه‌سازی و با استفاده از فواصل رده‌ها با عرض ۳، هیستوگرام تهیه کنید.

۲-۱۷ مهلت تحویل برای یکی از اقلام انبار، توزیع نرمال با میانگین ۷ روز و انحراف معیار ۲ روز دارد. تقاضای روزانه به شرح زیر توزیع می‌شود:

تقاضای روزانه	۰	۱	۲	۳	۴
احتمال	۰٫۳۶۷	۰٫۳۶۸	۰٫۱۸۴	۰٫۰۶۲	۰٫۰۱۹

با ۲۰ دور سفارش، تقاضا در مهلت تحویل را تعیین کنید.

۲-۱۸ مثال ۲-۴ را در نظر بگیرید.

الف) مثال را تا ۱۵ دور دیگر ادامه دهید و نتیجه‌گیری کنید.

ب) به ازای $M = 10$ مثال را برای ۱۰ دور در دو مرتبه انجام دهید.

ج) به ازای $N = 6$ مثال را برای ۱۰ دور در دو مرتبه انجام دهید.

۲-۱۹ در مورد یک سیستم موجودی که به شرح زیر عمل می‌کند، با شبیه‌سازی متوسط تعداد فروشهای از دست‌رفته در هفته را براورد کنید:

الف) هرگاه سطح موجودی به ۱۰ یا زیر آن برسد، سفارشی صادر شود. در هر زمان تنها یک سفارش ممکن است دریافت نشده باشد.

ب) مقدار هر سفارش مساوی با $20 - I$ باشد، که I سطح موجودی به هنگام صدور سفارش است.

ج) اگر به هنگام صفر بودن موجودی تقاضایی برسد فروش از دست برود.

د) تقاضای روزانه توزیع نرمال با میانگین ۵ واحد و انحراف معیار ۱٫۵ واحد دارد. (در جریان شبیه‌سازی، تقاضا را به نزدیکترین عدد صحیح گرد کنید.)

ه) مهلت تحویل بین صفر و ۵ روز توزیع یکنواخت دارد و تنها مقادیر صحیح می‌گیرد.

و) شبیه‌سازی با ۱۸ واحد موجودی در انبار آغاز می‌شود.

ز) برای سادگی، فرض کنید که تقاضا همواره در ۱۲ ظهر می‌رسد و سفارشها نیز در وقت واحدی صادر می‌شوند. به‌علاوه، فرض کنید که سفارشها در ساعت ۵ بعدازظهر یا پس از رسیدن تقاضای آن روز دریافت می‌شوند.

ح) شبیه‌سازی را برای ۵ هفته انجام دهید.

۲-۲۰ بالابری در یک کارخانهٔ تولیدی، موادی دقیقاً به وزن ۴۰۰ کیلوگرم را حمل می‌کند. این مواد از سه نوع‌اند و برای انتقال توسط بالابر به جعبه‌هایی وارد می‌شوند. این مواد و توزیعهای مدت بین ورودشان به شرح زیر است:

ماده	وزن (کیلوگرم)	مدت بین ورود (دقیقه)
A	۲۰۰	۵ ± ۲ (یکنواخت)
B	۱۰۰	۶ (ثابت)
C	۵۰	$P(2) = 0.33$
		$P(3) = 0.67$

بالابر در بالا رفتن به طبقهٔ دوم، یک دقیقه، در تخلیهٔ بار، ۲ دقیقه و در بازگشت به طبقهٔ اول، یک دقیقه وقت صرف می‌کند. تا کامل نشدن وزن محموله، بالابر طبقهٔ اول را ترک نمی‌کند. یک ساعت از عمل سیستم را شبیه‌سازی کنید. متوسط مدت انتقال یک جعبهٔ مواد A (از زمان ورود تا تخلیه) چقدر است؟ متوسط مدت انتظار یک جعبهٔ مواد B چقدر است؟ چند جعبهٔ مواد C طی یک ساعت منتقل شدند؟

۲-۲۱ متغیرهای تصادفی X و Y به شرح زیر توزیع می‌شوند.

$X \sim 10 \pm 10$ (یکنواخت)

$Y \sim 10 \pm 8$ (یکنواخت)

الف) ۲۰۰ مقدار متغیر تصادفی $Z = XY$ را شبیه‌سازی و از نتایج به‌دست آمده نمودار فراوانی تهیه کنید. دامنهٔ Z و مقدار متوسط آن چیست؟

ب) همانند (الف) عمل کنید با این تفاوت که $Z = \frac{X}{Y}$ باشد.

۲-۲۲ یکی از تمرینهای شبیه‌سازی در فوق را با فرترن، GPSS، SIMSCRIPT، GASP، SLAM یا هر زبان کامپیوتری دیگری انجام دهید.

۳

شبیه‌سازی گسسته پیشامد: اصول کلی و زبانهای شبیه‌سازی کامپیوتری

این فصل چارچوبی مشترک برای مدلسازی سیستمهای پیچیده ایجاد و برخی از زبانهای اصلی شبیه‌سازی پیشامدهای گسسته را معرفی می‌کند. این رهیافت مدلسازی شبیه‌سازی گسسته پیشامد نامیده می‌شود. همان‌طور که در فصل ۱ به اختصار گفتیم، هر سیستم را برحسب حالت خود در هر لحظه از زمان و پیشامدهای گوناگونی که رویدادشان تغییر حالت را به همراه دارد، می‌توان مدلسازی کرد. مدلسازی گسسته پیشامد برای سیستمهایی مناسب است که تغییرات مهم در حالت آنها در لحظه‌های گسستهٔ زمان روی می‌دهد.

هر زبان شبیه‌سازی گسسته پیشامد، دید کلی یا شیوهٔ نگرش خاص خود به سیستم در دست مدلسازی را دارد. زبانهای تشریح شده در این فصل را می‌توان به زبانهایی رده‌بندی کرد که یا رهیافت زمانبندی پیشامدها یا رهیافت پردازش-تقابل را در مدلسازی گسسته پیشامد در پیش می‌گیرند. رهیافت زمانبندی پیشامدها ایجاب می‌کند که تحلیلگر توجه خود را به پیشامدها و چگونگی تأثیر آنها بر حالت سیستم معطوف کند. رهیافت پردازش-تقابل به تحلیلگر اجازه می‌دهد تا توجه خود را به یک نهاد منفرد (مانند یک مشتری) و توالی پیشامدها و فعالیتهایی که او با «گذرکردن» از سیستم آنها را تجربه می‌کند معطوف دارد. به هنگام استفاده از یک زبان همه‌منظوره از قبیل FORTRAN، ALGOL، BASIC یا پاسکال شبیه‌ساز به احتمال فراوان رهیافت زمانبندی پیشامدها را برمی‌گزیند. زبانهایی از قبیل GASP استفاده از رهیافت زمانبندی پیشامدها را تسهیل می‌کند. اما GPSS امکان استفاده از رهیافت پردازش-تقابل را به نوعی فراهم می‌آورد. برخی از زبانهای جدیدتر مانند SIMSCRIPT و SLAM به شبیه‌ساز اجازه می‌دهند

یکی از دو طریق فوق یا آمیزه‌ای از هر دو را برحسب متناسب بودن با مسألهٔ مورد نظر به‌کار برد.

بخش ۳-۱ اصول کلی مربوط به رهیافتهای زمانبندی پیشامدها و پردازش تقابل را مورد بحث قرار می‌دهد و چند مثال شبیه‌سازی دستی را ارائه می‌کند. بخش ۳-۲ مثالهایی دربارهٔ مدلسازی هر سیستم ساده با استفاده از FORTRAN، SIMSCRIPT، GPSS و SLAM ارائه و به اختصار GASP را تشریح می‌کند.

۳-۱ مفاهیم مربوط به شبیه‌سازی گسسته پیشامد

مفهوم سیستم و مدلی از آن به اختصار در فصل ۱ مورد بحث قرار گرفت. این فصل منحصراً به سیستمهای تصادفی پویایی (یعنی دارای عناصر تصادفی و مرتبط با زمان) می‌پردازد که به طریقی گسسته تغییر می‌یابند. این بخش چنین مفاهیمی را بسط می‌دهد و چارچوبی برای ایجاد مدلی گسسته پیشامد از سیستم ارائه می‌کند. مفاهیم اصلی در آن به اختصار تعریف و سپس با مثالهایی تشریح می‌شوند.

سیستم. مجموعه‌ای از نهادها (مثل، آدمها و ماشینها) که طی زمان بر هم تأثیر متقابل می‌گذارند تا به یک یا چند هدف نائل شوند.

مدل. معرفی تجریدی سیستم که معمولاً شامل روابط منطقی و یا ریاضی است که سیستم را برحسب حالت، نهادها و ویژگیهای آنها، مجموعه‌ها، پیشامدها، فعالیتها، و تأخیرهایشان تشریح می‌کند.

حالت سیستم. جمعی از متغیرها که تمام اطلاعات لازم برای تشریح سیستم در هر لحظه را دربر داشته باشد.

نهاد. هر جزء از یک سیستم که معرفی صریح آن در مدل لازم باشد (مثلاً هر خدمت‌دهنده، هر مشتری، هر ماشین).

ویژگیها. خواص هر نهاد مفروض (مثلاً اولویت یک مشتری معین، ترتیب انجام سفارش معین در کارگاه).

مجموعه. جمعی (دائمی یا موقتی) از نهادهای مرتبط که به طریقی منطقی آراسته شده باشد (مانند تمام مشتریانی که در حال حاضر در صف انتظارند و به ترتیب ورود یا برحسب اولویت مرتب شده‌اند).

پیشامد. رویدادی لحظه‌ای که حالت سیستم را تغییر می‌دهد (مثل ورود هر مشتری جدید).

فعالیت. فاصله‌ای زمانی با طول مشخص (مانند مدت خدمتدهی یا مدت بین دو ورود) که طول آن با شروعش معلوم می‌شود (هر چند که بتوان آن را برحسب یک توزیع آماری تعریف کرد). (انتظار بدون شرط)

تأخیر. فاصله‌ای زمانی با طول نامشخص که تا پایان نیافته است طول آن معلوم نمی‌شود (مانند مدت تأخیر مشتری معین در صف انتظار با نظام عکس ترتیب ورود که زمان پایان آن به ورودهای آینده بستگی خواهد داشت). (انتظار مشروط)

مجموعه گاهی فهرست، صف، یا زنجیره نیز نامیده می‌شود. هر فعالیت ممکن است قطعی (مانند یک مدت خدمتدهی که همواره ۵ دقیقه طول می‌کشد)، یا احتمالی (مثلاً 5 ± 3 دقیقه با توزیع یکنواخت)، یا هر نوع تابع ریاضی باشد. صرفنظر از اینکه کدام یک از اینها باشد، در مدل، مدت تداوم هر فعالیت به محض شروع آن قابل محاسبه است. در مقابل، هر تأخیر معمولاً وقتی پایان می‌یابد که یک شرط منطقی تحقق یابد؛ چنین شرطی معمولاً ماحصل فعل و انفعال پیشامدهای بسیاری است. مدتی که یک مشتری در صف انتظار می‌ماند، مثالی بارز از تأخیر است. گاهی تأخیر، انتظار مشروط نامیده می‌شود. اما فعالیت را انتظار نامشروط می‌نامند. توجه داشته باشید که پایان هر فعالیت پیشامدی است که اغلب پیشامد اساسی نامیده می‌شود. آغاز و پایان هر تأخیر، پیشامدی شرطی نامیده می‌شود. (آغاز هر فعالیت ممکن است پیشامدی اساسی یا شرطی باشد.) واژهٔ «پیشامد» در این کتاب به پیشامد اساسی اشاره دارد.

سیستمهای مورد بررسی در اینجا پویاست، یعنی با گذشت زمان تغییر می‌کند. بنابراین، حالت سیستم، ویژگیها و تعداد نهادهای فعال، محتوای مجموعه‌ها و فعالیتها و تأخیر در جریان همه توابعی از زمان و همواره با گذشت زمان در حال تغییر است. خود زمان با متغیری به نام CLOCK معرفی می‌شود.

■ مثال ۳-۱ (بررسی مجدد هابیل و خباز)

سیستم غذارسانی هابیل و خباز در مثال ۲-۲ را در نظر بگیرید. مدل گسسته دارای اجزاء زیر است:

حالت سیستم. $L_Q(t)$، تعداد خودروهای در حال انتظار برای دریافت خدمت، در لحظهٔ t؛ $L_A(t)$، معرف بیکار (صفر) یا مشغول (یک) بودن هابیل در لحظهٔ t؛ $L_B(t)$، معرف بیکار (صفر) یا مشغول (یک) بودن خباز در لحظهٔ t

نهادها. نه مشتریها (یعنی، خودروها) و نه خدمت‌دهنده‌ها به‌جز در قالب متغیرهای حالت نیاز به معرفی صریح ندارد. مگر اینکه برخی متوسطهای مربوط به مشتریها مد نظر باشد (مثالهای ۳-۲ و ۳-۳ را مقایسه کنید)

پیشامدها. (event) پیشامد ورود؛ پیشامد تکمیل خدمتدهی توسط هابیل؛ پیشامد تکمیل خدمتدهی توسط خباز

فعالیتها. مدت بین دو ورود، به شرح جدول ۲-۱۱؛ مدت خدمتدهی توسط هابیل، به شرح جدول ۲-۱۲؛ مدت خدمتدهی توسط خباز، به شرح جدول ۲-۱۳

تأخیر. انتظار در صف تا هابیل یا خباز آزاد شود ■

تعریف اجزاء مدل شرحی ایستا از مدل را فراهم می‌آورد. علاوه بر این، تشریح روابط پویا و فعل و انفعالات بین اجزاء نیز مورد نیاز است. برخی از پرسشهای نیازمند پاسخ عبارت است از:

۱. چگونه هر پیشامد بر حالت سیستم، ویژگیهای نهاد و محتوای مجموعه تأثیر می‌گذارد؟

۲. فعالیتها چگونه تعریف می‌شود (یعنی، قطعی است، احتمالی است، یا نوعی دیگر از معادلات ریاضی دربارهٔ آن صدق می‌کند)؟ کدام پیشامد معرف شروع یا پایان هر فعالیت است؟ آیا فعالیت می‌تواند صرفنظر از حالت سیستم شروع شود، یا شروع شدن آن مشروط به بودن سیستم در حالت خاصی است؟ (مثلاً، «فعالیت» ماشینکاری را نمی‌توان شروع کرد مگر اینکه ماشین اولاً بیکار باشد، ثانیاً، شکسته یا در دست عملیات نگهداری و تعمیر نباشد.)

۳. کدام پیشامدها آغاز (و پایان) هر نوع تأخیر را سبب می‌شود؟ هر تأخیر در کدام شرایط شروع یا تمام می‌شود؟

۴. حالت سیستم در زمان صفر چیست؟ در زمان صفر چه پیشامدهایی برای «راه‌اندازی» شبیه‌سازی (یعنی، برای شروع شبیه‌سازی) باید تولید شود؟

هر شبیه‌سازی گسسته پیشامد، مدلسازی طی زمان از سیستمی است که تمام تغییر حالتهای آن در لحظه‌های گسستهٔ زمان، یعنی در لحظه‌های وقوع پیشامدها رخ می‌دهد. شبیه‌سازی پیشامدهای گسسته (که از این پس، شبیه‌سازی نامیده می‌شود) با ایجاد توالیی از تصاویر سیستم پیش می‌رود که معرف تکوین سیستم طی زمان است. هر تصویر در زمانی مفروض ($CLOCK = t$) نه تنها حالت سیستم در زمان t، بلکه فهرستی (به نام فهرست پیشامدهای آتی) از تمام فعالیتهای جاری در آن لحظه و زمان پایان هر فعالیت، وضعیت تمام نهادها و اعضای فعلی تمام مجموعه‌ها و مقادیر فعلی آمارهای تجمعی و شمارشگرهایی را دربر دارد که در پایان شبیه‌سازی به منظور محاسبهٔ آمار تلخیص شده به‌کار می‌رود. نمونه‌ای از تصویر سیستم در شکل ۳-۱ نشان داده شده است. (هر مدل دارای همهٔ عناصر نشان داده شده در شکل ۳-۱ نیست. تصاویر بیشتری در مثالهای موجود در این فصل ارائه خواهد شد.)

شیوهٔ جلو بردن زمان شبیه‌سازی و تضمین اینکه همهٔ پیشامدها به‌ترتیب صحیح روی دهد بر اساس فهرست پیشامدهای آتی استوار است. این فهرست، مجموعهٔ ویژه‌ای است که تمام

ساعت	حالت سیستم	نهادها و ویژگیها	مجموعهٔ ۱	مجموعهٔ ۲	...	فهرست پیشامدهای آتی FEL	آمارهای تجمعی و شمارشگرها
t	$(x, y, z, \ldots)$ ⋮					$(3, t_1)$ — قرار است پیشامد نوع ۳ در زمان t_1 رخ دهد $(1, t_2)$ — قرار است پیشامد نوع ۱ در زمان t_2 رخ دهد ⋮ ⋮	

شکل ۳-۱ نمونهٔ تصویر سیستم در زمان شبیه‌سازی t.

پیشامدهایی را که وقوع آنها در زمانی در آینده زمانبندی شده است دربر دارد. زمانبندی هر پیشامد آتی بدین معنی است که در لحظهٔ آغاز هر فعالیت، مدت تداوم آن محاسبه (شاید به طریقی «تصادفی» تولید) و پیشامد انتهایی فعالیت همراه با زمان آن در فهرست پیشامدهای آتی قرار داده شود. در واقع، اکثر پیشامدهای آتی زمانبندی نمی‌شود بلکه صرفاً روی می‌دهد ــ مانند خرابیهای تصادفی یا ورودهای تصادفی. این‌گونه پیشامدهای تصادفی، در مدل با پایان فعالیتی معین معرفی می‌شود که خود آن را با توزیعی آماری می‌توان معرفی کرد.

در زمان مفروض t، فهرست پیشامدهای آتی (FEL) دربر دارندهٔ تمام پیشامدهای از پیش برنامه‌ریزی شده برای آینده، و زمانهای مربوط به آنها (با نماد t_1، t_2، ... در شکل ۳-۱) است. FEL برحسب زمان وقوع پیشامد مرتب می‌شود. به عبارت دیگر، پیشامدها به‌ترتیب زمانی در آن‌جا می‌گیرد، یعنی زمان وقوع پیشامدها در رابطهٔ

$$t < t_1 \le t_2 \le t_3 \le \cdots \le t_n$$

صدق می‌کند. زمان t مقدار CLOCK، یعنی مقدارکنونی زمان شبیه‌سازی است. پیشامد مربوط به زمان t_1 را پیشامد قریب‌الوقوع می‌نامند؛ یعنی اولین پیشامدی است که رخ خواهد داد. پس از آنکه تصویر سیستم در زمان شبیه‌سازی $CLOCK = t$ کامل شود، CLOCK را به زمان شبیه‌سازی $CLOCK = t_1$ جلو می‌بریم و پیشامد قریب‌الوقوع را از FEL خارج و اجرا می‌کنیم. اجرای پیشامد قریب‌الوقوع بدین معنی است که بر اساس تصویر قبلی در زمان t و طبیعت پیشامد قریب‌الوقوع، تصویر تازه‌ای از سیستم برای زمان t_1 ایجاد کنیم. ممکن است در زمان t_1 پیشامدهای آتی جدیدی تولید شود. اگر چنین باشد آنها را با قرار دادن در موقعیت مناسب در FEL، زمانبندی می‌کنیم. پس از آنکه تصویر تازهٔ سیستم در زمان t_1 کامل شد، ساعت را به زمان پیشامد قریب‌الوقوع جدید جلو می‌بریم و این پیشامد را اجرا می‌کنیم. این فرایند را آن‌قدر تکرار می‌کنیم که شبیه‌سازی تمام شود. توالی اعمالی را که یک شبیه‌ساز (یا زبان شبیه‌سازی) برای جلو بردن ساعت و ایجاد تصویر تازه‌ای از سیستم انجام می‌دهد، الگوریتم زمانبندی پیشامدها و جلوبری زمان می‌نامند که گامهای آن در شکل ۳-۲ آورده شده است (و در زیر تشریح می‌شود).

با پیشرفت شبیه‌سازی، طول و محتوای FEL به‌طور مداوم در حال تغییر است و بدین‌ترتیب، ادارهٔ آن به‌طریقی کارا در یک شبیه‌سازی کامپیوتری، تأثیری عمده بر کارایی برنامهٔ کامپیوتری معرف مدل دارد. عملیات اصلی پردازش فهرست که روی FEL صورت می‌گیرد عبارت است از خارج‌کردن پیشامد قریب‌الوقوع، افزودن یک پیشامد جدید به آن و گهگاه حذف پیشامدی از آن (مشهور به منتفی شدن یک پیشامد). چون پیشامد قریب‌الوقوع معمولاً در رأس فهرست قرار دارد، خارج کردن آن با بیشترین کارایی ممکن همراه است. افزودن پیشامدی تازه (و حذف پیشامدی قدیمی به سبب منتفی شدن آن) نیاز به جستجو در فهرست دارد. کارایی این‌گونه جستجو به ترتیب منطقی فهرست و چگونگی انجام جستجو بستگی دارد. علاوه بر FEL، تمام مجموعه‌ها در یک

تصویر قبلی سیستم در زمان t

CLOCK	حالت سیستم	...	فهرست پیشامدهای آتی	...
t	(۵, ۱, ۶)		$(3, t_1)$- پیشامد نوع ۳ در زمان t_1 رخ می‌دهد $(1, t_2)$- پیشامد نوع ۱ در زمان t_2 رخ می‌دهد $(1, t_3)$- پیشامد نوع ۱ در زمان t_3 رخ می‌دهد ⋮ ⋮ ⋮ $(2, t_n)$- پیشامد نوع ۲ در زمان t_n رخ می‌دهد	

الگوریتم زمانبندی پیشامدها و جلوبری زمان

گام ۱. پیشامد قریب‌الوقوع (پیشامد ۳، زمان t_1) را از FEL خارج کنید.

گام ۲. CLOCK را به زمان پیشامد قریب‌الوقوع جلو ببرید (یعنی CLOCKرا از t به t_1 جلو ببرید).

گام ۳. پیشامد قریب‌الوقوع را اجرا کنید: حالت سیستم را تازه کنید. ویژگیهای نهاد و اعضای مجموعه‌ها را بر حسب نیاز تغییر دهید.

گام ۴. (در صورت لزوم) پیشامدهای آتی تولید و در موقعیت صحیح در FEL قرار دهید. (مثال: پیشامد ۴ قرار است در زمان t^* رخ دهد، به‌طوری که $t_2 < t^* < t_3$ باشد).

گام ۵. آمارهای تجمعی و شمارشگرها را تازه کنید.

تصویر جدید سیستم در زمان t_1

CLOCK	حالت سیستم	...	فهرست پیشامدهای آتی	...
t_1	(۵, ۱, ۵)		$(1, t_2)$- پیشامد نوع ۱ در زمان t_2 رخ می‌دهد. $(4, t^*)$-پیشامد نوع ۴ در زمان t^* رخ می‌دهد. $(1, t_3)$- پیشامد نوع ۱ در زمان t_3 رخ می‌دهد. ⋮ ⋮ ⋮ $(2, t_n)$-پیشامد نوع ۲ در زمان t_n رخ می‌دهد.	

شکل ۳-۲ جلوبری زمان شبیه‌سازی و تازه کردن تصویر سیستم.

مدل به ترتیبی منطقی نگهداری می‌شود. و عملیات افزودن و خارج کردن نهادها از مجموعه، معمولاً به شیوه‌های کارای پردازش فهرست نیازمند است. مقدمه‌ای کوتاه بر پردازش فهرست در شبیه‌سازی توسط لا و کلتون [۱۹۸۲، فصل ۲] ارائه شده است.

خارج کردن پیشامدها از FEL و افزودن پیشامدها به آن در شکل ۳-۲ نمایش داده شده است. مثلاً، پیشامد ۳ با زمان وقوع t_1 معرف پیشامد تکمیل خدمتدهی از سوی خدمت دهندهٔ ۳ است. چون این، پیشامد قریب‌الوقوع در زمان t است، در گام ۱ (شکل ۳-۲) از الگوریتم زمانبندی پیشامدها و جلوبری زمان از FEL خارج می‌شود. وقتی که پیشامد ۴ (مثلاً، یک پیشامد

ورود) با زمان وقوع t^* درگام ۴ تولید شود، یک راه ممکن برای تعیین موقعیت صحیح آن در FEL، انجام یک جستجوی از بالا به پایین است:

اگر $t^* < t_2$ است، پیشامد ۴ را در رأس FEL قرار دهید.

اگر $t_2 \le t^* < t_3$ است، پیشامد ۴ را در موقعیت دوم فهرست قرار دهید.

اگر $t_3 \le t^* < t_4$ است، پیشامد ۴ را در موقعیت سوم فهرست قرار دهید.

⋮

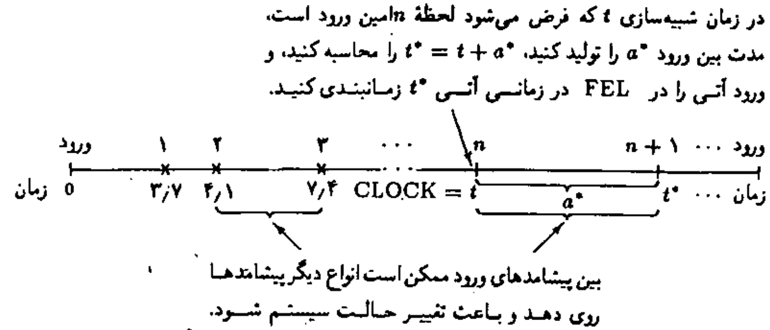
اگر $t_n \le t^*$ است، پیشامد ۴ را در موقعیت آخر فهرست قرار دهید.

(در شکل ۳-۲ فرض شده است که t^* بین t_2 و t_3 قرار دارد.) راه دیگر، انجام یک جستجوی از پایین به بالاست. راهی برای نگهداری FEL که کمترین کارایی را دارد، رها کردن آن به‌صورت یک فهرست مرتب نشده است (موارد افزوده به‌طور اختیاری در بالا یا پایین فهرست قرار داده می‌شود)، که درگام ۱ شکل ۳-۲، تعیین پیشامد قریب‌الوقوع پیش از هر مورد جلوبری ساعت، نیاز به جستجوی کامل در فهرست دارد. (پیشامد قریب‌الوقوع، پیشامدی در FEL است که کمترین زمان وقوع را داشته باشد.)

تصویر سیستم در زمان صفر با مشخص کردن شرایط اولیه و تولید پیشامدهای به اصطلاح برونزا تعریف می‌شود. شرایط اولیهٔ مشخص شده، حالت سیستم در زمان صفر را تعریف می‌کند. مثلاً اگر در شکل ۳-۲، $t = 0$ باشد، حالت (۵, ۱, ۶) را می‌توان معرف تعداد اولیهٔ مشتریها در سه نقطهٔ مختلف در سیستم قرار داد. پیشامد برونزا، رخدادی «خارج از سیستم» است که به سیستم نفوذ می‌کند. یک مثال مهم، ورود به سیستم صف است. اولین پیشامد ورود در زمان صفر تولید و در FEL برنامه‌ریزی می‌شود. مدت بین دو ورود مثالی در مورد فعالیت است. سرانجام وقتی ساعت به زمان این اولین ورود جلو برده می‌شود، یک پیشامد دوم ورود تولید می‌شود. ابتدا، یک مدت بین دو ورود مانند a^* تولید و سپس به زمان کنونی، مثلاً $\text{CLOCK} = t$، افزوده می‌شود؛ زمان به‌دست آمده، $t + a^* = t^*$، برای پیشامد (آتی)، به منظور تعیین موقعیت پیشامد جدید ورود در FEL مورد استفاده قرار می‌گیرد. این روش تولید رشته‌ای از ورودهای خارجی روش خود راه‌انداز نامیده می‌شود و مثالی از چگونگی تولید پیشامدهای آتی درگام ۴ الگوریتم زمانبندی پیشامدها و جلوبری زمان را ارائه می‌کند. روش خود راه‌انداز در شکل ۳-۳ نمایش داده شده است. اولین سه مدت بین دو ورود تولید شده، ۳٫۷، ۰٫۴، و ۳٫۳ واحد زمان است. شروع و پایان هر فاصلهٔ بین دو ورود، مثالهایی از پیشامدهای اساسی‌اند.

مثال دوم از چگونگی تولید پیشامدهای آتی (گام ۴ از شکل ۳-۲)، در قالب پیشامد تکمیل خدمتدهی در شبیه‌سازی یک سیستم صف معین ارائه می‌شود. هرگاه مشتری خدمتگیری را، مثلاً در زمان فعلی $\text{CLOCK} = t$ کامل کند در صورتی که مشتری بعدی حاضر باشد یک مدت جدید خدمتدهی مثل s^* برای او تولید خواهد شد. وقوع پیشامد بعدی تکمیل خدمتدهی در زمان $t^* = t + s^*$، با وارد کردن یک پیشامد تکمیل خدمتدهی با زمان رویداد t^* در FEL زمانبندی

مدت خدمت دهی ← فعالیت است
آغاز " " ← پیشامد شرطی است



شکل ۳-۳ تولید رشته‌ای از ورودهای خارجی با استفاده از روش خود راه‌انداز.

خواهد شد. به‌علاوه، در زمان وقوع هر پیشامد ورود، یک پیشامد تکمیل خدمتدهی تولید و زمانبندی می‌شود مشروط بر اینکه به هنگام ورود، دست‌کم یک خدمت‌دهندهٔ بیکار در گروه خدمت دهندگان وجود داشته باشد. مدت خدمتدهی نیز مثالی از فعالیت است. آغاز خدمتدهی پیشامدی شرطی است. زیرا صرفاً با این شرط شروع می‌شود که یک مشتری در سیستم حاضر و یک خدمت‌دهنده آزاد باشد. تکمیل خدمتدهی مثالی از پیشامد اساسی است. توجه کنید که هر پیشامد شرطی، از قبیل آغاز خدمتدهی، با وقوع یک پیشامد اساسی و وجود برخی شرایط حاکم در سیستم رخ می‌دهد.

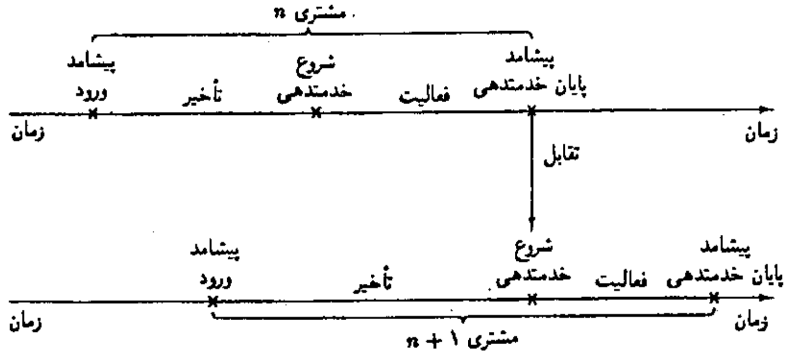
سومین مثال مهم، تولید متناوب مدتهای کار و مدتهای از کارماندگی ماشینی است که دچار خرابی می‌شود. در زمان صفر، اولین مدت کار تولید و یک پیشامد پایان مدت کار زمانبندی می‌شود. هرگاه یک پیشامد پایان مدت کار رخ دهد، یک مدت از کارماندگی تولید و یک پیشامد پایان مدت از کارماندگی در FEL زمانبندی می‌شود. هرگاه سرانجام CLOCK به زمان این پیشامد پایان مدت از کارماندگی جلو برده شود، یک مدت کار تولید و یک پیشامد پایان مدت کار در FEL زمانبندی می‌شود. بدین ترتیب، مدتهای کار و مدتهای از کارماندگی در سراسر مدت شبیه‌سازی به‌صورت متناوب قرار می‌گیرد. مدت کار و مدت از کارماندگی مثالهایی از فعالیت و پایان مدت کار و پایان مدت از کارماندگی پیشامدهای اساسی است.

(هر شبیه‌سازی باید دارای یک پیشامد پایان اجرا باشد که در اینجا با نماد E معرفی شده است. این پیشامد تعیین می‌کند شبیه‌سازی به چه مدت اجرا می‌شود. به‌طور کلی برای به پایان بردن هر شبیه‌سازی دو راه وجود دارد:)

۱. در زمان صفر، یک پیشامد پایان شبیه‌سازی را برای زمانی در آینده مانند T_E زمانبندی‌کنید. بنابراین، پیش از شبیه‌سازی معلوم است که شبیه‌سازی در فاصلهٔ زمانی $[۰, T_E]$ اجرا خواهد شد. مثال: شبیه‌سازی یک کارگاه به مدت $T_E =$ ۴۰ ساعت.

۲. مدت اجرا، T_E، توسط خود شبیه‌سازی تعیین می‌شود. به‌طور کلی T_E زمان وقوع پیشامد معینی مانند E است. چند مثال: T_E زمان تکمیل ۱۰۰امین خدمتدهی در یک مرکز خدمتدهی

مدتهای کار و مدتهای از کارماندگی ← فعالیت
پایان مدت کار و پایان مدت از کارماندگی ← پیشامدهای اساسی



شکل ۳-۴ در پردازش متقابل مشتری در صفی با یک خدمت‌دهنده.

است. T_E زمان خراب شدن یک سیستم پیچیده است. T_E زمان پایان درگیری یا نابودی کامل (برحسب اینکه کدام زودتر روی دهد) در یک شبیه‌سازی جنگی است.

در مورد۲، T_E از قبل معلوم نیست و در واقع، ممکن است از آمار بسیار مورد علاقه‌ای باشد که باید با شبیه‌سازی به‌دست آید.

رهیافتی نظام‌مند در شبیه‌سازی که بر پیشامدها و تأثیرشان بر حالت سیستم تأکید دارد، رهیافت *زمانبندی پیشامدها* در شبیه‌سازی گسسته پیشامد نام دارد. در شبیه‌سازیهای دستی زیربخش ۳-۱-۱ و شبیه‌سازیهای FORTRAN و SIMSCRIPT در بخش ۳-۲، این رهیافت را تشریح کرده‌ایم. رهیافت پردازش-تقابل، نگرشی متفاوت ارائه می‌دهد. هر فرایند، مجموعه‌ای مرتب و زمانی از پیشامدها، فعالیتها و تأخیرها است که به گونه‌ای، به یک نهاد مربوط است. مثالی از یک «پردازش مشتری» در شکل ۳-۴ نشان داده شده است. پردازشهای بسیاری معمولاً به‌طور همزمان فعال است، و ارتباط میان پردازشها ممکن است کاملاً پیچیده باشد. شکل ۳-۴ تقابل بین پردازش پی‌درپی دو مشتری را نیز در یک صف تک مجرایی با نظام خدمتگیری به‌ترتیب ورود نمایش می‌دهد. از جمله زبانهای مبتنی بر رهیافت پردازش-تقابل، GPSS و SLAM است؛ SIMSCRIPT II.5 (نشر ۸) نیز امکان اختیاری استفاده از رهیافت پردازش-تقابل را فراهم می‌آورد. در بخش ۳-۲ نمایشهایی از این زبانهای مبتنی بر پردازش ارائه شده است. ■

۳-۱-۱ شبیه‌سازی دستی با استفاده از زمانبندی پیشامدها

در اجرای هر شبیه‌سازی با به‌کارگیری رهیافت زمانبندی پیشامدها از یک جدول شبیه‌سازی برای ثبت تصاویر پیاپی سیستم با گذشت زمان استفاده می‌کنند.

www.matlabdl.com

■ **مثال ۳-۲ (صف تک مجرایی)**

بار دیگر فروشگاه مواد غذایی با یک باجهٔ صندوق را در نظر بگیرید که در مثال ۲-۱ با روشی موردی و مخصوص شبیه‌سازی شد. سیستم مشتریانی که در صف انتظارند به‌اضافهٔ کسی که در محل صندوق مشغول پرداخت وجه است را دربر می‌گیرد. مدل دارای اجزاء زیر است:

حالت سیستم. $(\mathrm{LQ}(t), \mathrm{LS}(t))$، که $\mathrm{LQ}(t)$ تعداد مشتریان در صف انتظار، و $\mathrm{LS}(t)$ تعداد مشتری در حال خدمتگیری (صفر یا یک) در زمان t است.

نهادها. خدمت‌دهنده و مشتریان صریحاً مدلسازی نمی‌شوند مگر در قالب متغیرهای حالت به شرح بالا

پیشامدها. ورود (A)؛ ترک (D)؛ پیشامد پایان اجرا (E)، که وقوع آن برای زمان ۶۰ زمانبندی شده است

فعالیتها. مدت بین دو ورود، به شرح جدول ۲-۶؛ مدت خدمتدهی، به شرح جدول ۲-۷

تأخیر. مدت ماندن مشتری در صف انتظار

پیشامدها در FEL به‌صورت (زمان پیشامد، نوع پیشامد) نوشته می‌شوند. FEL در این مدل همواره دو یا سه پیشامد خواهد داشت. تأثیر پیشامدهای ورود و ترک، نخست در شکلهای ۲-۲ و ۲-۳ نشان داده شد و با تفصیل بیشتری در شکلهای ۳-۵ و ۳-۶ نشان داده می‌شود.

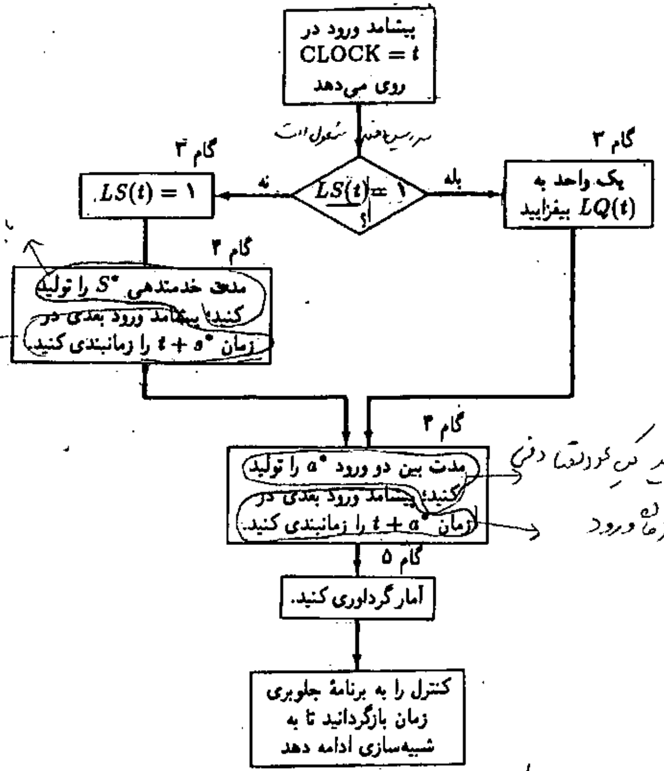
جدول شبیه‌سازی باجهٔ صندوق در جدول ۳-۱ ارائه شده است. خواننده باید با شروع از تصویر اول، تمام تصاویر سیستم به‌جز یکی را بررسی و سعی کند تصویر بعدی را از تصویر قبلی و منطق پیشامدهای شکلهای ۳-۵ و ۳-۶ به‌وجود آورد. مدتهای بین دو ورود و مدتهای خدمتدهی شبیه مدتهای مورد استفاده در جدول ۲-۱۰ است، یعنی:

مدتهای بین دو ورود	۸	۶	۱	۸	۳	۸	...
مدتهای خدمتدهی	۴	۱	۴	۳	۲	۴	...

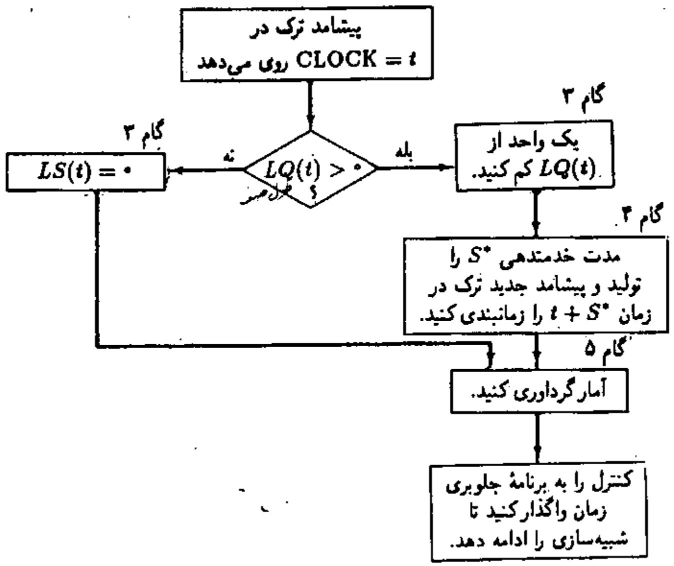
شرایط اولیه چنین است که اولین مشتری در زمان صفر وارد و خدمتدهی به او شروع می‌شود. این امر در جدول ۳-۱ و در قالب تصویر سیستم در زمان صفر $(\mathrm{CLOCK} = ۰)$ به گونه‌ای منعکس است که $\mathrm{LQ}(۰) = ۰$، $\mathrm{LS}(۰) = ۱$، و یک پیشامد ترک و یک پیشامد ورود با هم در FEL است. شبیه‌سازی نیز به گونه‌ای زمانبندی شده است که در زمان ۶۰ متوقف شود. تنها دو نوع آمار، یعنی بهره‌برداری از خدمت‌دهنده و بزرگترین طول صف گردآوری خواهد شد. بهره‌برداری از خدمت‌دهنده به‌صورت جمع مدت اشتغال خدمت‌دهنده (B) تقسیم بر مدت کل (T_E) تعریف می‌شود. هم‌چنانکه شبیه‌سازی پیشرفت می‌کند، جمع مدت اشتغال، B، و بزرگترین طول صف، MQ، تازه خواهد شد. به منظور کمک به خواننده، ستونی با نام «توضیحات» در جدول ۳-۱ گنجانیده شده است. (a^* و s^*، به‌ترتیب، مدتهای بین دو ورود و خدمتدهی است.)

به محض کامل شدن تصویر سیستم در زمان $\mathrm{CLOCK} = ۰$ شبیه‌سازی آغاز می‌شود. در

جمع مدت اشتغال خدمت‌دهنده B / T = بهره‌برداری از خدمت‌دهنده



شکل ۳-۵ اجرای پیشامد ورود.



شکل ۳-۶ اجرای پیشامد ترک.

www.matlabdl.com

مجموع مدت اشتغال خدمت دهنده

جدول ۳-۱ جدول شبیه‌سازی برای باجهٔ صندوق (مثال ۳-۲).

ساعت	حالت سیستم LQ(t)	حالت سیستم LS(t)	فهرست پیشامدهای آتی	توضیحات	آمار تجمعی B	آمار تجمعی MQ
۰	۰	۱	(D, ۴), (A, ۸), (E, ۶۰)	Aی اول رخ می‌دهد	۰	۰
				(۸ = a^*)Aی بعدی را زمانبندی کنید		
				(۴ = s^*)Dی اول را زمانبندی کنید		
۴	۰	۰	(A, ۸), (E, ۶۰)	Dی اول رخ می‌دهد: (D, ۴)	۴	۰
۸	۰	۱	(D, ۹), (A, ۱۴), (E, ۶۰)	Aی دوم رخ می‌دهد: (A, ۸)	۴	۰
				(۶ = a^*)Aی بعدی را زمانبندی کنید		
				(۱ = s^*)Dی بعدی را زمانبندی کنید		
۹	۰	۰	(A, ۱۴), (E, ۶۰)	Dی دوم رخ می‌دهد: (D, ۹)	۵	۰
۱۴	۰	۱	(A, ۱۵), (D, ۱۸), (E, ۶۰)	Aی سوم رخ می‌دهد: (A, ۱۴)	۵	۰
				(۴ = s^*)Dی بعدی را زمانبندی کنید		
۱۵	۱	۱	(D, ۱۸), (A, ۲۳), (E, ۶۰)	Aی چهارم رخ می‌دهد: (A, ۱۵)	۶	۱
				(مشتری به انتظار می‌ماند)		
۱۸	۰	۱	(D, ۲۱), (A, ۲۳), (E, ۶۰)	Dی سوم رخ می‌دهد: (D, ۱۸)	۹	۱
				(۳ = s^*)Dی بعدی را زمانبندی کنید		
۲۱	۰	۰	(A, ۲۳), (E, ۶۰)	Dی چهارم رخ می‌دهد: (D, ۲۱)	۱۲	۱

زمان صفر پیشامد قریب‌الوقوع، (D, ۴) است. CLOCK به زمان ۴ جلو برده و (D, ۴) از FEL بیرون آورده می‌شود. چون به ازای $۴ \le t \le ۱، ۰$ ،$\text{LS}(t) = ۱$ است (یعنی، خدمت‌دهنده به مدت ۴ دقیقه مشغول بوده است)، مدت اشتغال تجمعی از ۰ = B به ۴ = B افزایش می‌یابد. طبق منطق پیشامد در شکل ۳-۶، LS(۴) را مساوی صفر قرار دهید (خدمت‌دهنده بیکار می‌شود). FEL تنها با دو پیشامد آتی، (A, ۸) و (E, ۶۰) باقی می‌ماند. سپس، CLOCK شبیه‌سازی به زمان ۸ جلو برده و یک پیشامد ورود اجرا می‌شود. تعبیر بقیهٔ جدول ۳-۱ را به خواننده واگذار می‌کنیم.

شبیه‌سازی در جدول ۳-۱ فاصلهٔ زمانی [۰ ,۲۱] را می‌پوشاند. در زمان شبیه‌سازی‌شدهٔ ۲۱، سیستم خالی است ولی ورود بعدی در زمان آیندهٔ ۲۳، روی خواهد داد. از ۲۱ واحد زمان شبیه‌سازی شده، خدمت‌دهنده ۱۲ واحد زمان را مشغول بود که برای کسب نتایج اعتمادپذیر بسیار کوتاه است. تمرین ۱ از خواننده می‌خواهد که شبیه‌سازی را ادامه دهد و نتایج را با نتایج مثال ۳-۱ مقایسه کند. توجه کنید که جدول شبیه‌سازی، حالت سیستم را در تمام زمانها و نه تنها در زمانهای فهرست شده ارائه می‌دهد. مثلاً، از زمان ۱۵ تا زمان ۱۸، یک مشتری در حال خدمتگیری و یکی نیز در صف انتظار بوده است.

وقتی یک الگوریتم زمانبندی پیشامدها کامپیوتری می‌شود، در حافظهٔ کامپیوتر تنها یک تصویر

(تصویر کنونی یا تصویری که بخشهایی از آن تازه شده است) نگهداری می‌شود. با قصد به‌کارگیری زمانبندی پیشامدها در FORTRAN یا هر زبان هم‌منظوره دیگر، باید از قاعدهٔ زیر پیروی کرد. یک تصویر تازه تنها ممکن است با استفاده از تصویر قبلی، متغیرهای تصادفی تازه تولیدشده و منطق پیشامدها (شکلهای ۳-۵ و ۳-۶) ایجاد شود. به‌هنگام جلو بردن ساعت تصاویر گذشته را باید نادیده گرفت. تصویر کنونی نیز باید حاوی تمام اطلاعات لازم برای ادامه دادن شبیه‌سازی باشد. ■

■ مثال ۳-۳ (ادامهٔ شبیه‌سازی باجهٔ صندوق)

فرض کنید که در شبیه‌سازی باجهٔ صندوق در مثال ۳-۲، شبیه‌ساز مایل است میانگین مدت پاسخ و میانگین نسبت مشتریانی که ۴ دقیقه یا بیشتر در سیستم می‌مانند را براورد کند. یک مدت پاسخ مدتی است که یک مشتری در سیستم می‌ماند. به منظور براورد کردن تعداد این مشتریان لازم است مدل مثال ۳-۲ را چنان گسترش داد که هر مشتری را صریحاً معرفی کند. به‌علاوه، برای اینکه بتوان به‌هنگام ترک سیستم از سوی یک مشتری مدت پاسخ او را محاسبه کرد دانستن زمان ورود مشتری لازم خواهد بود. بنابراین، یک نهاد مشتری با ویژگی زمان ورود به فهرست اجزاء مدل در مثال ۳-۲ افزوده می‌شود. این‌گونه نهادهای مشتری را که در مجموعه‌ای به نام «زمان ورود» ذخیره می‌شود C۱، C۲، C۳، ... می‌نامیم. سرانجام، نمادهای مربوط به پیشامدها در FEL را برای نشان دادن اینکه کدام مشتری تحت تأثیر قرار می‌گیرد گسترش می‌دهیم. مثلاً، (D, ۴, C۱) بدین معنی است که مشتری C۱ در زمان ۴ سیستم را ترک خواهد کرد. فهرست اجزاء اضافی مدل به شرح زیر است:

نهادها. $(\text{C}i, t)$، معرف مشتری Ci است که در زمان t وارد شد

پیشامدها. $(\text{A}, t, \text{C}i)$، ورود مشتری Ci در زمان t ؛ $(\text{D}, t, \text{C}j)$، ترک مشتری Cj در زمان t

مجموعه. «زمان ورود»، مجموعهٔ تمام مشتریانی که در حال حاضر در سیستم‌اند (در حال خدمتگیری یا به انتظار دریافت خدمت)، به‌ترتیب زمان ورود

سه قلم آمار تازه گردآوری می‌کنیم: S، مجموع مدتهای پاسخ برای تمام مشتریانی که تا زمان کنونی سیستم را ترک کرده‌اند؛ F، جمع تعداد مشتریانی که ۴ دقیقه یا بیشتر در سیستم می‌مانند؛ و N_D، جمع موارد ترک سیستم تا زمان کنونی شبیه‌سازی. هرگاه پیشامد ترک روی دهد، این سه قلم آمار تجمعی را تازه می‌کنیم، منطق مربوط به گردآوری این اقلام آماری را، در گام ۵ پیشامد ترک در شکل ۳-۶ شرکت می‌دهیم.

جدول شبیه‌سازی برای مثال ۳-۳ در جدول ۳-۲ نشان داده شده است. برای مدتهای بین دو ورود و خدمتدهی از همان داده‌ها مجدداً استفاده می‌کنیم، به‌طوری که جدول ۳-۲، اساساً به‌صورت تکرار جدول ۳-۱ درآمده است. با این استثنا که اجزاء جدید نیز در آن گنجانیده (و ستون

جدول ۳-۲ جدول شبیه‌سازی برای مثال ۳-۲.

ساعت	حالت سیستم LQ(t)	LS(t)	مجموعهٔ «زمان ورود»	فهرست پیشامدهای آتی	آمار تجمعی S	N_D	F
۰	۰	۱	(C۱, ۰)	(D, ۴, C۱), (A, ۸, C۲), (E, ۶۰)	۰	۰	۰
۴	۰	۰		(A, ۸, C۲), (E, ۶۰)	۴	۱	۱
۸	۰	۱	(C۲, ۸)	(D, ۹, C۲), (A, ۱۴, C۳), (E, ۶۰)	۴	۱	۱
۹	۰	۰		(A, ۱۴, C۳), (E, ۶۰)	۵	۲	۱
۱۴	۰	۱	(C۳, ۱۴)	(A, ۱۵, C۴), (D, ۱۸, C۳), (E, ۶۰)	۵	۲	۱
۱۵	۱	۱	(C۳, ۱۴), (C۴, ۱۵)	(D, ۱۸, C۳), (A, ۲۳, C۵), (E, ۶۰)	۵	۲	۱
۱۸	۰	۱	(C۴, ۱۵)	(D, ۲۱, C۴), (A, ۲۳, C۵), (E, ۶۰)	۹	۳	۲
۲۱	۰	۰		(A, ۲۳, C۵), (E, ۶۰)	۱۵	۴	۳

توضیحات از آن حذف) شده است. این اجزاء جدید برای محاسبهٔ S، F، و N_D مورد نیاز است. مثلاً، در زمان ۴ یک پیشامد ترک در مورد مشتری C۱ رخ می‌دهد. نهاد مشتری C۱ از مجموعهٔ «زمان ورود» بیرون آورده می‌شود و همان‌طور که توجه دارید ویژگی «زمان ورود» صفر است. یعنی مدت پاسخ برای این مشتری ۴ دقیقه بوده است. بنابراین، S به مقدار ۴ دقیقه و F و N_D هر یک به اندازهٔ یک مشتری زیاد می‌شوند. همچنین به هنگام اجرای پیشامد ترک (D, ۲۱, C۴) در زمان ۲۱ مدت پاسخ مشتری C۴ به‌صورت

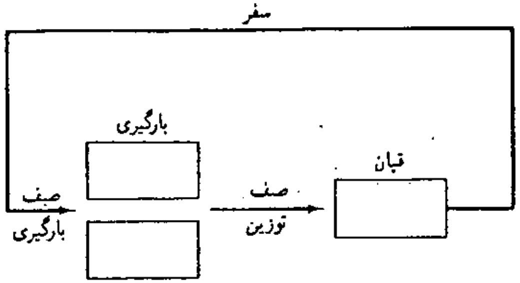
$$\text{مدت پاسخ} = \text{CLOCK} - \text{ویژگی «زمان ورود»} = ۲۱ - ۱۵ = ۶ \text{ دقیقه}$$

محاسبه می‌شود. سپس، S به مقدار ۶ دقیقه و F و N_D به تعداد یک مشتری زیاد می‌شوند.

برای شبیه‌سازی‌ای با طول اجرای ۲۱ دقیقه، متوسط مدت پاسخ، $\frac{S}{N_D} = \frac{۱۵}{۴} = ۳/۷۵$ دقیقه و نسبت مشاهده شدهٔ مشتریانی که ۴ دقیقه یا بیشتر در سیستم ماندند، $\frac{F}{N_D} = ۰/۷۵$ بود. در این مورد نیز، شبیه‌سازی، برای دانستن دقیق برآوردهای اخیر بسیار کوتاه بوده است. به‌هر صورت، مقصود مثال ۳-۳ نمایاندن این مطلب بود که در بسیاری از مدلهای شبیه‌سازی، اطلاعات مطلوب از شبیه‌سازی (از قبیل $\frac{S}{N_D}$ و $\frac{F}{N_D}$) تا حدی ساختار مدل را تعیین می‌کند. ■

■ مثال ۳-۴ (مسألهٔ کامیونها)

شش کامیون برای حمل زغال از مدخل یک معدن کوچک به راه‌آهن مورد استفاده قرار دارند. شکل ۳-۷ شمایی از عملیات کامیون را ارائه می‌دهد. هر کامیون به‌وسیلهٔ یکی از دو دستگاه بارگیری بار می‌گیرد و بلافاصله پس از بارگیری به سمت قپانی می‌رود تا در اسرع وقت توزین آن انجام



شکل ۳-۷ مسألهٔ کامیونها.

جدول ۳-۳ توزیع مدت بارگیری برای کامیونها.

مدت بارگیری	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۵	۰/۳۰	۰/۳۰	۱-۳
۱۰	۰/۵۰	۰/۸۰	۴-۸
۱۵	۰/۲۰	۱/۰۰	۹-۰

شود. دستگاههای بارگیری و قپان هر دو دارای صفهای انتظار به‌ترتیب ورود برای کامیونهاست. مدت سفر از دستگاه بارگیری به قپان[1] قابل‌اغماض محسوب می‌شود. پس از تعیین وزن، هر کامیون سفری را شروع می‌کند (که در این مدت، کامیون بار خود را تخلیه می‌کند) و سپس به صف بارگیری باز می‌گردد. توزیعهای مدت بارگیری، مدت توزین و مدت سفر، به‌ترتیب، در جدولهای ۳-۳، ۳-۴ و ۳-۵ همراه با تخصیص ارقام تصادفی برای تولید این متغیرها با استفاده از ارقام تصادفی جدول پ-۱ ارائه شده است. مقصود از شبیه‌سازی، برآورد کردن درصد مدت اشتغال هر دستگاه بارگیری و قپان است.

مدل دارای اجزاء زیر است:

حالت سیستم. $[\mathrm{LQ}(t), \mathrm{L}(t), \mathrm{WQ}(t), \mathrm{W}(t)]$ به‌طوری که در زمان شبیه‌سازی t، $\mathrm{LQ}(t)$ = تعداد کامیونها در صف بارگیری، $\mathrm{L}(t)$ = تعداد کامیونهای در حال بارگیری (صفر، ۱، یا ۲)، $\mathrm{WQ}(t)$ = تعداد کامیونها در صف توزین، $\mathrm{W}(t)$ = تعداد کامیونهای در حال توزین (صفر یا یک).

پیشامدها. (ALQ ،t ،DTi)، کامیون i در زمان t به صف بارگیری وارد می‌شود، (EL ،t ،DTi)، بارگیری کامیون i در زمان t به اتمام می‌رسد، (EW ،t ،DTi)، توزین کامیون i در زمان t به اتمام می‌رسد

نهادها. شش کامیون (DT۱، DT۲، ...، DT۶)

جدول ۳-۴ توزیع مدت توزین برای کامیونها.

مدت توزین	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۱۲	۰٫۷۰	۰٫۷۰	۱-۷
۱۶	۰٫۳۰	۱٫۰۰	۸-۰

جدول ۳-۵ توزیع مدت سفر برای کامیونها.

مدت سفر	احتمال	احتمال تجمعی	تخصیص ارقام تصادفی
۴۰	۰٫۴۰	۰٫۴۰	۱-۴
۶۰	۰٫۳۰	۰٫۷۰	۵-۷
۸۰	۰٫۲۰	۰٫۹۰	۸-۹
۱۰۰	۰٫۱۰	۱٫۰۰	۰

مجموعه‌ها. صف بارگیری، تمام کامیونهای منتظر برای شروع بارگیری که به‌ترتیب ورود مرتب شده است، صف توزین، تمام کامیونهای منتظر برای توزین که به‌ترتیب ورود مرتب شده است

فعالیتها. مدت بارگیری، مدت توزین، مدت سفر

تأخیرها. تأخیر در صف بارگیری و تأخیر در محل قپان

جدول شبیه‌سازی در جدول ۳-۶ ارائه شده است. فرض بر این است که در زمان صفر، پنج کامیون در قسمت بارگیری و یک کامیون در قسمت توزین است. مدتهای فعالیت بسته به نیاز از فهرست زیر استخراج می‌شود:

مدت بارگیری	۱۰	۵	۵	۱۰	۱۵	۱۰	۱۰
مدت توزین	۱۲	۱۲	۱۲	۱۶	۱۲	۱۶	
مدت سفر	۶۰	۱۰۰	۴۰	۴۰	۸۰		

هرگاه پیشامد اتمام بارگیری (EL)، مثلاً برای کامیون j در زمان t رخ دهد، ممکن است این رویداد پیشامدهای دیگری را سبب شود. اگر قپان بیکار باشد [$W(t) = ۰$] توزین کامیون j را آغاز و یک پیشامد اتمام توزین (EW) در FEL زمانبندی می‌کنیم؛ در غیر این صورت، کامیون j به صف توزین می‌پیوندد. اگر در این زمان کامیون دیگری به انتظار یک دستگاه بارگیری باشد، آن را از صف بارگیری خارج و با زمانبندی یک پیشامد اتمام بارگیری (EL) در FEL بارگیری آن را شروع می‌کنیم. چنین منطقی برای وقوع پیشامد اتمام بارگیری به اضافهٔ منطق مناسب برای دو پیشامد دیگر باید در نمودارهای جریان مربوط به پیشامدها، همانند شکلهای ۳-۵ و ۳-۶ شرکت داده شود. ایجاد این نمودارهای جریان به‌عنوان تمرین به خواننده واگذار می‌شود (تمرین ۲).

جدول ۳-۶ جدول شبیه‌سازی برای عملیات کامیونها (مثال ۳-۴).

ساعت t	حالت سیستم LQ(t)	L(t)	WQ(t)	W(t)	مجموعه‌ها: صف بارگیری	صف توزین	فهرست پیشامدهای آتی	آمار تجمعی B_L	B_S
۰	۳	۲	۰	۱	DT۴		(EL, ۵, DT۳)	۰	۰
					DT۵		(EL, ۱۰, DT۲)		
					DT۶		(EW, ۱۲, DT۱)		
۵	۲	۲	۱	۱	DT۵	DT۳	(EL, ۱۰, DT۲)	۱۰	۵
					DT۶		(EL, ۵ + ۵, DT۴)		
							(EW, ۱۲, DT۱)		
۱۰	۱	۲	۲	۱	DT۶	DT۳	(EL, ۱۰, DT۴)	۲۰	۱۰
						DT۲	(EW, ۱۲, DT۱)		
							(EL, ۱۰ + ۱۰, DT۵)		
۱۰	۰	۲	۳	۱		DT۳	(EW, ۱۲, DT۱)	۲۰	۱۰
						DT۲	(EL, ۲۰, DT۵)		
						DT۴	(EL, ۱۰ + ۱۵, DT۶)		
۱۲	۰	۲	۲	۱		DT۲	(EL, ۲۰, DT۵)	۲۴	۱۲
						DT۴	(EW, ۱۲ + ۱۲, DT۳)		
							(EL, ۲۵, DT۶)		
							(ALQ, ۱۲ + ۶۰, DT۱)		
۲۰	۰	۱	۳	۱		DT۲	(EW, ۲۴, DT۳)	۴۰	۲۰
						DT۴	(EL, ۲۵, DT۶)		
						DT۵	(ALQ, ۷۲, DT۱)		
۲۴	۰	۱	۲	۱		DT۴	(EL, ۲۵, DT۶)	۴۴	۲۴
						DT۵	(EW, ۲۴ + ۱۲, DT۲)		
							(ALQ, ۷۲, DT۱)		
							(ALQ, ۲۴ + ۱۰۰, DT۳)		
۲۵	۰	۰	۳	۱		DT۴	(EW, ۳۶, DT۲)	۴۵	۲۵
						DT۵	(ALQ, ۷۲, DT۱)		
						DT۶	(ALQ, ۱۲۴, DT۳)		
۳۶	۰	۰	۲	۱		DT۵	(EW, ۳۶ + ۱۶, DT۴)	۴۵	۳۶
						DT۶	(ALQ, ۷۲, DT۱)		
							(ALQ, ۳۶ + ۴۰, DT۲)		
							(ALQ, ۱۲۴, DT۳)		
۵۲	۰	۰	۱	۱		DT۶	(EW, ۵۲ + ۱۲, DT۵)	۴۵	۵۲
							(ALQ, ۷۲, DT۱)		
							(ALQ, ۷۶, DT۲)		
							(ALQ, ۵۲ + ۴۰, DT۴)		
							(ALQ, ۱۲۴, DT۳)		

جدول ۳-۶ (ادامه) جدول شبیه‌سازی برای عملیات کامیونها (مثال ۳-۴).

آمار تجمعی B_S	آمار تجمعی B_L	فهرست پیشامدهای آتی	مجموعه‌ها: صف توزین	مجموعه‌ها: صف بارگیری	حالت سیستم: $W(t)$	حالت سیستم: $WQ(t)$	حالت سیستم: $L(t)$	حالت سیستم: $LQ(t)$	ساعت t
۶۴	۴۵	(ALQ, ۷۲, DT۱)			۱	۰	۰	۰	۶۴
		(ALQ, ۷۶, DT۲)							
		(EW, ۶۴ + ۱۶, DT۶)							
		(ALQ, ۹۲, DT۴)							
		(ALQ, ۱۲۴, DT۳)							
		(ALQ, ۶۴ + ۸۰, DT۵)							
۷۲	۴۵	(ALQ, ۷۶, DT۲)			۱	۰	۱	۰	۷۲
		(EW, ۸۰, DT۶)							
		(EL, ۷۲ + ۱۰, DT۱)							
		(ALQ, ۹۲, DT۴)							
		(ALQ, ۱۲۴, DT۳)							
		(ALQ, ۱۴۴, DT۵)							
۷۶	۴۹	(EW, ۸۰, DT۶)			۱	۰	۲	۰	۷۶
		(EL, ۸۲, DT۱)							
		(EL, ۷۶ + ۱۰, DT۲)							
		(ALQ, ۹۲, DT۴)							
		(ALQ, ۱۲۴, DT۳)							
		(ALQ, ۱۴۴, DT۵)							

برای کمک به خواننده، هرگاه پیشامد جدیدی در جدول ۳-۶ زمانبندی شده، زمان پیشامد آن به‌صورت «مدت فعالیت + t» نوشته شده است. مثلاً پیشامد قریب‌الوقوع در زمان صفر، یک پیشامد EL با زمان پیشامد ۵ است. ساعت به زمان $t = ۵$ جلو برده می‌شود. کامیون ۳ به صف توزین می‌پیوندد (زیرا قپان اشغال است)، و کامیون ۴ شروع به بارگیری می‌کند. بنابراین، برای کامیون ۴ یک پیشامد EL در زمان آتی ۱۰ زمانبندی می‌شود که محاسبهٔ این زمان به‌صورت $۱۰ = ۵ + ۵ =$ (مدت بارگیری) + (زمان کنونی) است.

به منظور برآورد کردن بهره‌برداری از دستگاههای بارگیری و قپان، دو آمار تجمعی را نگهداری می‌کنیم:

$B_L =$ مجموع مدت اشتغال هر دو دستگاه بارگیری از زمان صفر تا زمان t

$B_S =$ مجموع مدت اشتغال قپان از زمان صفر تا زمان t

چون از زمان صفر تا زمان ۲۰ هر دو دستگاه بارگیری مشغول است، در زمان ۲۰ $= t$، ۴۰ $= B_L$ است. اما از زمان ۲۰ تا زمان ۲۴ تنها یک دستگاه بارگیری مشغول است. بنابراین، در فاصلهٔ زمانی [۲۰، ۲۴]، B_L تنها به مقدار ۴ دقیقه افزایش می‌یابد. همچنین، از زمان ۲۵ تا زمان ۳۶ هر دو دستگاه بارگیری بیکار است ($L(۲۵) = ۰$)، و به این ترتیب B_L تغییر نمی‌کند. برای شبیه‌سازی نسبتاً کوتاه در جدول ۳-۶، معیارهای بهره‌برداری به شرح زیر برآورد می‌شود

$$\text{متوسط بهره‌برداری از هر دستگاه بارگیری} = \frac{۴۹/۲}{۷۶} = ۰/۳۲$$

$$\text{متوسط بهره‌برداری از قپان} = \frac{۷۶}{۷۶} = ۱/۰۰$$

این برآوردها را نمی‌توان به‌عنوان برآوردهای دقیق بهره برداری در «حالت پایا» از دستگاههای بارگیری و قپان در بلندمدت محسوب داشت؛ به منظور کاستن از تأثیر شرایط مفروض در زمان صفر (حضور ۵ کامیون از مجموع ۶ کامیون در محل بارگیری) و به‌دست آوردن برآوردهای دقیق، یک شبیه‌سازی بسیاری طولانی‌تر مورد نیاز است. از سوی دیگر، اگر شبیه‌ساز به رفتار گذرای سیستم در مدتی کوتاه (مثل یک یا دو ساعت)، و به ازای شرایط مفروض اولیه علاقه‌مند باشد، می‌تواند نتایج موجود در جدول ۳-۶ را معرف (یا نمونه‌ای از) این رفتار گذرا بداند. با انجام شبیه‌سازیهای بیشتر، می‌توان نمونه‌های دیگری به‌دست آورد. به‌طوری که هر شبیه‌سازی دارای همان شرایط اولیه باشد ولی از رشتهٔ متفاوتی از ارقام تصادفی برای تولید مدتهای مربوط به فعالیتها استفاده کند.

جدول ۳-۶، یعنی جدول شبیه‌سازی مربوط به عملیات کامیونها را می‌شد با صریحاً مدلسازی نکردن کامیونها به‌عنوان نهاد، قدری ساده‌تر کرد. به عبارت دیگر، پیشامدها ممکن بود به‌صورت (EL, t)، و ... نوشته شوند، و متغیرهای حالت را می‌شد فقط برای نگهداشتن حساب تعداد کامیونها در هر بخش از سیستم به‌کار برد نه برای تعیین اینکه کدام کامیونها در این امور درگیرند. با این‌گونه معرفی، همان آمار مربوط به بهره‌برداری را می‌توان گردآوری کرد. از سوی دیگر، اگر میانگین مدت پاسخ «سیستم» یا نسبت کامیونهایی را که بیش از ۳۰ دقیقه در «سیستم» می‌مانند برآورد می‌کردیم، به‌طوری که «سیستم» صف بارگیری و دستگاههای بارگیری و صف توزین و قپان را در برگیرد، نهادهای کامیون (DTi) به‌اضافهٔ یک ویژگی، یعنی زمان ورود به صف بارگیری مورد نیاز خواهد بود. با ترک قپان از سوی یک کامیون، مدت پاسخ این کامیون را ممکن بود به‌صورت زمان کنونی شبیه‌سازی (t) منهای ویژگی زمان ورود محاسبه کرد. این مدت پاسخ جدید برای تازه کردن آمار تجمعی مورد استفاده قرار می‌گیرد: $S =$ مجموع مدت پاسخ تمام کامیونهایی که سراسر «سیستم» را پیموده‌اند و $F =$ تعداد مدتهای پاسخ کامیونها که بیش از ۳۰ دقیقه است. مثال اخیر، مجدداً این نکته را به نمایش می‌گذارد که میزان پیچیدگی مدل تا حدی به معیارهای عملکردی که برآورد می‌شود بستگی دارد. ■

۲-۳ زبانهای برنامه‌نویسی برای شبیه‌سازی سیستمهای گسسته پیشامد

زبانهای کامپیوتری شبیه‌سازی، به‌طور قابل‌توجهی ایجاد و اجرای شبیه‌سازی سیستمهای پیچیدهٔ واقعی را تسهیل می‌کنند. به‌طور کلی، هر زبان نسبت به اوضاع واقعی یک جهت‌گیری یا «نگرش کلی» دارد که می‌توان آن را به مانند بحث ارائه شده در بخش ۳-۱ به نگرش پیشامدگرا یا نگرش پردازشگرا رده‌بندی کرد. به هنگام استفاده از یکی از این زبانها، مدل حاصله نگرش پیشامدگرا یا نگرش پردازشگرا یا احتمالاً، ترکیبی از هر دو نگرش خواهد داشت. در زیر بخشهای بعد، چهار زبان FORTRAN، SIMSCRIPT، GPSS، و SLAM همراه با برخی جزئیات آنها به بحث گذاشته می‌شود. زبان GASP نیز به‌اختصار شرح داده می‌شود.

FORTRANیک زبان برنامه‌نویسی علمی است و مشخصاً برای استفاده در شبیه‌سازی طراحی نشده است. به هنگام استفاده از FORTRAN، تحلیلگر احتمالاًگرایش زمانبندی پیشامدها را برمی‌گزیند. در سوی دیگر این طیف، GPSS قرار دارد که یک زبان خاص و بسیار منسجم شبیه‌سازی و زبانی نهادگراست، که این، مورد خاصی از نگرش پردازشگرا در حالت کلی آن است. GPSS برای شبیه‌سازی کردن نسبتاً سادهٔ سیستمهای صف، از قبیل سیستمهای صف کارگاهی طراحی شد. FORTRAN و GPSS در سطح وسیعی برای طراحی مدل شبیه‌سازی گسسته پیشامد مورد استفاده قرار می‌گیرد.

SIMSCRIPT و SLAM زبانهای سطح بالای برنامه‌نویسی شبیه‌سازی است با ساختاری که به منظور تسهیل مدلسازی طراحی شده است. SIMSCRIPTII.۵ و SLAM هر دو، انتخاب یکی از دوگرایش را در اختیار تحلیلگر قرار می‌دهد، یا توسط آنها می‌توان مدلی با استفادهٔ توأم از دو گرایش ساخت. برخلاف FORTRAN، این دو زبان امکاناتی مانند مدیریت فهرست پیشامدهای آتی و دیگر مجموعه‌ها، مولدهای تعبیه شدهٔ مقادیر تصادفی و برنامه‌های تعبیه شدهٔ گردآوری آمار را فراهم می‌آورد، برخلاف GPSS، محاسبات پیچیده توسط هر یک از این دو زبان به سادگی انجام می‌گیرد. SLAM و نسخه‌ای بسط داده شده از SIMSCRIPT (به نام C-SIMSCRIPT) توانایی انجام شبیه‌سازیهای پیوسته را نیز فراهم می‌آورد، یعنی مدلسازی از سیستمهایی که متغیرهای وضعیت با تغییرات پیوسته دارند. SLAM مبتنی بر FORTRAN است و به‌عنوان زیرمجموعه‌ای از خود، زبانی GASP گونه دارد. GASP که در زیر بخش ۳-۲-۲ شرح داده شده است، مجموعه‌ای از زیر برنامه‌های FORTRAN برای تسهیل شبیه‌سازیهای دارای نگرش پیشامدگراست که به زبان FORTRAN نوشته می‌شود. جزء مربوط به نگرش پردازشگرا در SLAM به‌اختصار در زیربخش ۳-۲-۵ شرح داده شده است. از سوی دیگر، SIMSCRIPT به‌عنوان زیرمجموعهٔ خود، یک زبان کامل برنامه‌نویسی علمی و قابل مقایسه با FORTRAN، PL/1، یا ALGOL است. زیربخش ۳-۲-۳ مثال کاملی از کاربرد SIMSCRIPT از نقطه‌نظر زمانبندی پیشامدها ارائه می‌کند. از هر چهار زبان برای شبیه‌سازی نسخه‌ای اصلاح شده از مثال ۲-۱ استفاده شده است.

■ مثال ۳-۵ (باجهٔ پرداخت: صف تک خدمت‌دهندهٔ نمونه‌وار)
سیستم، یعنی باجهٔ پرداخت یک فروشگاه مواد غذایی و لوازم خانگی، به‌صورت یک صف تک خدمت‌دهنده مدلسازی می‌شود. شبیه‌سازی آن‌قدر ادامه می‌یابد تا به ۱۰۰۰ مشتری خدمت داده شود. به‌علاوه، فرض کنید که مدتهای بین دو ورود مشتریها توزیع نمایی دارد با میانگین ۴٫۵ دقیقه و مدتهای خدمتدهی توزیع (تقریباً) نرمال با میانگین ۳٫۲ دقیقه و انحراف معیار ۰٫۶ دقیقه دارد. (تقریب این است که مدتهای خدمتدهی همواره مثبت است.) هرگاه صندوقدار مشغول باشد، صفی تشکیل می‌شود بدون اینکه مشتری از سیستم رانده شود. این مسأله در مثالهای ۳-۲ و ۳-۳ با استفاده از نقطه‌نظر زمانبندی پیشامدها، با دست شبیه‌سازی شد. مدل دو پیشامد دارد، پیشامد ورود و پیشامد ترک. شکلهای ۳-۵ و ۳-۶ منطق پیشامد را ارائه می‌کند. چهار زیربخش بعد، شبیه‌سازی این صف تک خدمت‌دهنده را به زبان FORTRAN، SIMSCRIPT، GPSS و SLAM نمایش می‌دهد. گرچه این مثال بسیار ساده‌تر از مدلهایی است که در بررسی سیستمهای پیچیده مطرح می‌شود، شبیه‌سازی آن دربردارندهٔ اجزای شبیه‌سازی سیستمهای پیچیده‌تر نیز هست.
■

۳-۲-۱ شبیه‌سازی به زبان FORTRAN

FORTRAN یک زبان برنامه‌نویسی است که در سطح وسیعی شناخته شده و در دسترس است، اما هیچ تسهیلاتی که مستقیماً هدف کمک‌رسانی به شبیه‌ساز داشته باشد را ارائه نمی‌دهد. شبیه‌ساز ناگزیر است الگوریتم زمانبندی پیشامدها و جلوبری زمان، توانایی گردآوری آمار، تولید نمونه‌ها از توزیعهای مشخص احتمال و مولد گزارش را، خود برنامه‌نویسی کند. (اما، چند مجموعه از زیربرنامه‌های علمی، مانند IMSL حاوی مولدهای متعدد مقادیر تصادفی است.) استفاده از زبان FORTRAN برای مدلهای بزرگ ممکن است کاملاً دشوار باشد؛ به‌علاوه، ممکن است این امر به مدلهایی بیانجامد که غلطگیری آن مشکل و اجرای آن کُند باشد مگر اینکه برنامه‌نویس نحوهٔ برخوردی به‌دقت سازمان یافته در پیش گیرد و از روشهای کارای پردازش فهرستها شناخت داشته باشد. برای مدلهای کوچک، می‌توان شبیه‌سازی با زبان FORTRAN (یا هر زبان هم‌منظوره دیگر) را به‌عنوان ابزار فراگیری مفاهیم الگوریتم زمانبندی پیشامدها و جلوبری زمان به‌کار گرفت. زبانهای ویژهٔ شبیه‌سازی عموماً ریزه‌کاریهای زمانبندی پیشامدها را از نظر پنهان می‌دارد.

هر مدل شبیه‌سازی گسسته پیشامد نوشته شده به زبان FORTRAN دربردارندهٔ اجزاء مورد بحث در بخش ۳-۱ یعنی حالت سیستم، نهادها و ویژگیها، مجموعه‌ها، پیشامدها، فعالیتها و تأخیرها، به‌اضافهٔ اجزاء مندرج در فهرست زیر است. به منظور تسهیل ایجاد مدل FORTRAN و غلطگیری از آن، بهترین راه این است که مدل با به‌کارگیری زیر برنامه‌ها، به‌گونه‌ای نامتمرکز سازماندهی شود. اجزاء زیر در تقریباً تمام مدلهای نوشته به زبان FORTRAN مشترک است:

CLOCK. متغیری که زمان شبیه‌سازی شده را تعریف می‌کند

زیربرنامهٔ راه‌اندازی. برنامه‌ای که برای تعریف حالت سیستم در زمان صفر به‌کار می‌رود

زیربرنامهٔ جلوبری زمان. برنامه‌ای که برای یافتن پیشامد بعدی (به نام پیشامد قریب‌الوقوع و معروف به (IMEVT) فهرست پیشامدهای آتی (FEL) را جستجو می‌کند و ساعت را به زمان وقوع پیشامد قریب‌الوقوع جلومی‌برد

زیربرنامهٔ زمانبندی. برنامه‌ای که پیشامدهای آتی تولیدشده را در FEL قرار می‌دهد(این مورد در مثال ۳-۶ مورد استفاده قرار نگرفت)

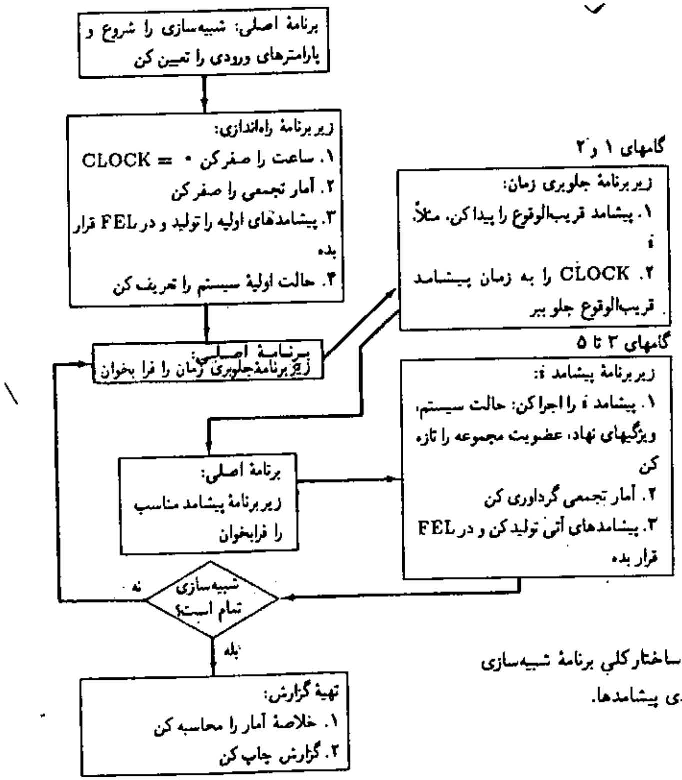
زیربرنامه‌های پیشامدها. زیربرنامه‌های مربوط به هر نوع پیشامد برای تازه کردن حالت سیستم (و آمار تجمعی) هرگاه آن پیشامد رخ دهد

مولدهای مقادیر تصادفی. برنامه‌هایی برای تولید نمونه از توزیعهای احتمال مورد نظر

برنامهٔ اصلی. کنترل کلی بر الگوریتم زمانبندی پیشامدها را فراهم می‌آورد

تهیهٔ گزارش. برنامه‌ای که از آمار تجمعی، آمار خلاصه را محاسبه و در پایان شبیه‌سازی گزارشی چاپ می‌کند.

ساختار کلی برنامهٔ شبیه‌سازی FORTRAN در شکل ۳-۸ نشان داده شده است. این نمودار جریان بسط یافتهٔ الگوریتم زمانبندی پیشامدها و جلوبری زمان است که به اختصار



شکل ۳-۸ ساختار کلی برنامهٔ شبیه‌سازی با دید زمانبندی پیشامدها.

در شکل ۳-۲ ارائه شد (گامهای مذکور در شکل ۳-۸ به پنج گام شکل ۳-۲ اشاره دارد.)

شبیه‌سازی با تنظیم ساعت (CLOCK) شبیه‌سازی روی صفر، قرار دادن صفر در محلهای گردآوری اطلاعات تجمعی، تولید پیشامدهای اولیه (همواره، دستکم، یک پیشامد از این‌گونه وجود دارد) و قرار دادن آنها در FEL، و تعریف حالت سیستم در لحظهٔ صفر شروع می‌شود. پس از این، برنامهٔ شبیه‌سازی آن‌قدر بین زیربرنامهٔ جلوبری زمان و زیربرنامه‌های مربوط به پیشامدها می‌گردد تا شبیه‌سازی به پایان برسد. زیربرنامهٔ جلوبری زمان به منظور یافتن پیشامد قریب‌الوقوع که مثلاً پیشامدی از نوع i است، به جستجوی FEL می‌پردازد. سپس CLOCK شبیه‌سازی به زمان رویداد پیشامد قریب‌الوقوع جلو برده می‌شود. (به یاد دارید که حالت سیستم و ویژگیهای نهاد از لحاظ مقدار در دورهٔ زمانی بین رویداد دو پیشامد متوالی دستخوش تغییر نمی‌شوند. در واقع، این تعریف شبیه‌سازی گسسته پیشامد است: حالت سیستم تنها هرگاه پیشامدی روی دهد تغییر می‌کند.) پس از این، زیربرنامهٔ پیشامد i فراخوانده می‌شود تا پیشامد قریب‌الوقوع را اجرا، آمار تجمعی را تازه و پیشامدهای آتی را تولید کند (تا در FEL قرار داده شوند). اجرای پیشامد قریب‌الوقوع به این معناست که حالت سیستم، ویژگیهای نهاد و اعضای مجموعه‌ها به منظور انعکاس این واقعیت که پیشامد i رخ داده است تغییر می‌پذیرد. توجه داشته باشید که تمام فعل و انفعالات موجود در زیربرنامهٔ هر پیشامد در یک لحظهٔ زمان شبیه‌سازی شده روی می‌دهد. مقدار متغیر CLOCK در زیربرنامهٔ مربوط به پیشامدها تغییر نمی‌کند. اگر شبیه‌سازی پایان نیافته باشد مجدداً کنترل به زیربرنامهٔ جلوبری زمان، سپس به زیربرنامهٔ پیشامد موردنظر، و ... سپرده می‌شود. هرگاه شبیه‌سازی پایان یابدکنترل به زیربرنامهٔ تهیهٔ گزارش سپرده می‌شود تا بر اساس آمار تجمعی گردآوری شده خلاصهٔ آمار مورد نظر را محاسبه کند و گزارشی نیز چاپ کند.

کارایی هر مدل شبیه‌سازی برحسب مدت اجرای کامپیوتری تا اندازهٔ زیادی به‌وسیلهٔ تکنیکهای مورد استفاده در ادارهٔ FEL و سایر مجموعه‌ها تعیین می‌شود. چنانکه قبلاً در بخش ۳-۱ شرح دادیم، بیرون آوردن پیشامد قریب‌الوقوع و افزودن پیشامد جدید دو عمل اصلی است که روی FEL انجام می‌شود. در مثال بعد که تنها دو نوع پیشامد دارد، FEL به روشی نسبتاً ساده اداره می‌شود اما همین روش در مدلی با پیشامدهای فراوان با عدم‌کارایی بسیار همراه خواهد بود.

■ مثال ۳-۶ (شبیه‌سازی صف تک خدمت‌دهنده به زبان FORTRAN)

باجهٔ صندوق یک فروشگاه مواد غذایی که به تفصیل در مثال ۳-۵ معرفی شد را اینک با استفاده از زبان FORTRAN شبیه‌سازی می‌کنیم. نمونه‌ای از این مثال در مثالهای ۳-۲ و ۳-۳ با دست شبیه‌سازی شد که طی آنها حالت سیستم، نهادها و ویژگیها، مجموعه‌ها، پیشامدها، فعالیتها و تأخیرها تحلیل و تشریح شد.

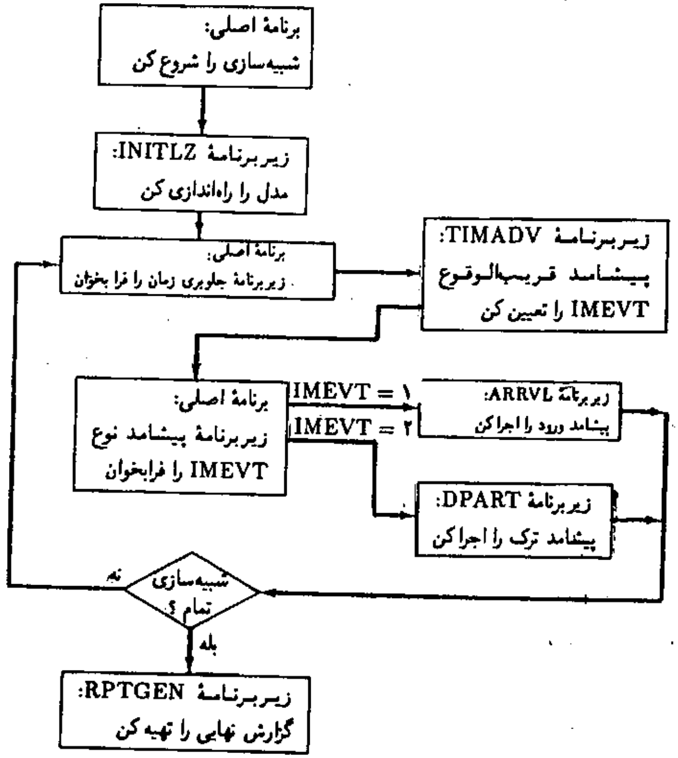
دو پیشامد ورود و ترک، به‌ترتیب، پیشامدهای نوع ۱ و ۲ نامیده می‌شود. نوع پیشامد قریب‌الوقوع به کمک متغیر IMEVT معرفی می‌شود. زیربرنامه‌های مربوط به این مدل و جریان کنترل در

جدول ۳-۷ تعاریف مربوط به متغیرها، توابع، و زیربرنامه‌های موجود در مدل FORTRAN برای صف تک خدمت‌دهنده.

متغیرها		شرح
	حالت سیستم	
LQT		تعداد متقاضی حاضر در صف انتظار در زمان کنونی شبیه‌سازی
LST		تعداد در حال خدمتگیری (۰ یا ۱) در زمان کنونی شبیه‌سازی
	ویژگیهای نهاد و مجموعه‌ها	
CHKOUT(I)		زمان ورود (۱ − I)امین متقاضی به صف خروج بنابراین، CHKOUT(۲) زمان ورود اولین متقاضی به صف است که در حال حاضر به او خدمت داده نمی‌شود.
CHKOUT(۱)		زمان ورود آن متقاضی که در حال حاضر به او خدمت داده می‌شود.
	فهرست پیشامدهای آتی	
FEL(I)		زمان وقوع پیشامد بعد از نوع I(۱,۲ = I)
IMEVT		نوع پیشامد قریب‌الوقوع (۱ یا ۲)
	مدتهای فعالیتها	
IAT		مدت زمان بین ورود متقاضی قبلی و بعدی
SVT		مدت خدمتدهی به آخرین متقاضی که خدمتگیری را شروع می‌کند
	پارامترهای ورودی	
MIAT		میانگین مدت بین دو ورود (۴٫۵ دقیقه)
MSVT		میانگین مدت خدمتدهی (۳٫۲ دقیقه)
SIGMA		انحراف معیار مدت خدمتدهی (۰٫۶ دقیقه)
NCUST		ضابطهٔ توقف‌تعداد متقاضیانی که باید خدمت بگیرند (۱۰۰۰)
	متغیرهای شبیه‌سازی	
CLOCK		مقدار کنونی زمان شبیه‌سازی شده
NUMEVS		تعداد انواع پیشامدها (۲ = NUMEVS)
	متغیرهای مربوط به گردآوری آمار	
TLE		زمان وقوع آخرین پیشامد (برای تازه کردن B بکار می‌رود)
B		مجموع مدت اشتغال خدمت‌دهنده (تاکنون)
MQ		ماکسیمم طول صف انتظار (تاکنون)
S		مجموع مدتهای پاسخ متقاضیانی که سیستم را (تاکنون) ترک کرده‌اند
ND		تعداد موارد ترک (تاکنون)
F		تعداد متقاضیانی که ۴ دقیقه یا بیشتر در باجهٔ صندوق (تاکنون) مانده‌اند
	آمار خلاصه شده	
RHO=B/CLOCK		درصد زمان اشتغال خدمت‌دهنده (در اینجا مقدار CLOCK مقدار نهایی زمان شبیه‌سازی شده است)
MQ		ماکسیمم طول صف انتظار
AVGR=S/ND		متوسط مدت پاسخ
PC4=F/ND		درصد متقاضیانی که ۴ دقیقه یا بیشتر در باجهٔ صندوق مانده‌اند.

جدول ۳-۷ (ادامه)

توابع	شرح
EXPON (FMEAN)	تابعی که نمونه‌هایی از توزیع نمایی با میانگین FMEAN تولید می‌کند
NORML (XMU,SIGMA)	تابعی که نمونه‌هایی از توزیع نرمال با میانگین XMU و انحراف معیار SIGMA تولید می‌کند
زیربرنامه‌ها	**شرح**
INITLZ	زیربرنامهٔ راه‌اندازی
TIMADV	زیربرنامهٔ جلوبری زمان
ARRVL	زیربرنامهٔ پیشامد ورود که پیشامد ورود را اجرا می‌کند
DPART	زیربرنامهٔ اجراکنندهٔ پیشامد ترک
RPTGEN	تهیکنندهٔ گزارش



شکل ۳-۹ ساختار کلی شبیه‌سازی FORTRAN در مورد صف تک خدمت‌دهنده.

شکل ۳-۹ نشان داده شده است که به‌کارگیری شکل ۳-۸ در مورد این مسألهٔ خاص است. جدول ۳-۷ ارائه‌دهندهٔ فهرست متغیرهای FORTRAN مورد استفاده برای حالت سیستم، ویژگیهای نهاد و مجموعه‌ها، مدت فعالیتها، و خلاصهٔ آمار تجمعی؛ همچنین توابع FORTRAN

مورد استفاده در نمونه‌گیری از توزیعهای نمایی و نرمال، و سرانجام تمام زیربرنامه‌های مورد نیاز دیگر است.

برنامهٔ اصلی به شرح شکل ۳-۱۰ جریان کلی الگوریتم زمانبندی پیشامدها و جلوبری زمان را کنترل می‌کند. فهرست متغیرهای سراسری که مقادیرشان برای همهٔ زیربرنامه‌ها شناخته شده است، در دو بلوک COMMON به نامهای SIM و TIMEKP آورده شده است. برنامهٔ اصلی از منطق شکل ۳-۹ پیروی می‌کند. ابتدا پارامترهای ورودی MIAT، MSVT، SIGMA و NCUST، و تعداد پیشامدها، NUMEVS، مشخص می‌شود. سپس، زیربرنامهٔ INITLZ فراخوانده می‌شود تا مدل را راه‌اندازی کند. بعد، زیربرنامهٔ TIMADV به منظور تعیین پیشامد قریب‌الوقوع و جلو بردن ساعت به زمان وقوع این پیشامد فراخوانده می‌شود. پس از این، زیربرنامهٔ پیشامد موردنظر فراخوانده می‌شود. (اگر IMEVT = ۱ باشد جملهٔ GO TO کنترل را به CALL ARRVL و اگر IMEVT = ۲ باشد کنترل را به CALL DPART واگذار می‌کند.) بعد از اینکه پیشامد موردنظر اجرا شد، بررسی می‌کنیم که آیا شبیه‌سازی تمام است یا نه. اگر شبیه‌سازی تمام نشده باشد، برنامه مکرراً بین جلوبری زمان و اجرای پیشامد می‌گردد تا ضابطهٔ توقف تأمین شود. سرانجام وقتی شبیه‌سازی تمام می‌شود، زیربرنامهٔ RPTGEN فراخوانده می‌شود تا گزارش نهایی تهیه کند.

فهرست زیربرنامهٔ INITLZ در شکل ۳-۱۱ ارائه شده است. مقادیر اولیهٔ ساعت شبیه‌سازی، حالت سیستم، و سایر متغیرها در این زیربرنامه مشخص شده است. توجه داشته باشید که زمان اولین ورود به‌طور تصادفی توسط زیربرنامهٔ تابعی EXPON تولید و در FEL(۱) ذخیره می‌شود. بنابراین، فرض می‌کنیم که در لحظهٔ شبیه‌سازی شدهٔ CLOCK = ۰، سیستم خالی است. چون سیستم (در CLOCK = ۰) خالی است نمی‌توان هیچ پیشامد ترکی را زمانبندی کرد. بنابراین، زمان وقوع ترک بعدی، FEL(۲) را مساوی با «بینهایت» قرار می‌دهیم (یعنی، مساوی با یک مقدار بسیار بزرگ که در اینجا 10^{30} = ۱٫۰E + ۳۰). بدین ترتیب، اولین پیشامدی که باید رخ دهد یک پیشامد ورود خواهد بود. از سوی دیگر، اگر فرض می‌شد که در زمان شبیه‌سازی شدهٔ CLOCK = ۰ یک متقاضی حاضر بوده و درست در همین لحظه خدمتگیری خود را شروع کرده است انجام تغییرات زیر لازم می‌شد:

LST = ۱

CHKOUT(۱) = CLOCK

FEL(۲) = CLOCK + NORML(MSVT,SIGMA)

اعمال سایر فرضهای مربوط به راه‌اندازی به طریقی همانند انجام می‌گیرد.

```
      PROGRAM FTNSIM(OUTPUT,TAPE6=OUTPUT)
C
C
C     MAIN PROGRAM
C                         ۱) مدل را راه‌اندازی می‌کند.
C
C       ۲) زیربرنامه‌های جلوبری زمان و پیشامد را فرا می‌خواند.
C
C       ۳) زیربرنامهٔ گزارش‌نویسی را برای اتمام شبیه‌سازی فرا می‌خواند.
      REAL MIAT,MSVT
      COMMON /SIM/ MIAT,MSVT,SIGMA,NCUST,LQT,LST,TLE,
     1             CHKOUT(100),B,MQ,S,F,ND
      COMMON /TIMEKP/ CLOCK,IMEVT,NUMEVS,FEL(2)
       NUMEVS=2
C
C       به پارامترهای ورودی مقدار دهید (می‌توان برای سهولت انجام
C
C       کار، این مقادیر را در پرونده‌ای ذخیره و به برنامه وارد کرد)
C
      MIAT  = 4.5
      MSVT  = 3.2
      SIGMA = .6
      NCUST = 1000
C
C     زیربرنامهٔ راه‌اندازی را فرا بخوانید.
C
      CALL INITLZ
C
C       زیربرنامهٔ جلوبری زمان را فرابخوانید تا پیشامد قریب‌الوقوع را
C
C       تعیین کند و ساعت را به زمان پیشامد قریب‌الوقوع جلو برد
C
   30 CALL TIMADV
C
C       متغیر IMEVT پیشامد قریب‌الوقوع را معرفی می‌کند.
C
C       IMEVT = ۱) مصرف ورود است.
C
C       IMEVT = ۲) مصرف ترک است.
      GO TO (40,50),IMEVT
C
C     زیربرنامهٔ پیشامد موردنظر را فرابخوانید
C
   40 CALL ARRVL
C
      GO TO 30
C
   50 CALL DPART
C
C
C       بررسی کنید آیا شبیه‌سازی تمام است. اگر نیست به زیربرنامهٔ
C
C       جلوبری زمان برگردید
      IF(ND .LT. NCUST)GO TO 30
C
C       هرگاه شبیه‌سازی تمام شد، زیربرنامهٔ تهیه گزارش را فرابخوانید
C
      CALL RPTGEN
      STOP
      END
```

شکل ۳-۱۰ برنامهٔ اصلی به زبان FORTRAN برای شبیه‌سازی صف تک خدمت‌دهنده.

```
C
C
C     زیربرنامهٔ راه‌اندازی
C
      SUBROUTINE INITLZ
      REAL MIAT,MSVT
      COMMON /SIM/ MIAT,MSVT,SIGMA,NCUST,LQT,LST,TLE,
     1             CHKOUT(100),B,MQ,S,F,ND
      COMMON /TIMEKP/ CLOCK,IMEVT,NUMEVS,FEL(2)
C
C     شبیه‌سازی را شروع کنید:
C     ۱) ساعت شبیه‌سازی را صفر کنید.
C     ۲) فرض کنید سیستم در لحظهٔ صفر، خالی و بیکار است.
C     ۳) به آمار تجمعی مقدار صفر بدهید.
C
        CLOCK=0.0
        IMEVT = 0
        LQT = 0
        LST = 0
        TLE = 0
        B = 0
        MQ = 0
        S = 0
        F = 0
        ND = 0
C
C
C     زمان اولین ورود، IAT، را تولید و اولین ورود را در FEL(۱)
C     زمانبندی کنید. FEL(۲) را مساوی «بینهایت» قرار دهید تا
C     نشان دهد که وقتی سیستم خالی است ترک آن ممکن نیست
C
C
        FEL(1)=CLOCK + EXPON(MIAT)
        FEL(2)=1.0E+30
      RETURN
      END
```

شکل ۳-۱۱ زیربرنامهٔ راه‌اندازی برای شبیه‌سازی صف تک خدمت‌دهنده به زبان FORTRAN.

فهرست زیربرنامهٔ جلوبری زمان، SUBROUTINE TIMADV، در شکل ۳-۱۲ ارائه شده است. برای پردازش FEL، این زیربرنامه طبق رهیافت «قوهٔ قهریه» عمل می‌کند. به این ترتیب که آرایهٔ FEL، یعنی FEL(۱)، FEL(۲)، ...، FEL(NUMEVS) برای یافتن مقدار می‌نیمم که مثلاً در موقعیت IMEVT قرار دارد، جستجو می‌شود. به این ترتیب پیشامد IMEVT پیشامد قریب‌الوقوع است و در زمان FEL(IMEVT) رخ خواهد داد. زمان شبیه‌سازی که CLOCK


```
C
C
C     زیربرنامهٔ جلوبری زمان پیشامد بعدی را از فهرست
C     پیشامدهای آتی می‌یابد و ساعت را جلو می‌برد
C
C
      SUBROUTINE TIMADV
      REAL MIAT,MSVT
      COMMON /SIM/ MIAT,MSVT,SIGMA,NCUST,LQT,LST,TLE,
     1             CHKOUT(100),B,MQ,S,F,ND
      COMMON /TIMEKP/ CLOCK,IMEVT,NUMEVS,FEL(2)
        FMIN=1.E+29
        IMEVT=0
C
C     فهرست پیشامدهای آتی را برای پیشامد بعد جستجو کنید
C
      DO 30 I=1,NUMEVS
        IF(FEL(I).GE.FMIN)GO TO 30
        FMIN=FEL(I)
        IMEVT=I
   30 CONTINUE
      IF(IMEVT.GT.0)GO TO 50
C
C     اشتباه-فهرست پیشامدهای آتی خالی است
C
        WRITE(6,40)
   40   FORMAT(1X,"*****FUTURE EVENT LIST EMPTY*****",
     1         1X,"***SIMULATION CANNOT CONTINUE***")
        CALL RPTGEN
        STOP
C
C     ساعت شبیه‌سازی را جلو ببرید.
C     پیشامد بعدی از نوع «IMEVT» است که در زمان
C     FEL (IMEVT) رخ خواهد داد
C
C
   50 CLOCK = FEL(IMEVT)
      RETURN
      END
```

شکل ۳-۱۲ زیربرنامهٔ جلوبری زمان به FORTRAN برای شبیه‌سازی صف تک خدمت‌دهنده.

نامیده می‌شود، به‌زمان پیشامد قریب‌الوقوع جلوبرده می‌شود وکنترل به برنامهٔ اصلی بازگردانده می‌شود. توجه داشته باشید که اگر FEL(I) در مورد تمام انواع I پیشامد مساوی «بینهایت» باشد، پیامی چاپ می‌شود مبنی بر اینکه فهرست پیشامدهای آتی خالی است و شبیه‌سازی نمی‌تواند ادامه یابد. (چرا؟) زیرا یا اشتباهی در منطق برنامه‌نویسی روی داده یا برنامه‌نویس عمداً تمام پیشامدهای آتی را لغو کرده است. در هر یک از این موارد، زیربرنامهٔ تهیهٔ گزارش فراخوانده می‌شود. (اگر اشتباهی رخ داده باشد، خروجی مدل می‌تواند در تعیین محل آن کمک کند.)

شکل ۳-۱۳ فهرست زیربرنامهٔ پیشامد ARRVL را نشان می‌دهد. هرگاه پیشامد ورود رخ

```
C
C
C زیربرنامهٔ پیشامد ورود.
C
C
      SUBROUTINE ARRVL
      REAL MIAT,MSVT,IAT
      COMMON /SIM/ MIAT,MSVT,SIGMA,NCUST,LQT,LST,TLE,
     1             CHKOUT(100),B,MQ,S,F,ND
      COMMON /TIMEXP/ CLOCK,IMEVT,NUMEVS,FEL(2)
C
C
C تعیین کنید که آیا خدمت‌دهنده مشغول است
C
      IF(LST .EQ. 1) GO TO 20
C
C خدمت‌دهنده بیکار است. حالت سیستم را تازه و زمان
C
C ورود متقاضی جدید را ثبت کنید
        LST = 1
        CHKOUT(1) = CLOCK
C
C برای ورودی جدید، یک مدت خدمت‌دهی تولید و ترک این
C
C ورود را زمانبندی کنید
        SVT = NORML(MSVT,SIGMA)
        FEL(2) = CLOCK + SVT
C
C آمار تجمعی، MQ (همچنین TLE) را تازه کنید
C
        TLE = CLOCK
        IF(LQT .GT. MQ) MQ = LQT
      GO TO 100
C
C
C خدمت‌دهنده مشغول است. حالت سیستم را تازه و زمان
C
C ورود متقاضی جدید را ثبت کنید
 20     LQT = LQT + 1
        I = LQT + LST
        IF(I .GT. 100) GO TO 200
        CHKOUT(I) = CLOCK
C
C آمار تجمعی، B و MQ را تازه کنید (هرگاه یک ورود رخ دهد،
C
C S، ND و F جدید نمی‌شود.)
C
        B = B + (CLOCK - TLE)
        TLE = CLOCK
        IF(LQT .GT. MQ) MQ = LQT
C
C یک مدت بین دو ورود تولید و پیشامد ورود بعد را زمانبندی کنید
C
C
 100  IAT = EXPON(MIAT)
      FEL(1) = CLOCK + IAT
      RETURN
C
C اشتباه رخ داده است. آرایهٔ CHKOUT سرریز کرده است.
C
C بعد آرایه را افزایش دهید.
C
 200  WRITE(6,205)
 205  FORMAT(1X,"*****OVERFLOW IN ARRAY CHKOUT. INCREASE ",
     1          "DIMENSION.*****",
     2      /,1X,"*****SIMULATION CANNOT CONTINUE*****")
C
      CALL RPTGEN
      STOP
      END
```

شکل ۳-۱۳ زیربرنامهٔ پیشامد ورود برای شبیه‌سازی صف تک خدمت‌دهنده به FORTRAN.

دهد. این زیربرنامه پیشامد ورود را اجرا می‌کند. منطق اساسی پیشامد ورود برای یک صف تک خدمت‌دهنده قبلاً در شکل ۳-۵ ارائه شد. ابتدا وضعیت خدمت‌دهنده (یعنی مشغول یا بیکار بودن او) که با مقادیر به‌ترتیب ۱ و ۰ برای متغیر LST نشان داده شده است تعیین می‌شود. اگر خدمت‌دهنده بیکار باشد آمار تجمعی B، MQ، S، ND و F نیاز به تازه شدن ندارد. وضعیت خدمت‌دهنده به مشغول (LST = ۱) تغییر یافته و زمان ورود در CHKOUT(۱) ثبت می‌شود. توجه کنید که CHKOUT معرف مجموعه‌ای است که ویژگی «زمان ورود» متقاضی را ذخیره می‌کند که بعداً در زیربرنامهٔ DPART برای محاسبهٔ مدت پاسخ متقاضی به‌کار گرفته می‌شود. چون یک خدمتدهی در حال شروع شدن است، یک مدت خدمتدهی (SVT) به‌وسیلهٔ FUNCTION NORML تولید و با قرار دادن زمان ترک در FEL(۲) یک پیشامد ترک زمانبندی می‌شود. توجه کنید که FEL(۲) با زمان فعلی شبیه‌سازی (CLOCK) به‌اضافهٔ مدت آن خدمتدهی‌که در حال شروع است (SVT) مساوی قرار داده می‌شود. کنترل به جملهٔ ۱۰۰ (شکل ۳-۱۳) انتقال می‌یابد. یعنی نقطه‌ای که مدت بین دو ورود (IAT) با استفاده از FUNCTION EXPON تولید و ورود بعدی از طریق محاسبهٔ زمان ورود بعد (CLOCK + IAT) و ذخیره‌سازی آن در محل FEL(۱) از فهرست پیشامدهای آتی زمانبندی می‌شود. سپس کنترل به برنامهٔ اصلی بازگردانده می‌شود. از سوی دیگر، هرگاه ورود روی دهد اگر خدمت‌دهنده مشغول باشد (LST = ۱)، کنترل به جملهٔ ۲۰ (شکل ۳-۱۳) منتقل می‌شود. تعداد افراد در صف انتظار (LQT) یک واحد افزایش می‌یابد و ویژگی زمان ورود متقاضی در «عقب» مجموعهٔ CHKOUT ثبت می‌شود. (آرایهٔ CHKOUT از بُعد ۱۰۰ برخوردار است. بنابراین، تا زمانی که تعداد متقاضی (LQT+LST) در سیستم مساوی ۱۰۰ یا کمتر است، هیچ مسأله‌ای رخ نخواهد داد. توجه کنید که آزمایشی صورت می‌گیرد تا در صورتی‌که I = LQT + LST بزرگتر از ۱۰۰ باشد کنترل به جملهٔ ۲۰۰ انتقال یابد. یک پیام در مورد اشتباه چاپ و زیربرنامهٔ تهیهٔ گزارش، فعال و سپس شبیه‌سازی متوقف می‌شود. ممکن است شبیه‌سازی صحیح باشد ولی سیستم بسیار شلوغتر از آن شود که پیش‌بینی می‌شد. در چنین مواردی بُعد CHKOUT باید مثلاً به ۲۰۰ افزایش داده شود. مقدار به‌کار رفته در جملهٔ IF نیز باید از ۱۰۰ به ۲۰۰ افزایش یابد. چنین حالتی نیز ممکن است پیش آید که شبیه‌سازی دارای غلطی در منطق یا مشخص‌سازی داده‌ها باشد.) سپس آمار تجمعی B و MQ تازه می‌شود. مدت کل اشتغال، B، به‌صورت

$$\text{«B جدید»} = \text{«B قدیم»} + \text{«مدت اشتغال پس از آخرین پیشامد»}$$

یا در زبان FORTRAN به‌صورت

$$B = B + (CLOCK - TLE)$$

تازه می‌شود. به یاد دارید که TLE زمان وقوع پیشامد قبلی است. چون معلوم است که خدمت‌دهنده

در فاصلهٔ زمانی (TLE,CLOCK) مشغول بوده است نتیجه می‌شود که مدت کل اشتغال باید به مقدار (CLOCK − TLE) افزایش یابد. پس از تازه شدن B، باید TLE و MQ را تازه کرد. همانند قبل، ورود بعدی تولید و در FEL زمانبندی می‌شود و سپس کنترل به برنامهٔ اصلی انتقال می‌یابد.

زیربرنامهٔ DPART که پیشامد ترک را اجرا می‌کند در شکل ۳-۱۴ ارائه شده است. نمودار جریان منطق پیشامد ترک قبلاً در شکل ۳-۶ ارائه شد. ابتدا آمار تجمعی B، S، ND و F تازه می‌شود (توجه داشته باشید که با وقوع پیشامد ترک، مقدار ماکسیمم طول صف، MQ، ممکن نیست تغییر کند.) مدت پاسخ، RT، برای متقاضی در حال ترک به‌صورت

$$RT = \text{(زمان کنونی)} - \text{(زمان ورود متقاضی در حالت ترک)}$$

یا

$$RT = CLOCK - CHKOUT(۱)$$

محاسبه می‌شود. سپس به مدت پاسخ تجمعی S و تعداد موارد ترک ND افزوده می‌شود. به تعداد متقاضیانی که با مدت پاسخ ۴ دقیقه یا بیشتر روبه‌رو می‌شوند، افزوده می‌شود، یعنی اگر $RT \geq ۴/۰$ باشد، یک واحد به F اضافه می‌شود. پس از این صف انتظار را بازرسی می‌کنیم (آیا $LQT \geq ۱$ هست یا نه) تا معلوم شود یک متقاضی در انتظار برای شروع خدمت وجود دارد یا نه. اگر چنین نباشد (یعنی، اگر $LQT = ۰$ باشد)، وضعیت خدمت‌دهنده را مساوی با صفر ($LST = ۰$) می‌گیریم و برای تضمین اینکه بعداً یک ورود رخ دهد، پیشامد بعدی ترک را مساوی با «بینهایت» (یعنی $۱/۰E + ۳۰$) قرار می‌دهیم و سپس، کنترل به برنامهٔ اصلی انتقال می‌یابد. اگر یک متقاضی در صف انتظار (یعنی اگر $LQT \geq ۱$) باشد، تمام متقاضیان «یک موقعیت به جلو رانده می‌شوند». یعنی ویژگیهای زمان ورود در آرایهٔ CHKOUT به جلو رانده می‌شود. با اتمام این «جلورانی»، CHKOUT(۱) مجدداً دربردارندهٔ زمان ورود متقاضی (جدید) در حال خدمتگیری، CHKOUT(۲) دربردارندهٔ زمان ورود اولین متقاضی حاضر در صف (پشت سر متقاضی در حال خدمتگیری)؛ و ... است. پس از این، حالت سیستم با کاستن یک واحد از تعداد حاضر در صف، LQT تازه می‌شود. سرانجام، مدت خدمتگیری (SVT) آن متقاضی که خدمتدهی به او در حال شروع است تولید و پیشامد ترک این متقاضی با مساوی قرار دادن FEL (۲) با (زمان کنونی) + (مدت خدمتدهی)

یا

$$FEL(۲) = CLOCK + SVT$$

زمانبندی می‌شود. سپس کنترل به برنامهٔ اصلی بازگردانده می‌شود.

زیربرنامهٔ تهیهٔ گزارش، RPTGEN، در شکل ۳-۱۵ ارائه شده است. خلاصهٔ آمار، RHO، AVGR و PC۴ طبق فرمولهای مندرج در جدول ۳-۷ محاسبه می‌شود. سپس پارامترهای

```
C
C
C   زیربرنامهٔ ترک
C
C
      SUBROUTINE DPART
      REAL MIAT,MSVT
      COMMON /SIM/ MIAT,MSVT,SIGMA,NCUST,LQT,LST,TLE,
     1              CHKOUT(100),B,MQ,S,F,ND
      COMMON /TIMEKP/ CLOCK,IMEVT,NUMEVS,FEL(2)
C
C        آمار تجمعی یعنی B، S، ND، و F را جدید کنید.
C
C        (چون LQT کم می‌شود، در این لحظه MQ تغییر نمی‌کند).
C
      B = B + (CLOCK - TLE)
      TLE = CLOCK
      RT = CLOCK - CHKOUT(1)
      S = S + RT
      ND = ND + 1
      IF(RT .GE. 4.0) F = F + 1
C
C   وضعیت صف انتظار را بررسی کنید
C
      IF(LQT .GE. 1) GO TO 20
C
C        چون هیچ متقاضی در صف نیست، خدمت‌دهنده آزاد می‌شود و
C        زمان ترک بعدی را بی‌نهایت می‌گیریم.
C
        LST = 0
        FEL(2) = 1.E+30
      RETURN
C
C        دستکم، یک متقاضی در صف است، پس هر متقاضی حاضر
C        در صف را یک خانه به جلو برانید.
C
   20 DO 30 I = 1,LQT
        I1 = I + 1
        CHKOUT(I) = CHKOUT(I1)
   30 CONTINUE
C
C   حالت سیستم را تازه کنید
C
      LQT = LQT - 1
C        برای آن متقاضی که شروع به خدمتگیری می‌کند، مقدار خدمت‌دهی
C        تازه‌ای تولید کرده و پیشامد ترک بعدی را زمانبندی کنید.
C
C
      SVT = NORML(MSVT,SIGMA)
      FEL(2) = CLOCK + SVT
      RETURN
      END
```

شکل ۳-۱۴ زیربرنامهٔ FORTRAN پیشامد ترک برای شبیه‌سازی صف تک‌خدمت‌دهنده.

```
C
C
C  تهیه‌کنندهٔ گزارش
C
C
      SUBROUTINE RPTGEN
      REAL MIAT,MSVT
      COMMON /SIM/ MIAT,MSVT,SIGMA,NCUST,LQT,LST,TLE,
     1             CHKOUT(100),B,MQ,S,F,ND
      COMMON /TIMEXP/ CLOCK,IMEVT,NUMEVS,FEL(2)
C
C  محاسبهٔ خلاصهٔ آمار
C
      RHO = B/CLOCK
      AVGR = S/ND
      PC4 = F/ND
C
C  چاپ خروجیها
C
      WRITE(6,10)
   10 FORMAT(6X,"SINGLE SERVER QUEUE SIMULATION - GROCERY"
     1       " STORE CHECKOUT COUNTER")
C
      WRITE(6,20) MIAT,MSVT,SIGMA,NCUST
   20 FORMAT(//,17X,"MEAN INTERARRIVAL TIME          ",F10.2,
     1        /,17X,"MEAN SERVICE TIME               ",F10.2,
     2        /,17X,"STANDARD DEVIATION OF SERVICE TIMES",F5.2,
     3        /,17X,"NUMBER OF CUSTOMERS SERVED         ",I6)
C
      WRITE(6,30)RHO,MQ,AVGR,PC4,CLOCK,ND
   30 FORMAT(///,17X,"SERVER UTILIZATION        ",F8.2,
     1        /,17X,"MAXIMUM LINE LENGTH       ",I8
     2        /,17X,"AVERAGE RESPONSE TIME     ",F8.2," MINUTES"
     3        /,17X,"PROPORTION WHO SPEND FOUR",
     4        /,19X,"MINUTES OR MORE IN SYSTEM",F6.2,
     5        /,17X,"SIMULATION RUNLENGTH      ",F8.2," MINUTES"
     6        /,17X,"NUMBER OF DEPARTURES      ",I8)
      RETURN
      END
```

شکل ۳-۱۵ زیربرنامهٔ FORTRAN برای تهیهٔ گزارش شبیه‌سازی صف تک خدمت‌دهنده.

```
C
C
C  مولد مقادیر تصادفی نمایی
C
C
      FUNCTION EXPON(FMEAN)
C
C  یک عدد تصادفی از U(۰,۱) تولید کنید
      R=RANF(DUMMY)
C
C  یک مقدار تصادفی از توزیع نمایی منفی با میانگین FMEAN
C  تولید کنید (معادلهٔ (۸-۲ (ب)) را ببینید).
C
      EXPON = -FMEAN*ALOG(R)
      RETURN
      END
```

شکل ۳-۱۶ مولد مقدار تصادفی نمایی برای شبیه‌سازی صف تک خدمت‌دهنده.

ورودی MIAT، MSVT، SIGMA، و NCUST و به دنبال آنها خلاصهٔ آمار چاپ می‌شود. چاپ پارامترهای ورودی در پایان شبیه‌سازی به منظور حصول اطمینان از صحت مقادیرشان و اینکه به‌طور ناخواسته دچار تغییر نشده باشد فکر خوبی است.

فهرست زیربرنامهٔ تابعی EXPON در شکل ۳-۱۶ و فهرست زیربرنامهٔ تابعی NORML

در شکل ۳-۱۷ ارائه شده است. هر دوی این توابع، ابتدا تابع RANF را فرا می‌خواند که زیربرنامه‌ای کتابخانه‌ای (و در دسترس برخی کامپیوترها) است و نمونه‌هایی با توزیع یکنواخت در فاصلهٔ (۰,۱) تولید می‌کند. این‌گونه برنامه‌ها، که مولدهای اعداد تصادفی نامیده می‌شود، در فصل ۷ مورد بحث قرار گرفته است. تکنیکهای تولید مقادیر تصادفی برخوردار از توزیع نمایی و نرمال که در فصل ۸ تشریح می‌شود، مبتنی بر تولید اولیهٔ یک عدد تصادفی، R، با توزیع $U(۰,۱)$ است. برای توضیح بیشتر خواننده به فصلهای ۷ و ۸ رجوع کند.

خروجی حاصل از شبیه‌سازی باجهٔ صندوق فروشگاه مواد غذایی در شکل ۳-۱۸ نشان داده شده است. باید تأکید کرد که آمار خروجی، برآوردهای آماری و دارای خطای تصادفی است. مقادیر نشان داده شده تحت تأثیر اعداد تصادفی خاصی که به‌طور تصادفی مورد استفاده قرار گرفته و همچنین شرایط شروع در زمان صفر و مدت اجرا (در این مورد، معادل ۱۰۰۰ ترک) واقع می‌شود. در فصل ۱۱ روشهای براورد انحراف استاندارد این‌گونه براوردها مورد بحث قرار می‌گیرد.

در برخی شبیه‌سازیها این خواسته تعقیب می‌شود که شبیه‌سازی پس از یک مدت زمان ثابت، مثلاً دقیقه ۷۲۰ = ساعت ۱۲ = TE متوقف شود. در این مورد، یک پیشامد اضافی، یعنی پیشامد متوقف‌کننده (مثلاً، نوع ۳) را تعریف و با مساوی قرار دادن FEL(۳) با TE وقوع آن را برنامه‌ریزی می‌کنند. هرگاه پیشامد متوقف‌کننده رخ دهد، آمار تجمعی تازه و زیربرنامهٔ تهیهٔ گزارش فراخوانده خواهد شد. برنامهٔ اصلی و زیربرنامهٔ INITLZ نیز تغییرات کوچکی را طلب می‌کند. مشخصاً، NUMEVS باید مساوی ۳ قرار داده شود و جملهٔ GO TOی محاسباتی در برنامهٔ اصلی باید تغییر یابد. تمرین ۴ از خواننده می‌خواهد که این تغییرات را اعمال کند. تمرین ۵ این تغییر اضافی را ملاحظه می‌کند که هر متقاضی حاضر در باجهٔ صندوق در زمان شبیه‌سازی CLOCK = TE باید اجازهٔ ترک فروشگاه را داشته باشد ولی هیچ ورود جدیدی پس از زمان TE مجاز نباشد. ■

۳-۲-۲ شبیه‌سازی با GASP

GASP IV مجموعه‌ای است از زیربرنامه‌های FORTRAN که به منظور تسهیل شبیه‌سازی مبتنی بر زمانبندی پیشامدها به زبان FORTRAN طراحی شده است. این مجموعه شامل بیش از ۳۰ زیربرنامه و تابع است که امکانات مورد نیاز فراوانی از جمله یک برنامهٔ جلوبری زمان (به نام GASP)؛ برنامه‌های ادارهٔ فهرست پیشامدهای آتی (یعنی افزودن پیشامدهای جدید به فهرست پیشامدهای آتی)؛ برنامه‌های افزودن وکاستن نهادها از مجموعه‌ها؛ برنامه‌های گردآوری آمار، برنامه‌های تولید مقدار تصادفی؛ و یک برنامهٔ استاندارد تهیهٔ گزارش را فراهم می‌آورد. برنامه‌نویس باید یک برنامهٔ اصلی، یک برنامه راه‌اندازی مدل، برنامه‌های مربوط به پیشامدها و در صورت تمایل، یک برنامهٔ تهیهٔ گزارش، به‌اضافهٔ یک زیربرنامه به نام EVNTS را خود فراهم کند. برنامهٔ اصلی باید دارای جملهٔ CALL GASP باشد تا شبیه‌سازی را شروع کند. زیربرنامهٔ GASP پیشامد قریب‌الوقوع را تعیین و از طریق مقدار شاخص خود (مثل شاخص IMEVT در مدل

```
C
C                مولد مقادیر تصادفی نرمال
C
      FUNCTION NORML(MEAN,SIGMA)
      REAL MEAN,SIGMA
      DATA K/0/,PI/3.14159/
C
C                تعیین کنید کدام مقدار تصادفی نرمال استاندارد باید استفاده شود.
C
      IF(K.EQ.1)GO TO 10
C
C                دو عدد تصادفی تولید کنید.
C
        RONE=RANF(DUMMY)
        RTWO=RANF(DUMMY)
C
C                دو مقدار تصادفی از نرمال استاندارد تولید کنید (معادلات
C                (۸-۲۶) را ببینید).
C
        ZONE=SQRT(-2*ALOG(RONE)) * COS(2*PI*RTWO)
        ZTWO=SQRT(-2*ALOG(RONE)) * SIN(2*PI*RTWO)
C
C                مقدار تصادفی نرمال برخوردار از میانگین MEAN و
C                انحراف معیار SIGMA را محاسبه کنید.
C
        NORML = ZONE*SIGMA + MEAN
C
        K = 1
      RETURN
C
C                مقدار تصادفی نرمال را محاسبه کنید.
C
   10   NORML = ZTWO*SIGMA + MEAN
        K = 0
C
      RETURN
      END
```

شکل ۳-۱۷ مولد مقدار تصادفی نرمال برای شبیه‌سازی صف تک خدمت‌دهنده.


```
MEAN INTERARRIVAL TIME                     4.50
MEAN SERVICE TIME                          3.20
STANDARD DEVIATION OF SERVICE TIMES         .60
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 ۳-۱۸ خروجی برنامهٔ FORTRAN برای شبیه‌سازی صف تک خدمت‌دهنده.

FORTRAN زیربخش ۳-۲-۱ که در GASP آن را NEXT نام نهاده‌اند) زیربرنامه‌ای نوشتهٔ کاربر، یعنی EVNTS، را فرا می‌خواند. این شاخص (یعنی NEXT) تعیین می‌کند پیشامد قریب‌الوقوع کدام است تا به‌وسیلهٔ EVNTS فراخوانده شود.

GASP IV برای اکثر کامپیوترهایی که از همگردان FORTRAN برخوردارند در دسترس است. شرحی کامل در این مورد به‌وسیلهٔ پریتسکر [۱۹۷۴] ارائه شده است. لا و کلتون [۱۹۸۲] نیز توضیحی مختصر همراه با یک مثال ارائه داده‌اند.

۳-۲-۳ شبیه‌سازی با SIMSCRIPT

SIMSCRIPT II.5 یک زبان سطح بالای برنامه‌نویسی با امکاناتی است که مشخصاً برای ایجاد مدل شبیه‌سازی گسسته پیشامد طراحی شده است. به‌عنوان یک زبان شبیه‌سازی، نقطه‌نظر زمانبندی پیشامدها و پردازش‌ـتقابل را مجاز می‌دارد. به‌عنوان یک زبان علمی، دستکم به توانمندی FORTRAN، PL/1، ALGOL، یا پاسکال است. بسیاری از کارهای برنامه‌نویسی به زبان SIMSCRIPT با کارایی بیشتری (یعنی صرف وقت کمتر برنامه‌نویس) نسبت به FORTRAN انجامپذیر است. هر برنامهٔ SIMSCRIPT به جملات شبیه زبان انگلیسی و با شیوهٔ قالب‌بندی آزاد قابل نوشتن است؛ یک چنین برنامه تقریباً خود مستند ساخته است و به راحتی برای افرادی غیر از برنامه‌نویس قابل تشریح است. برخلاف FORTRAN، SIMSCRIPT امکان نگهداری خودکار فهرست پیشامدهای آتی و الگوریتم جلوبری زمان و زمانبندی پیشامدها؛ نگهداری خودکار مجموعه‌ها، شامل عملیات افزودن و حذف نهادها از مجموعه‌ها؛ گردآوری خودکار آمارهای مورد نیاز و مولدهای متعدد مقادیر تصادفی انواع وسیعی از توزیعهای احتمال را فراهم می‌کند.

SIMSCRIPT را در ابتدا شرکت RAND در دههٔ ۱۹۶۰ ایجاد کرد و در اصل مبتنی بر FORTRAN بود. آخرین نسخهٔ این زبان، یعنی SIMSCRIPT II.5 در تملک شرکت

CACI (لس‌آنجلس و واشنگتن دی‌سی) است و برای اکثر سیستمهای کامپیوتر بزرگ (آی‌بی‌ام ۳۶۰/۳۷۰، CDC/۶۰۰۰-۷۰۰۰، VAX، هانی‌ول ۶۰۰۰ – H/۶۰۰، و یونیوک ۱۱۰۰) در دسترس و قابل اجاره یا خرید از طریق CACI است. این زبان از توانایی شبیه‌سازی به طریق پردازش-تقابل، همچنین به طریق زمانبندی پیشامدها که در اینجا تشریح شده، برخوردار است و نسخه‌ای تکمیل شده از آن وجود دارد که اجازهٔ شبیه‌سازیهای پیوسته را نیز می‌دهد [دلفاس، ۱۹۷۶]. CACI چند جزوهٔ کوچک عرضه می‌کند که حاوی شرح کلی زبان به‌اضافهٔ مثالهای ساده است [راسل و انیتو، ۱۹۷۹؛ راسل، ۱۹۷۶]. منابع کاملتر زیر نیز دسترسپذیر است: کیویات و همراهان [۱۹۷۳]، راسل [۱۹۸۳] و کتاب مأخذ SIMSCRIPT II.5 از انتشارات CACI [۱۹۷۶]. علاوه بر اینها، CACI به‌طور متناوب سمینارهایی در مورد SIMSCRIPT II.5 عرضه می‌کند.

نگرش کلی برگزیدهٔ SIMSCRIPT مبتنی بر نهادها، ویژگیها، و مجموعه‌هاست. نهادها به دائمی و موقت رده‌بندی می‌شود. نهادهای دائمی معرف عناصری در سیستم است که در دورهٔ شبیه‌سازی در سیستم می‌ماند. مثالها شامل تعداد ثابت خدمت دهندگان ثابت در یک مدل صف، تعداد ثابت کشتیها در یک مدل کشتیرانی، یا تعداد ثابت کامیونهای کمپرسی در مثال ۳-۴ است. نهادهای موقت معرف عناصری از قبیل متقاضیان در یک مدل صف است که به سیستم «وارد می‌شوند». مدتی می‌مانند و سپس سیستم را «ترک می‌گویند». در خلال دورهٔ شبیه‌سازی، تعداد نهادهای موقت فعال در مدل می‌تواند به‌طور قابل‌ملاحظه‌ای تغییر کند. نهادها ممکن است ویژگیهایی داشته باشد و نهادهای نظیر ممکن است (درست همانند شبیه‌سازیهای دستی مثالهای ۳-۳ و ۳-۴) متعلق به یک مجموعه باشد. هر برنامهٔ SIMSCRIPT از یک دیباچه، یک برنامهٔ اصلی، برنامه‌های پیشامدها، و زیربرنامه‌های متعارف تشکیل می‌شود. همان‌طور که اشاره شد، برنامهٔ جلوبری زمان، برنامه‌های تولید مقادیر تصادفی، و برنامه‌های گردآوری آمار، به‌طور خودکار موجود است. دیباچه شرحی ایستا از سیستم را از طریق تعریف تمام نهادها، ویژگیهای آنها و مجموعه‌هایی که نهادها احتمالاً به آنها متعلق است ارائه می‌دهد. دیباچه همچنین متغیرهای سراسری (مورد استفاده در تعریف جزیی حالت سیستم) را تعریف و آماری را که در مورد برخی متغیرها لازم است گردآوری شود، مشخص می‌کند، مجموعهٔ وسیعی از متغیرها به‌طور خودکار نگهداری می‌شود. مثلاً، TIME.V معرف ساعت شبیه‌سازی است. اگر نام یک مجموعه QUEUE باشد، N.QUEUE تعداد نهادهای درون مجموعه است. (توجه داشته باشید که نامها در SIMSCRIPT ممکن است هر طولی داشته باشد و ممکن است نقاط در میان گرفته شده داشته باشند.) برنامهٔ اصلی، مقدار پارامترهای ورودی را می‌خواند (یا مشخص می‌کند)، حالت سیستم در شروع کار را مشخص می‌کند، و اولین پیشامدها را تولید می‌کند. برنامه‌های پیشامدها با زیربرنامهٔ جلوبری زمان به‌طور خودکار فراخوانده می‌شود که این زیربرنامه به‌نوبهٔ خود توسط جملهٔ "START SIMULATION" در برنامهٔ اصلی فعال می‌شود. زیربرنامه‌های متعارف را می‌توان از هر زیربرنامهٔ پیشامد یا از برنامهٔ اصلی فراخواند.

■ **مثال ۳-۷ (شبیه‌سازی صف تک خدمت‌دهنده به زبان SIMSCRIPT)**

اینک یک مدل SIMSCRIPT باجهٔ صندوق فروشگاه مواد غذایی (مثال ۳-۵) شرح داده می‌شود. در حل FORTRAN، اولین زمان ورود یک زمان تصادفی منتخب از توزیع نمایی مدتهای بین دو ورود متوالی بود؛ در مدل SIMSCRIPT فرض می‌شود که اولین ورود در زمان صفر رخ می‌دهد. صرف‌نظر از این مورد، در دو مدل فرضهای یکسانی را می‌پذیریم.

دیباچه در شکل ۳-۱۹ ارائه شده است. جملات درون دیباچه را اینک شرح می‌دهیم. حالت متغیرها ممکن است صحیح یا اعشاری باشد؛ جملهٔ "NORMALLY, MODE IS ..." زمینه‌ساز تعیین حالت تمام متغیرهای اعلام نشده است. هر چند نامها در SIMSCRIPT می‌تواند هر طولی داشته باشد، پنج یا شش کاراکتر اول (برحسب نوع نام و کامپیوتر مورد استفاده) باید منحصر به فرد باشد؛ جملهٔ «DEFINE کلمه TO MEAN تعریف» به همگردان دستور می‌دهد که در هر برخورد به «کلمه» در برنامه، «تعریف» را جانشین آن کند. در این مثال خط سوم دیباچه تضمین می‌کند که نام "DEPARTING.CUSTOMER" به‌عنوان "ATRB1" همگردانی شود و این با پیشامد "DEPARTURE" اشتباه نشود. پس از این، دو پیشامد

```
PREAMBLE
    NORMALLY, MODE IS INTEGER
    DEFINE DEPARTING.CUSTOMER TO MEAN ATRB1
    DEFINE ARRIVAL.TIME TO MEAN ATRB2

    EVENT NOTICES INCLUDE ARRIVAL
         EVERY 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 MIAT, MSVT, SIGMA, ARRIVAL.TIME, AND RESPONSE.TIME
         AS REAL VARIABLES
    DEFINE SERVER, NCUST, IS.RT.4 AND NUMBER.OF.DEPARTURES
         AS INTEGER VARIABLES
    DEFINE IDLE TO MEAN 0
    DEFINE BUSY TO MEAN 1

    ACCUMULATE RHO AS THE AVERAGE OF SERVER
    TALLY MAX.Q.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 ۳-۱۹ دیباچهٔ SIMSCRIPT برای مدل صف تک خدمت‌دهنده.

ARRIVAL و DEPARTURE تعریف می‌شود. در SIMSCRIPT هر اخطار در مورد پیشامد سابقه‌ای است مبنی بر اینکه پیشامدی از نوع خاص در زمانی (معمولاً آینده) رخ خواهد داد. اخطارهای پیشامد در فهرست پیشامدهای آتی درج می‌شود. درست به همان گونه که نهادها در یک مجموعه درج می‌شود. به‌علاوه، اخطارهای مربوط به پیشامد ممکن است ویژگیهایی داشته باشد. پیشامد "ARRIVAL" هیچ ویژگی ندارد. اما پیشامد "DEPARTURE" از ویژگی "DEPARTING.CUSTOMER" برخوردار است. برای هر رده از پیشامدها یک برنامهٔ پیشامد وجود دارد که پس از این مورد بحث قرار می‌گیرد. یک نوع نهاد موقت، یعنی یک "CUSTOMER" با یک ویژگی "ARRIVAL.TIME" تعریف شده است. هر نهاد "CUSTOMER" ممکن است متعلق به "QUEUE" باشد که مجموعه‌ای بر اساس ترتیب ورود و «متعلق» به سیستم است (و اساساً بدین معناست که تنها یک مجموعه به نام "QUEUE" وجود دارد). (در این مدل مجموعهٔ موصوف به "QUEUE" دربردارندهٔ متقاضیان در حال انتظار ولی نه متقاضی در حال خدمتگیری است. این تعریف در اختیار برنامه‌نویس بوده است.) "REPORT.GENERATOR" یک زیربرنامهٔ متعارف است که برنامه‌نویس آن را فرا می‌خواند (این زیربرنامه را برنامهٔ جلوبری زمان و زمانبندی پیشامدها فرا نمی‌خواند). سپس، برخی متغیرهای سراسری صریحاً با حالت اعشاری یا صحیح تعریف می‌شود. دو جملهٔ "DEFINE...TO MEAN..." بدین معناست که در تمام برخوردها به کلمهٔ IDLE (یا BUSY) پیش از همگردانی برنامه، کلمهٔ مزبور با عدد صفر (یا یک) جانشین می‌شود. این جانشینی نمادین خوانایی برنامه را بهبود می‌بخشد. جملات "ACCUMULATE" و "TALLY" تقاضای گردآوری آمار خاصی را مطرح می‌کند. "SERVER" یک متغیر حالت سیستم است که برنامه‌نویس برای معرفی وضعیت خدمت‌دهنده، به آن مقادیر یک (BUSY) و صفر (IDLE) را می‌دهد. با جملهٔ "ACCUMULATE" زبان SIMSCRIPT به‌طور خودکار جمع انباشتهٔ موزون از لحاظ زمانی (به نام B در مدل FORTRAN) متغیر "SERVER" را نگه می‌دارد. در پایان شبیه‌سازی حاصل تقسیم این جمع انباشته به TIME.V، به مقدار "RHO" خواهد انجامید. متوسط "RHO" که به‌طور خودکار از جملهٔ "ACCUMULATE" تولید می‌شود، مثالی از یک میانگین موزون زمانی است (که بیشتر در بخش ۵-۴ مورد بررسی قرار می‌گیرد). RHO با درصد زمانی که خدمت‌دهنده مشغول است مساوی خواهد بود. "TALLY" اول، MAX.Q.LENGTH را به‌عنوان ماکسیمم تعداد نهادهای مجموعهٔ "QUEUE" تولید خواهد کرد زیرا "N. *نام مجموعه*" توسط SIMSCRIPT به‌طور خودکار برای هر مجموعه نگهداری می‌شود. هرگاه یک پیشامد DEPARTURE روی دهد، برنامه‌نویس "RESPONSE.TIME" مربوط به "DEPARTING.CUSTOMER" را محاسبه می‌کند و SIMSCRIPT به‌طور خودکار این مقدار را به آمار تجمعی می‌افزاید که این جمع نیز در پایان شبیه‌سازی برای محاسبهٔ متوسط مدت پاسخ، "AVG.RT" مورد استفاده قرار می‌گیرد. هرگاه طول مدت پاسخ ۴ دقیقه یا بیشتر باشد، متغیر "IS.RT.4" مساوی یک قرار داده می‌شود؛ در غیر این‌صورت، مقدار آن


```
MAIN
    SCHEDULE AN ARRIVAL NOW

    LET MIAT = 4.5   ''MINUTES, THE MEAN INTERARRIVAL TIME
    LET MSVT = 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 ۳-۲۰ برنامهٔ اصلی SIMSCRIPT برای مدل صف تک خدمت‌دهنده.

صفر است؛ به این ترتیب، RROB.RT.GE.4 درصد موارد ترکی خواهد بود که مدت پاسخ آنها بزرگتر یا مساوی با ۴ دقیقه باشد.

برنامهٔ اصلی در شکل ۳-۲۰ نشان داده شده است. جملهٔ SCHEDULE به منظور قرار دادن اخطارهای پیشامد در فهرست پیشامدهای آتی به‌کار برده شده است. اولین پیشامد ARRIVAL، برای وقوع در زمان صفر SCHEDULE شده است. به پارامترهای ورودی MIAT، MSVT، SIGMA، و NCUST (که همان تعابیر مدل FORTRAN را دارد) مقدار داده می‌شود؛ هر چند که به طریقی دیگر امکانپذیر بود که این مقادیر از پرونده‌ای با استفاده از ورودی میدان آزاد و قالب آزاد SIMSCRIPT خوانده شود. متغیر حالت سیستم "SERVER" مساوی با "IDLE" (یعنی صفر) قرار داده می‌شود. هر چند که فوراً توسط برنامهٔ پیشامد "ARRIVAL" تبدیل به "BUSY" می‌شود. جملهٔ START SIMULATION اجرای الگوریتم جلوبری زمان و زمانبندی پیشامدها را شروع می‌کند. توجه کنید که اگر مایل بودیم اولین ورود در یک زمان تصادفی رخ دهد، جملهٔ SCHEDULE می‌توانست با جملهٔ زیر جانشین شود

SCHEDULE AN ARRIVAL IN EXPONENTIAL.F(MIAT,1) MINUTES

مولد مقدار تصادفی نمایی (و مولدهای بسیار دیگر) در SIMSCRIPT تعبیه شده است. (MIAT میانگین نمایی است؛ آرگومان دوم می‌تواند هر عدد صحیح از ۱ تا ۱۰ باشد تا مشخص کند کدام یک از ۱۰ رشته اعداد تصادفی مورد نظر است.)

برنامهٔ پیشامد ARRIVAL در شکل ۳-۲۱ نشان داده شده است. جملهٔ "CREATE..." یک نسخه یا یک مورد از نهاد موقتی نامبرده ایجاد می‌کند. ویژگی "ARRIVAL.TIME" این CUSTOMER تازه ایجاد شده با زمان کنونی شبیه‌سازی مساوی قرار داده می‌شود.

```
EVENT ARRIVAL
    CREATE A CUSTOMER
        LET ARRIVAL.TIME = TIME.V

    IF SERVER IS EQUAL TO IDLE,
        LET SERVER = BUSY
        SCHEDULE A DEPARTURE GIVEN CUSTOMER IN
            NORMAL.F(MSVT, SIGMA, 1) MINUTES

    ELSE
        FILE CUSTOMER IN QUEUE

    REGARDLESS
        SCHEDULE AN ARRIVAL IN EXPONENTIAL.F(MIAT, 1) MINUTES

    RETURN
    END
```

شکل ۳-۲۱ برنامهٔ پیشامد ورود به SIMSCRIPT برای مدل صف تک خدمت‌دهنده.

SIMSCRIPT یک جملهٔ IF ساختاربندی شده به شکل زیر دارد که خوانایی مدل را به‌طور قابل‌توجهی بهبود می‌بخشد.

IF شرط برقرار است، این جملات را انجام بده.
OTHERWISE (یا ELSE) این جملات را انجام بده.
ALWAYS (یا REGARDLESS).

اگر SERVER بیکار، IDLE، باشد (یعنی اگر متغیر SERVER مساوی صفر باشد)، SERVER مشغول، BUSY، و وقوع پیشامد ترک CUSTOMER تازه وارد برای پایان مدت خدمتدهی (که طبق فرض با توزیع نرمال) تولید شده زمانبندی می‌شود. این زمانبندی بدان معناست که یک اخطار پیشامد DEPARTURE در فهرست پیشامدها قرار داده می‌شود؛ این اخطار اطلاعات بیشتری (به‌نام نشانگر یا شاخص) را در بر دارد که به CUSTOMER مشخصی که سیستم را ترک خواهد کرد اشاره می‌کند. هر نهاد موقت CREATE شده نشانگری نظیر خود خواهد داشت تا آن را از سایر نهادهای موقت همنوع متمایز کند. مقادیر این نشانگرها باید ذخیره شود. در غیر این صورت از دست می‌رود. نشانگر مربوط به آن CUSTOMER که تازه مشغول خدمتگیری شده است در اخطار پیشامد DEPARTURE ذخیره می‌شود. نشانگرهای مربوط به سایر نهادهای CUSTOMER، یعنی آن CUSTOMERهایی که در حال انتظار برای خدمتگیری‌اند، با جملهٔ "FILE CUSTOMER IN QUEUE" در مجموعه‌ای به نام "QUEUE" ذخیره می‌شود. بدیهی است این عمل وقتی انجام شود که یک پیشامد ARRIVAL رخ دهد و SERVER در حالت BUSY باشد. تحت تمام این شرایط، هرگاه ورودی رخ دهد، با تولید یک مقدار برای مدت

بین دو ورود، ورود بعدی SCHEDULE می‌شود. توجه داشته باشید که مولد مقدار تصادفی نرمال، یعنی NORMAL.F به سه نشانوند نیاز دارد: میانگین (MSVT)، انحراف معیار (SIGMA) و رشتهٔ مورد نظر اعداد تصادفی (عددی صحیح از یک تا ۱۰).

برنامهٔ پیشامد DEPARTURE در شکل ۳-۲۲ نشان داده شده است. هرگاه این پیشامد رخ دهد، ویژگی DEPARTING.CUSTOMER دربردارندهٔ نشانگری خواهد بود که قبلاً (با جملهٔ SCHEDULE در برنامهٔ ARRIVAL یا جملهٔ SCHEDULE موجود در همین برنامه) در اخطار پیشامد ذخیره شده بود. این نشانگر مشخص می‌کند که کدام نهاد CUSTOMER مربوط به این پیشامد DEPARTURE است و برای بازیابی ARRIVAL.TIME مربوط به DEPARTING.CUSTOMER مورد استفاده قرار می‌گیرد به‌طوری که RESPONSE.TIME این متقاضی محاسبه‌پذیر می‌شود. پس از این، نهاد CUSTOMER که DEPARTING.CUSTOMER نام داشت، DESTROYED می‌شود، یعنی آن بخش از حافظهٔ کامپیوتر که قبلاً به ذخیره‌سازی این نهاد و ویژگیهایش اختصاص یافته بود، اینک در اختیار SIMSCRIPT است تا برای سایر مقاصد مورد استفاده قرار گیرد. (وقتی نهادهای موقت سیستم را ترک می‌کنند، همگی باید نابود شوند، در غیر این‌صورت

```
EVENT DEPARTURE GIVEN DEPARTING.CUSTOMER
    LET RESPONSE.TIME = TIME.V - ARRIVAL.TIME(DEPARTING.CUSTOMER)
    DESTROY A CUSTOMER CALLED DEPARTING.CUSTOMER

    IF RESPONSE.TIME*HOURS.V*MINUTES.V IS NOT LESS THAN 4,
        LET IS.RT.4 = 1
    ELSE
        LET IS.RT.4 = 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(MSVT, SIGMA, 1) MINUTES

    REGARDLESS

RETURN
END
```

شکل ۳-۲۲ برنامهٔ پیشامد ترک به SIMSCRIPT برای مدل صف تک خدمت‌دهنده.

www.matlabdl.com

حافظهٔ کامپیوتر ممکن است مملو از اطلاعاتی شود که به‌کار شبیه‌سازی نمی‌آید. اگر حافظه کاملاً پر شود، شبیه‌سازی را نمی‌توان ادامه داد.) سپس، بسته به اینکه RESPONSE.TIME کمتر از ۴ دقیقه باشد یا نه متغیر نشانگر IS.RT.۴ مساوی با صفر یا یک قرار داده می‌شود. واحد (ضمنی) زمان در SIMSCRIPT روز است که به این ترتیب، RESPONSE.TIME برحسب روز (در جملهٔ IF) با ضرب کردن در ۲۴ (HOURS.V) و در ۶۰ (MINUTES.V) به دقیقه تبدیل می‌شود. NUMBER.OF.DEPARTURES یک آمار خروجی است که برنامه‌نویس آن را گردآوری می‌کند (ADD 1 TO X معادل LET X = X + ۱ است.) در جملهٔ IF بعدی NUMBER.OF.DEPARTURES با NCUST = ۱۰۰۰ مورد مقایسه قرار می‌گیرد تا معلوم شود آیا باید REPORT.GENERATOR را فراخواند یا نه. (توجه داشته باشید که ELSE در ترکیب IF... ALWAYS اختیاری است.) آخرین جملهٔ IF منطق پیشامد DEPARTURE را اجرا می‌کند. اگر مجموعهٔ "QUEUE" خالی، EMPTY، باشد SERVER بیکار، IDLE و کنترل به برنامهٔ جلوبری زمان و زمانبندی پیشامدها بازگردانده می‌شود. (EMPTY یک واژهٔ کلیدی است.) اگر "QUEUE" خالی نباشد دست‌کم باید یک نهاد CUSTOMER در مجموعه وجود داشته باشد. پس، اولین آنها REMOVE می‌شود. به یاد دارید که در دیباچه، مجموعهٔ "QUEUE" نوعی مجموعهٔ FIFO (به‌ترتیب ورود) تعریف شد. پس از این، یک پیشامد DEPARTURE برای نهاد CUSTOMER حذف (REMOVE) شده SCHEDULE می‌شود و کنترل به برنامهٔ جلوبری زمان انتقال می‌یابد.

شکل ۳-۲۳ فهرست برنامهٔ REPORT.GENERATOR را ارائه می‌دهد. (این، یکی از

```
ROUTINE REPORT.GENERATOR
PRINT 1 LINE THUS
        SINGLE SERVER QUEUE SIMULATION - GROCERY STORE CHECKOUT COUNTER
SKIP 2 OUTPUT LINES
PRINT 4 LINES WITH MIAT, MSVT, SIGMA, NCUST THUS
                 MEAN INTERARRIVAL TIME                     **.**
                 MEAN SERVICE TIME                          **.**
                 STANDARD DEVIATION OF SERVICE TIMES        **.**
                 NUMBER OF CUSTOMERS TO BE SERVED            ****
SKIP 3 OUTPUT LINES
PRINT 7 LINES WITH RHO, MAX.Q.LENGTH, AVG.RT*HOURS.V*MINUTES.V
    PROB.RT.GE.4, TIME.V*HOURS.V*MINUTES.V, AND
    NUMBER.OF.DEPARTURES  THUS
                 SERVER UTILIZATION                    **.**
                 MAXIMUM LINE LENGTH                    ****
                 AVERAGE RESPONSE TIME                 **.** MINUTES
                 PROPORTION WHO SPEND FOUR
                   MINUTES OR MORE IN SYSTEM            *.**
                 SIMULATION RUNLENGTH                ****.**
                 NUMBER OF DEPARTURES                   ****
STOP
END
```

شکل ۳-۲۳ برنامهٔ تهیهٔ گزارش به SIMSCRIPT برای مدل صف تک خدمت‌دهنده.

برنامه‌های پیشامدها نیست.) در اولین موردی که ضابطهٔ متوقف کردن راضی شود، این برنامه را برنامهٔ پیشامد DEPARTURE فرامی‌خواند. شکل کلی جملهٔ PRINT به‌صورت

PRINT n LINES WITH نام متغیرها THUS

است که در پی آن باید n خط با ظاهری دقیقاً به همان گونه بیاید که برنامه‌نویس در خروجی می‌خواهد. به قالب‌بندی «تصویری» توجه کنید: مقدار MIAT دقیقاً در جایی که ** . ** واقع است قرار داده می‌شود، و ... (این‌گونه قالب‌بندی تصویری، گزارش‌نویسی را بسیار ساده‌تر از وقتی می‌کند که از جملات مربوط به قالب شبیه FORTRAN استفاده شود.) جملهٔ STOP در آخر برنامه موجب متوقف شدن شبیه‌سازی می‌شود.

خروجی این شبیه‌سازی در شکل ۳-۲۴ نمایش داده شده است. به تفاوتهای موجود بین خروجی مدل SIMSCRIPT در شکل ۳-۲۴ و خروجی مدل FORTRAN در شکل ۳-۱۸ توجه کنید. این تفاوتها مدیون این واقعیت است که هر دو زبان از رشته‌های گوناگون اعداد تصادفی استفاده کردند. باز دیده می‌شود که هر اجرای شبیه‌سازی، براوردی از عملکرد سیستم فراهم می‌آورد و ممکن است این براورد دربر دارندهٔ خطای تصادفی ناشی از نوسانات ذاتی سیستم باشد. اگر مدلهای FORTRAN و SIMSCRIPT هر دو را می‌توانستیم برای مدتی «نامتناهی» اجرا کنیم (به‌جای به‌ترتیب ۴۴۶۰ دقیقه و ۴۷۹۴ دقیقه)، هر دو مدل براوردهای یکسانی تولید می‌کردند و این براوردها فاقد خطای تصادفی بودند. در عمل، دو براورد از یک پارامتر که با استفاده از اعداد تصادفی مختلف به‌دست آمده باشد، با افزایش طول اجرای شبیه‌سازی سرانجام از لحاظ مقدار به‌هم نزدیک و نزدیکتر می‌شود.

اگر شبیه‌ساز تمایل به شبیه‌سازی برای مدتی ثابت، مثلاً ۸۰ ساعت را می‌داشت یک پیشامد

```
SINGLE SERVER QUEUE SIMULATION - GROCERY STORE CHECKOUT COUNTER

          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                   .67
          MAXIMUM LINE LENGTH                  8
          AVERAGE RESPONSE TIME             6.51 MINUTES
          PROPORTION WHO SPEND FOUR
            MINUTES OR MORE IN SYSTEM          .63
          SIMULATION RUNLENGTH            4794.13
          NUMBER OF DEPARTURES               1000
```

شکل ۳-۲۴ خروجی مدل SIMSCRIPT برای صف تک خدمت‌دهنده.

سوم به نام STOP.SIMULATION در دیباچه می‌توانست تعریف شود. این پیشامد در برنامهٔ اصلی با جملهٔ SCHEDULE A STOP.SIMULATION IN 80 HOURS زمانبندی می‌شود. برنامهٔ پیشامد STOP.SIMULATION صرفاً برنامهٔ REPORT.GENERATOR را فرا می‌خواند. به‌علاوه، جملهٔ IF که NUM-BER. OF. DEPARTURES را در برنامهٔ پیشامد DEPARTURE بررسی می‌کند باید حذف شود.

به‌طور خلاصه، باید آشکار شده باشد که SIMSCRIPT زبانی توانا برای شبیه‌سازی گسسته پیشامد است. اگر SIMSCRIPT به‌خوبی فهمیده شود، انتظار می‌رود ایجاد و غلطگیری مدل آن به‌طور قابل ملاحظه‌ای کمتر از مدل FORTRAN وقت بگیرد. SIMSCRIPT از امکانات متعدد غلطگیری برخوردار است که می‌تواند مدت زمان ایجاد مدل را هرچه کوتاهتر کند. به‌علاوه، استفاده از توان پردازش-تقابل SIMSCRIPT، اگر میسر باشد، ممکن است به‌طور قابل ملاحظه‌ای تعداد جملات مدل را کاهش دهد و از این‌رو بر سرعت ایجاد مدل بیفزاید. این توان به اختصار از سوی لا وکلتون [۱۹۸۲] و راسل [۱۹۷۶] و به تفصیل از سوی راسل [۱۹۸۳] تشریح شده است. ■

۴-۲-۳ شبیه‌سازی با GPSS

GPSS یک زبان شدیداً ساختاربندی شده و ویژهٔ شبیه‌سازی است که رهیافت پردازش-تقابل را به‌کار می‌برد، نسبت به شبیه‌سازی مسائل صف گرایش دارد و دیاگرام بلوکی برای شرح سیستم فراهم می‌کند. نهادهایی موقت موسوم به نهادهای گذرنده ایجاد می‌شود و می‌توان آنها را چنین تصویر کرد که در دیاگرام بلوکی جاری است. به این ترتیب، از GPSS می‌توان برای هر وضعیتی استفاده کرد که در آن نهادها (مثلاً، متقاضیان) را بتوان به‌صورت گذرکننده از میان سیستم مجسم کرد (مثلاً، شبکه‌ای از صفها). GPSS به مانند FORTRAN یا SIMSCRIPT یک زبان دستوری نیست. بلکه روشی ساختاربندی شده برای تشریح تفصیلی انواع خاصی از سیستمهاست. پردازشگر GPSS سپس این شرح (یعنی، دیاگرام بلوکی) را می‌گیرد و به‌طور خودکار یک شبیه‌سازی انجام می‌دهد. مکانیزم جلوبری زمان و زمانبندی پیشامدها کاملاً از دید پنهان است.

GPSS اصلاً به‌وسیلهٔ جفری گوردون از شرکت IBM حوالی سال ۱۹۶۰ به‌وجود آمد و طی چند نسخه که تازه‌ترین آنها GPSS/۳۶۰ و GPSS V است تکامل یافت. این زبان نه تنها برای کامپیوترهای IBM ردهٔ ۳۶۰ و ۳۷۰، بلکه شکلی از آن برای اکثر سیستمهای کامپیوتر بزرگ دسترسپذیر است. مقدمه‌ای بر GPSS را می‌توان در گوردون [۱۹۷۵، ۱۹۷۸] یا در شرایبر [۱۹۷۴] یافت. در سالهای اخیر، پروفسور شرایبر در هر تابستان سمیناری یک هفته‌ای در مورد GPSS در دانشگاه میشیگان برگزار کرده است. به‌کارگیری تازه‌ای از GPSS به‌نام GPSS/H اخیراً توسط هنریکسن [۱۹۷۹] از شرکت نرم‌افزار وولورین، شهر فالز چرچ در ویرجینیا توسعه داده

1. General Purpose Simulation System

شده است. به‌موجب گزارشهای منتشر شده، مدلهای نوشته شده به GPSS/۳۶۰ (یا GPSS V که زیرمجموعه‌ای از آن GPSS/۳۶۰ است) بدون تغییر یا با انجام تغییرات کوچکی از ۴ تا ۳۰ مرتبه سریعتر (مدت CPU) با GPSS/H اجرا می‌شود. به‌علاوه، GPSS/H از امکانات بیشتری برخوردار است که برخی از محدودیتهای GPSS/۳۶۰ را برطرف می‌کند. در حال حاضر، GPSS/H تنها برای ماشینهای IBM دسترسپذیر است ولی سرانجام برای کامپیوترهای دیگر نیز ارائه خواهد شد.

از دههٔ ۱۹۶۰ تاکنون شاید GPSS وسیعترین کاربرد را در میان زبانهای شبیه‌سازی گسسته پیشامد داشته است. دو دلیل این محبوبیت عبارت از آسانی فراگیری زبان (به‌ویژه برای اشخاصی غیر از برنامه‌نویسان) و مدت نسبتاً کوتاه مورد نیاز برای ساختن یک مدل پیچیده است. از سوی دیگر، GPSS دارای نقصهایی است که برخی از آنها ممکن است برنامه‌نویس بی‌دقت را به ایجاد مدلهایی ظاهراً معتبر ولی واقعاً بی‌اعتبار رهنمون شود. سایر نقصها مدل کردن سیستمهای خاصی را بی‌اندازه دشوار و پر زحمت می‌سازد:

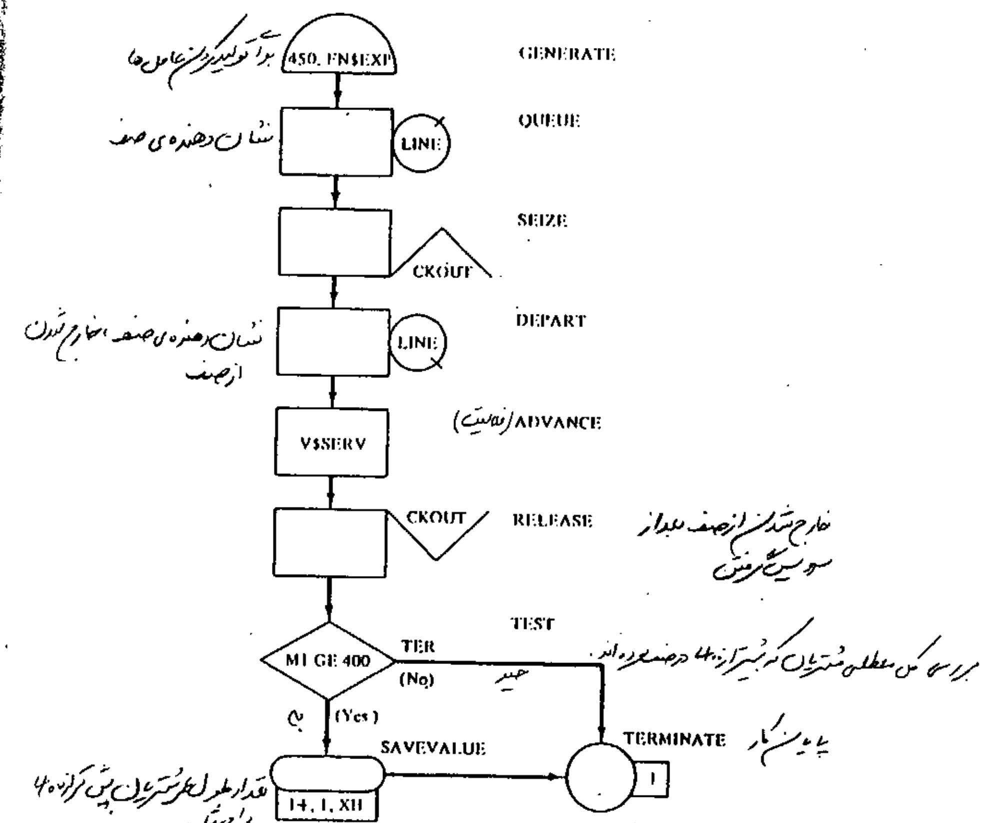
۱. ساعت شبیه‌سازی، موسوم به AC1، تنها می‌تواند مقادیر صحیح ۱، ۲، ۳، ... بگیرد. نتیجهٔ نهایی هر محاسبه همواره پیش از آنکه به‌عنوان زمان پیشامد به‌کار گرفته شود، بریده (گرد به پایین) می‌شود. بنابراین، باید یک واحد زمان کوچک مناسب چنان برگزید که مدتهای فعالیتها و ساعت به‌قدر کافی دقیق باشد.

۲. اجرای محاسبات عددی پیچیده و عملیات منطقی با GPSS می‌تواند اگر نه غیرممکن، بی‌اندازه دشوار باشد. مشخصاً، GPSS فاقد توانایی محاسبهٔ مستقیم توابع لگاریتمی، سینوسی، نمایی، قدر مطلق، ماکسیمم، و سایر توابع متداول ریاضی است. از لحاظ نظری، GPSS ممکن است این توابع را با یک تابع پاره‌خطی یا با بلوک GPSS HELP که اجازهٔ فراخواندن یک برنامهٔ FORTRAN را می‌دهد، تقریب بزند. از لحاظ عملی، این تقریبها توسعه داده نشده است یا خیلی دقیق نیست. به‌علاوه، توانایی فراخواندن برنامهٔ FORTRAN نیز بسیار پرزحمت است و سودمندی محدودی دارد.

۳. GPSS فاقد هرگونه مولد مقادیر تصادفی درونی است. بدون دسترسی به توابع ریاضی مذکور در (۲)، GPSS ناگزیر از درونیابی در چارچوب یک تابع به منظور تولید مقادیر تصادفی است. این تقریب پاره‌خطی در زیربخش ۸-۱-۶ شرح داده شده است. تقریبهای استانداردی برای توزیع نمایی با میانگین ۱ و توزیع نرمال استاندارد ارائه شده است [گوردون، ۱۹۷۵].

۴. تمام هشت رشته اعداد تصادفی، دنبالهٔ یکسانی از اعداد تصادفی را تولید می‌کنند مگر اینکه برنامه‌نویس به صراحت «هسته‌های» متمایزی برای هر رشته تعریف کند. «هسته» به مثابه نقطهٔ شروع در یک رشته از اعداد تصادفی همانند اعداد جدول پ-۱ است. جملهٔ RMULT به منظور مشخص کردن هسته‌ها در مورد هر رشته به‌کار می‌رود. (اگر در GPSS V از RMULT استفاده نشود، تمام هشت هسته به‌طور ضمنی مقدار مشترک ۳۷ را می‌گیرد.) باید توجه داشت که GPSS/H اکثر این نقصها را برطرف کرده است.

دو مفهوم اصلی در GPSS، عبارت از نهادهای گذرنده و بلوکهاست. هر بلوک می‌تواند به‌صورت نمادی تصویری یا جمله‌ای به زبان GPSS معرفی شود. در GPSS بیش از ۴۰ بلوک استاندارد وجود دارد. هر بلوک معرف فعالیت یا پیشامد مشخصی است که در هر سیستم عادی ممکن است رخ دهد. این بلوکها در یک دیاگرام بلوکی که معرف پردازش یک «متقاضی» است مرتب می‌شود. شکل ۳-۴ مثالی از یک پردازش «متقاضی» را برای صف تک خدمت‌دهنده نشان داد. توصیف GPSS چنین پردازشی در دیاگرام بلوکی شکل ۳-۲۵ نشان داده و در مثال ۳-۸ شرح داده شده است. نهادهای گذرنده که معرف نهادهای پویا و فعال است را می‌توان چنین تصویر کرد که در سراسر دیاگرام بلوکی جاری است. هر مسیری را که یک نهاد گذرنده بتواند در سیستم در پیش گیرد باید در دیاگرام بلوکی، که می‌تواند شاخه‌هایی داشته باشد نشان داده شود. منابع محدود هر سیستم با نهادها، امکانات و انبارهای از قبل تعریف شدهٔ GPSS معرفی می‌شود. هر



شکل ۳-۲۵ دیاگرام بلوکی GPSS برای شبیه‌سازی صف تک خدمت‌دهنده.

مورد از امکانات، اساساً یک خدمت‌دهنده است. هر انبار متشکل از گروهی از خدمت‌دهنده‌های موازی است. آمار به‌طور خودکار در مورد ضریب بهره‌برداری از امکانات و انبارها گردآوری می‌شود. به‌علاوه، نهادهای صفی و جدولی به منظور گردآوری آمار در مورد صفهای انتظار یا مدتهای انتقال دسترسپذیر است.

■ **مثال ۳-۸ (شبیه‌سازی صف تک خدمت‌دهنده به GPSS)**

شکل ۳-۲۵ دیاگرام بلوکی و شکل ۳-۲۶ برنامهٔ GPSS را برای مدل باجهٔ صندوق فروشگاه مواد غذایی مشروح در مثال ۳-۵ نمایش می‌دهد. توجه داشته باشید که برنامه (شکل ۳-۲۶) ترجمهٔ دیاگرام بلوکی همراه با تعاریف اضافی و کارتهای کنترلی است.

```
BLOCK
NUMBER  *LOC   OPERATION  A,B,C,D,E,F,G,H,I,J        COMMENTS                              CARD
                                                                                            NUMBER
        *
               SIMULATE
        * SIMULATION OF A SINGLE SERVER QUEUE
        *
1        EXP   FUNCTION   RN1,C24          EXPONENTIAL GENERATOR
1       0,0/.1,.104/.2,.222/.3,.355/.4,.509/.5,.69
1       .6,.915/.7,1.2/.75,1.38/.8,1.6/.84,1.83/.88,2.12
1       .9,2.3/.92,2.52/.94,2.81/.95,2.99/.96,3.2/.97,3.5
1       .98,3.9/.99,4.6/.995,5.3/.998,6.2/.999,7/.9997,8
        *
2        NORM  FUNCTION   RN1,C25          NORMAL GENERATOR
2       0,-5/.00003,-4/.00135,-3/.00621,-2.5/.02275,-2/.06681,-1.5
2       .11507,-1.2/.15866,-1/.21186,-.8/.27425,-.6/.34458,-.4/.42074,-.2
2       .5,0/.57926,.2/.65542,.4/.72575,.6/.78814,.8/.84134,1/.88493,1.2
2       .93319,1.5/.97725,2/.99379,2.5/.99865,3/.99997,4/1,5
        *
1        SERV  FVARIABLE  320+60*FN$NORM          GENERATE SERVICE TIME
        *
    1          GENERATE   450,FN$EXP       CUSTOMERS ARRIVE AT RANDOM, ON THE
        *                                   AVERAGE EVERY 4.5 MINUTES
        *                                  (TIME UNIT = 1/100 MINUTE)
    2          QUEUE      LINE             CUSTOMER JOINS WAITING LINE
    3          SEIZE      CKOUT            BEGIN CHECKOUT AT CASH REGISTER
    4          DEPART     LINE             CUSTOMER STARTING SERVICE LEAVES QUEUE
    5          ADVANCE    V$SERV           CUSTOMER'S SERVICE TIME
    6          RELEASE    CKOUT            CUSTOMER LEAVES CHECKOUT AREA
    7          TEST GE    M1,400,TER       IS RESPONSE TIME GE 4 MINUTES?
    8          SAVEVALUE  1+,1,XH          IF SO, ADD 1 TO COUNTER(XH1)
    9   TER    TERMINATE  1
        *
               START      1000             SIMULATE FOR 1000 DEPARTURES
```

شکل ۳-۲۶ برنامهٔ GPSS برای شبیه‌سازی صف تک خدمت‌دهنده.

www.matlabdl.com

بلوک GENERATE در شکل ۳-۲۵ معرف پیشامد ورود است که از طریق عملوندهای خود، یعنی "FN$EXP، ۴۵۰"، مدتهای بین دو ورود را نیز تعریف می‌کند. واحد زمان $\frac{1}{100}$ دقیقه است که به این ترتیب، ۴۵۰ واحد زمان = ۴/۵ دقیقه میانگین مدت بین دو ورود است. عملوند دوم، یعنی FN$EXP به FUNCTION EXP اشاره دارد که به منظور تولید مقادیر تصادفی با توزیع تقریباً نمایی با میانگین ۱ به‌کار برده می‌شود. (ضرب کردن ۴۵۰ در FN$EXP میانگین را به ۴۵۰ تغییر می‌دهد.) بلوک GENERATE نهادهای گذرنده با مدتهای بین دو ورود به شرح عملوندهای خود تولید می‌کند و این نهادها به محض ورود شروع به گذشتن از سراسر دیاگرام بلوکی می‌کنند.

تک خدمت‌دهنده با یک مورد امکانات که CKOUT نام داده شده است معرفی می‌شود. هر مورد امکانات با یک زوج بلوک، یعنی SEIZE و RELEASE، به شرح زیر مدلسازی می‌شود:

خدمتدهی را هر چه زودتر شروع کن	CKOUT	SEIZE
فعالیت خدمتدهی	(بلوکهایی که مدت خدمتدهی را مدلسازی می‌کند)	.
پایان خدمتدهی	CKOUT	RELEASE

در GPSS نهادهای گذرندهٔ بسیاری می‌تواند به‌طور همزمان در دیاگرام بلوکی حاضر باشد؛ هر نهاد گذرندهٔ فعال، همواره در یک بلوک مشخص قرار دارد و هر بلوک وقتی به‌کار می‌آید که یک نهاد گذرنده به آن وارد شود؛ برخی از بلوکها (مانند QUEUE، DEPART، RELEASE، و ADVANCE) همواره پذیرای نهادهای گذرنده است، یعنی، اجازهٔ ورود به خود و در صورت امکان گذر به بلوک بعدی را به آنها می‌دهد. برخی دیگر از بلوکها گاهی مانع از ورود نهادهای گذرنده می‌شود. بلوک SEIZE هرگاه که یک مورد از امکانات مشغول یا در اشغال نهاد گذرندهٔ دیگری باشد از پذیرش هر نهاد گذرنده‌ای امتناع می‌کند. در این‌گونه موارد، نهادهایی که در تلاش برای SEIZE هر مورد از امکانات است، در بلوک بلافاصله قبل از بلوک SEIZE باقی می‌ماند. نهادهای گذرنده بر اساس ضابطهٔ به‌ترتیب ورود پذیرفته خواهد شد. (نظام صف را می‌توان با استفاده از انواع اولویتها، یا با به اصطلاح زنجیرهای کاربر، مجموعه‌هایی از نهادهای گذرنده که برنامه‌نویس می‌تواند تقریباً به هر شکلی آنها را مرتب کند، تغییر داد.)

فعالیتها با بلوکهای ADVANCE نمایش داده می‌شود. بلوک ADVANCE در شکل ۳-۲۵ معرف مدتهای خدمتدهی با توزیع نرمال است. هر مدت خدمتدهی با فرمول موجود در جملهٔ FVARIABLE محاسبه می‌شود که این جمله به FUNCTION NORM رجوع می‌دهد. این FUNCTION مقادیر تصادفی (تقریباً) نرمال با میانگین صفر و انحراف معیار ۱ تولید می‌کند. جملهٔ FVARIABLE این مقدار تصادفی نرمال استاندارد را به مقداری تصادفی و نرمال با میانگین ۳۲۰ و انحراف معیار ۶۰ واحد زمانی تبدیل می‌کند.

آمار مربوط به صف انتظار با زوج QUEUE ... DEPART گردآوری می‌شود که به همراه

یکدیگر یک نهاد صف GPSS را تعریف می‌کند. توجه داشته باشید که هر نهاد صف GPSS موجب تشکیل صف انتظار نمی‌شود. در واقع بلوک SEIZE موجب تشکیل صف انتظار می‌شود و بلوکهای QUEUE ... DEPART صرفاً به سنجش آمار گوناگون در مورد این صف انتظار می‌پردازد. در حالت کلی‌تر می‌توان از یک نهاد صف GPSS به منظور گردآوری آمار مربوط به مدت انتقال و ازدحام در مورد هر سیستم فرعی از سیستم در دست بررسی استفاده کرد. به محض ورود هر نهاد گذرنده به بلوک QUEUE در شکل ۳-۲۵ گفته می‌شود که نهاد مزبور عضوی از صف موسوم به "LINE" است. به محض ورود به بلوک DEPART، یک نهاد گذرنده دیگر عضوی از صف نیست. متوسط مدت ماندن در صف به اضافهٔ میانگین زمانی تعداد نهاد گذرنده در صف از جملهٔ آماری است که به‌طور خودکار گردآوری و در پایان شبیه‌سازی چاپ می‌شود.

اینک شکل ۳-۲۵ را می‌توان به اختصار به شرح زیر تعریف کرد: گاه‌بگاه یک نهاد گذرنده به بلوک GENERATE وارد می‌شود و بلادرنگ به صف "LINE" می‌پیوندد. در اسرع وقت امکانات "CKOUT" را SEIZE و فوراً صف را DEPART می‌کند. مدتی از زمان شبیه‌سازی را در بلوک ADVANCE می‌ماند، که پس از آن، امکانات RELEASE می‌شود و در نتیجه برای نهاد گذرندهٔ بعدی که در تلاش SEIZE کردن آن است مهیا می‌شود. پس از RELEASE کردن امکانات، نهاد گذرنده وارد بلوک TERMINATE می‌شود که این بلوک آن را نابود می‌کند.

در شکل ۳-۲۶، جملهٔ SIMULATE یک کارت کنترلی است که به GPSS فرمان شبیه‌سازی را می‌دهد. (اگر این کارت ارائه نشود، GPSS صرفاً به بررسی خطاهای دستوری می‌پردازد.) در پی این کارت تعاریف دو تابع GPSS می‌آید که به تولید مقادیر تصادفی تقریبی نمایی و نرمال استاندارد مربوط است. جملهٔ FVARIABLE مدت خدمتدهی با توزیع نرمال مورد نظر را از مقدار تصادفی نرمال استاندارد محاسبه می‌کند. پس از این، جملات بلوکی مربوط به دیاگرام بلوکی می‌آید. جملهٔ آخر، یعنی START ۱۰۰۰، شبیه‌سازی را شروع و متوقف می‌کند. هر بار که نهاد گذرنده‌ای به بلوک TERMINATE وارد شود، از شمارشگری که مقدار اولیهٔ آن ۱۰۰۰ بوده است، مقدار "A" کم می‌شود که "A" عملوند بلوک TERMINATE است. (در شکل ۳-۲۶، A مساوی با ۱ است.) هرگاه شمارشگر به صفر برسد، شبیه‌سازی متوقف و گزارش خروجی به‌طور خودبه‌خود تهیه می‌شود (شکل ۳-۲۷).

توجه کنید که همهٔ آمار مورد نظر (به طریقی که در مدل FORTRAN نیز گردآوری شد) به استثنای درصدی از متقاضیان که مدت پاسخشان ۴ دقیقه یا بیشتر است، درگزارش خروجی استاندارد وجود دارد. GPSS به‌طور خودکار به نگهداری بسیاری از ویژگیهای سیستم و نهاد گذرنده می‌پردازد، که یکی از آنها، یعنی M1 معرف مدت انتقال نهاد گذرنده در مدل از لحظهٔ ایجاد آن است. به منظور شمردن تعداد نهادهای گذرنده‌ای که مدت پاسخشان در سیستم، M1، ۴ دقیقه یا بیشتر است، بلوکهای زیر به مدل افزوده شده‌اند.

.
.
.

TEST GE	M1,400,TER	بررسی کنید آیا M1 بزرگتر یا مساوی با ۴۰۰ است
SAVEVALUE	1+,1,XH	اگر چنین است، ۱ واحد به XH۱ اضافه کنید

.
.
.

شمارهٔ نهایی در «ذخیره‌گاه» XH1 ذخیره (و سپس چاپ) می‌شود.

گزارش استاندارد خروجی با اضافات فوق به مدل در شکل ۳-۲۷ نشان داده شده است. توجه کنید که:

درصد مدت اشتغال خدمت‌دهنده	$= ۰{,}۶۸۸$
ماکسیمم طول صف	$= ۱۱$
متوسط مدت پاسخ	$۳۱۸{,}۱۹۶ + ۴۲۲{,}۶۷۱ = ۷۴۰{,}۸۶۷$
درصد متقاضیانی که ۴ دقیقه یا بیشتر در باجهٔ صندوق می‌مانند	$\frac{۶۴۸}{۱۰۰۰} = ۰{,}۶۴۸$

به‌طور خلاصه، یک ویژگی قابل توجه GPSS در مقایسه با مدلهای FORTRAN و SIMSCRIPT، تعداد کم جملات مورد نیاز برای مدلسازی صف تک خدمت‌دهنده است. به‌طور کلی، هر زبان مبتنی بر نگرش پردازش-تقابل، خواه GPSS، یا بخش پردازش SIMSCRIPT یا SLAM، به منظور مدلسازی پدیده‌های صف متداول به جملات بسیار کمتری نیاز دارد. از سوی دیگر، GPSS در مقایسه با SIMSCRIPT یا SLAM از انعطاف و قدرت کمتری برخوردار است. معرفی برخی از پدیده‌های پیچیده با تعداد محدود بلوکهای موجود در GPSS ممکن است امری دشوار و پر زحمت باشد. علیرغم این نکته، GPSS به‌طور وسیع و موفقی در پروژه‌های شبیه‌سازی فراوانی مورد استفاده قرار گرفته است. ■

۳-۲-۵ شبیه‌سازی با SLAM

SLAM یک زبان سطح بالا و مبتنی بر زبان FORTRAN در شبیه‌سازی است که گرایش زمانبندی پیشامدها یا پردازش-تقابل، یا ترکیبی از هر دو را میسر می‌سازد. بخش زمانبندی پیشامدها در SLAM کاملاً شبیه GASP است که به اختصار در زیربخش ۳-۲-۳ تشریح شد. بخش پردازش-تقابل SLAM از بسیاری جنبه‌ها شبیه GPSS است. اینک بخش پردازش-تقابل SLAM را به اختصار توضیح می‌دهیم. مرجع و کتاب درسی مناسبی برای SLAM توسط ایجادکنندگان آن، یعنی پریتسکر و پگدن [۱۹۷۹] به رشتهٔ تحریر در آمده است. SLAM را شرکت پریتسکر

```
FINAL TERMINATION PRINTOUT

RELATIVE CLOCK      462628   ABSOLUTE CLOCK      462628 TERMINATIONS TO GO          0
BLOCK COUNTS
BLOCK   CURRENT   TOTAL  BLOCK     CURRENT  TOTAL  BLOCK     CURRENT  TOTAL  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

                                   *************************************
                                   *                                   *
                                   *            FACILITIES             *
                                   *                                   *
                                   *************************************

                                   - AVERAGE  UTILIZATION  DURING -

FACILITY        NUMBER     AVERAGE      TOTAL      AVAIL.     UNAVAIL.     CURRENT      PERCENT        TRANSACTION NUMBER
                ENTRIES    TIME/TRAN    TIME       TIME       TIME         STATUS    AVAILABILITY      SEIZING  PREEMPTING
 CKOUT            1000      318.196     0.688                                 A        100.000

                                   *************************************
                                   *                                   *
                                   *              QUEUES               *
                                   *                                   *
                                   *************************************

QUEUE     MAXIMUM    AVERAGE    TOTAL     ZERO      PERCENT     AVERAGE     $AVERAGE     TABLE     CURRENT
          CONTENTS   CONTENTS   ENTRIES   ENTRIES   ZEROS       TIME/TRANS  TIME/TRANS   NUMBER    CONTENTS
LINE          11      0.915      1001      326       32.6        422.671     626.806                   1
 $AVERAGE TIME/TRANS = AVERAGE TIME/TRANS EXCLUDING ZERO ENTRIES

                                   *************************************
                                   *                                   *
                                   *        HALFWORD SAVEVALUES        *
                                   *                                   *
                                   *************************************

NUMBER .. CONTENTS    NUMBER .. CONTENTS    NUMBER .. CONTENTS    NUMBER .. CONTENTS    NUMBER .. CONTENTS    NUMBER .. CONTENTS:
    1         648
```

شکل ۳-۲۷ گزارش خروجی استاندارد GPSS برای شبیه‌سازی صف تک خدمت‌دهنده.

و شرکاء، شهر وست لفه‌یت، ایالت ایندیانا که در زمینهٔ کاربرد آن دوره‌های کوتاه مدتی نیز ارائه می‌کند، به بازار عرضه کرده است.

به منظور استفاده از رهیافت پردازش_تقابل با SLAM، شبیه‌ساز باید شبکه‌ای متشکل از گرهها و شاخه‌ها ایجاد کند که به‌صورت تصویری پردازشها را در سیستم معرفی می‌کند. عناصر جاری در سیستم، نهاد نامیده می‌شود. (توجه داشته باشید که تعریف SLAM از یک نهاد محدودتر از تعریفی است که در این کتاب به‌کار گرفته‌ایم. هر نهاد SLAM همانند یک نهاد گذرندهٔ GPSS است. این‌گونه نهادها پویاست و سراسر هر پردازش را طی می‌کنند.) به یاد دارید که هر پردازش، توالی پیشامدها و فعالیتهایی است که هر نهاد به هنگام پیمودن سیستم با آنها مواجه می‌شود. هر مدل شبکه‌ای کامل SLAM از یک سیستم معرف تمام مسیرهایی است که هر نهاد به هنگام پیمودن سیستم ممکن است در پیش گیرد. به منظور اجرای مدل شبیه‌ساز مدل شبکه‌ای را مستقیماً به جملات کامپیوتری ترجمه می‌کند که به پردازندهٔ SLAM وارد می‌شود.

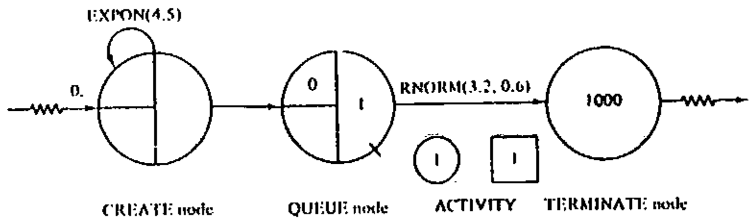
SLAM به‌طور خودکار الگوریتم زمانبندی پیشامدها و جلوبری زمان، عملیات مجموعه‌ها (یا پرونده‌ها) از قبیل افزودن یا کاستن نهادها، گردآوری آمارهای بسیار و تولید نمونه‌های تصادفی را اجرا می‌کند. مجموعه در SLAM پرونده نامیده می‌شود. با امکان ادارهٔ خودکار پرونده‌ها، SLAM به راحتی از عهدهٔ ادارهٔ صفها بر اساس ضابطهٔ به‌ترتیب ورود یا عکس ترتیب ورود برمی‌آید. به‌علاوه، می‌توان نهادها را برحسب یک ویژگی از قبیل اولویت رتبه‌بندی کرد (و خدمت داد). SLAM برخلاف GPSS مولدهای درونی مقادیر تصادفی برای انواع گسترده‌ای از توزیعهای آماری دارد.

هر شبکهٔ SLAM از شاخه‌ها و گرهها تشکیل می‌شود. هر شاخه معرف گذر زمان است، یعنی یک فعالیت را نشان می‌دهد. به‌علاوه، هر شاخه ممکن است معرف تعداد محدودی خدمت دهنده باشد. هر شاخه به‌صورت یک جملهٔ ACTIVITY رمزگذاری می‌شود. از گرهها برای معرفی پیشامد ورود (گره CREATE)، تأخیرها یا انتظارهای مشروط (گره QUEUE)، پیشامد ترک (گره TERMINATE)، یا سایر اعمال متداول سیستم استفاده می‌شود.

■ **مثال ۳-۹ (شبیه‌سازی صف تک خدمت‌دهنده با SLAM)**

مدل شبکه‌ای SLAM برای باجهٔ صندوق فروشگاه مواد غذایی در شکل ۳-۲۸ نشان داده شده است. توجه کنید که هر گره یک نماد مربوط به آن به‌اضافهٔ عملوندهاست. ترجمهٔ این شبکه به جملات SLAM در شکل ۳-۲۹ نشان داده شده است.

شکل ۳-۲۸ نشان می‌دهد که یک صف تک خدمت‌دهنده را می‌توان به سه گره و یک شاخهٔ

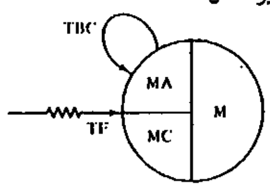


شکل ۳-۲۸ شبکهٔ SLAM برای شبیه‌سازی صف تک خدمت‌دهنده.

```
GEN,BANKS AND CARSON,SINGLE SERVER QUEUE EXAMPLE,5/31/1982,1;
LIMITS,1,0,30;  MODEL CAN USE 1 FILE,MAX NO. OF SIMULTANEOUS ENTRIES 30
NETWORK;        BEGINNING OF MODEL
     CREATE,EXPON(4.5);               CUSTOMERS ARRIVE AT CHECKOUT
     QUEUE(1);                        CUSTOMERS WAIT FOR SERVICE IN QUEUE FILE ONE (1)
     ACTIVITY(1)/1,RNORM(3.2,.6);     CHECKOUT SERVICE TIME IS N(3.2, 0.6)
     TERMINATE,1000;                  SIMULATE UNTIL 1000 CUSTOMERS ARE CHECKED OUT
     ENDNETWORK;                      END OF MODEL
FIN;            END OF SIMULATION
```

شکل ۳-۲۹ مدل SLAM صف تک خدمت‌دهنده.

غیر صفر مدلسازی کرد. گره CREATE پیشامد ورود را معرفی می‌کند. در فواصل مشخصی یک نهاد CREATE می‌شود و شروع به پیمودن شبکه می‌کند. نماد گره CREATE، جملهٔ مربوط به SLAM و عملوندهای منتخب در زیر نشان داده شده است:



CREATE, TBC, TF;

TBC_ مدت بین ایجاد نهادها _

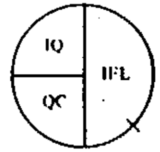
TF_ زمان ایجاد نهاد اول

(عملوندهای دیگر، MA، MC و M در پریتسکر و پگدن [۱۹۷۹] توضیح داده شده است. بحث این بخش محدود به گرهها و عملوندهای مورد نیاز در مثال ۳-۹ است.) در شکل ۳-۲۸ می‌بینیم TBC=EXPON(۴/۵) است، یعنی مدتهای بین دو ورود توزیع نمایی با میانگین ۴/۵ واحد زمان دارد. عملوند TF حذف شده است. مقدار ضمنی آن صفر گرفته می‌شود، که بدین ترتیب، اولین ورود در زمان صفر شبیه‌سازی رخ می‌دهد. به عبارت دیگر، TF ممکن است مقداری ثابت، مثلاً ۱۰۰ = TF، باشد که این نکته معرف وقوع اولین ورود در زمان ۱۰۰ است؛ یا اینکه TF ممکن است احتمالی، مثلاً TF=EXPON(۴/۵) باشد.

در شبکهٔ شکل ۳-۲۸، در پی گره CREATE شاخه‌ای می‌آید که نیاز به مدت زمان

شبیه‌سازی دارد. این‌گونه شاخه‌ها رابط نامیده می‌شود و برای مرتبط‌سازی دو گره در شبکهٔ تصویری قرار داده می‌شود. رابط نیاز به هیچ جمله‌ای ندارد.

گره بعدی، یعنی گره QUEUE، معرف تأخیر یا انتظار مشروط است. یعنی محلی را معرفی می‌کند که تا زمان شروع خدمت نهادها در آن به انتظار می‌ماند. نماد گره QUEUE، جملهٔ نظیر SLAM، و عملوندهای منتخب در زیر نشان داده شده است:



QUEUE (IFL), IQ, QC;

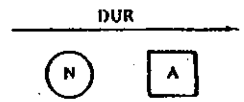
IFL ـ شمارهٔ پرونده برای ذخیره‌سازی نهادهای منتظر

IQ ـ تعداد اولیه در صف

QC ـ بیشترین تعداد حاضر در صف

عملوند IFL شمارهٔ پرونده است و پرونده (یا مجموعه)‌ای را مشخص می‌کند که نهادها (و ویژگیهای آنها) تا زمانی که خدمتدهی بتواند شروع شود در آن ذخیره می‌شود. نهادها تنها وقتی در پرونده ذخیره می‌شود که تمام خدمت‌دهنده‌ها مشغول باشند. در غیر این صورت، نهاد فوراً مشغول خدمتگیری می‌شود. نظام ضمنی صف برای پرونده‌ها، به‌ترتیب ورود است، هر چند که می‌توان از نظامهای پیچیده‌تر نیز استفاده کرد. مقدار ضمنی عملوند IQ مساوی صفر و مقدار ضمنی QC مساوی «بینهایت» گرفته می‌شود. استفاده از IQ اجازهٔ غیرخالی بودن صف در زمان صفر را می‌دهد. اما به‌کارگیری QC محدود شدن فضای انتظار را میسر می‌کند. [اگر گره QUEUE به هنگام ورود یک نهاد در بالاترین ظرفیت خود، QC، باشد عملوندهای دیگر (که فهرستشان در اینجا آورده نشده) تکلیف ورود جدید را که انصراف از ورود یا انسداد ورود است تعیین می‌کنند.] در شکل ۳-۲۸، IFL مقدار ۱ را دارد و IQ و QC حذف شده‌اند. بدین ترتیب، صف انتظار موسوم به «پروندهٔ ۱» در ابتدا خالی است و ظرفیت نامحدود دارد. آمار در مورد متوسط تعداد نهاد در پرونده (یعنی متوسط طول صف انتظار) و متوسط مدت انتظار نهادها در پرونده به‌طور خودکار گردآوری می‌شود.

در شکل ۳-۲۸، در پی گره QUEUE یک شاخهٔ ACTIVITY می‌آید که نماد، جملهٔ نظیر، و عملوندهای منتخب آن به شرح زیر است:



ACTIVITY (N)/A, DUR;

N ـ تعداد خدمت‌دهنده‌های موازی؛

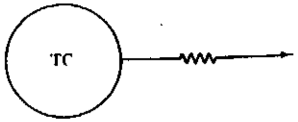
A ـ شمارهٔ فعالیتکه توسط برنامه‌نویس داده می‌شود؛

DUR ـ مدت فعالیت؛

هر شاخه معرف یک فعالیت، یعنی مدت مشخصاً تعریف شده‌ای مانند مدت خدمتدهی است. عملوند N تعداد خدمت‌دهنده‌های یکسان موازی را مشخص می‌کند. عملوند A شمارهٔ فعالیت است و توسط برنامه‌نویس به منظور معرفی یگانهٔ یک گروه از خدمت‌دهنده‌ها داده می‌شود. آمار مربوط به بهره‌برداری هر گروه از این خدمت‌دهنده‌ها به‌طور خودکار گردآوری می‌شود. از عملوند DUR به منظور تعریف مدت تداوم فعالیت استفاده می‌شود. مدت فعالیت، DUR،

ممکن است ثابت، تابعی از ویژگیهای سیستم (مثلاً تابعی از طول صف)، یا تصادفی باشد. در شکل ۳-۲۸، ۱ = N نشان‌دهندهٔ یک خدمت‌دهنده، ۱ = A معرف فعالیت شمارهٔ یک و DUR=RNORM(۳٫۲, ۰٫۶) نشان‌دهندهٔ این است که مدتهای خدمتدهی به‌صورت نمونه‌هایی از توزیع نرمال با میانگین ۳٫۲ و انحراف معیار ۰٫۶ واحد زمان (دقیقه) تولید می‌شود.

هر شاخه در مدل SLAM باید یک گرهٔ آغازی و یک گرهٔ پایانی داشته باشد. هر گره QUEUE معمولاً گره آغازی شاخهٔ مشخصی است. گره پایانی می‌تواند هر نوع گره دیگری باشد. عنوان گره پایانی فعالیت یا یک گره دیگر به‌منزلهٔ عملوند، مشخصاً در جملهٔ ACTIVITY آورده می‌شود. [مثلاً، جملهٔ "ACTIVITY(۱)/۱, RNORM(۳٫۲, ۰٫۶),, NLBL" مشخص می‌کند که NLBL عنوان گره پایانی است. به هرگرهی می‌توان عنوان داد. توجه داشته باشید که هر عملوند جا افتاده با ویرگولهای متوالی (بدون هیچ عملوندی بین آنها) مشخص می‌شود.] گره پایانی شاخه در شکل ۳-۲۸، گره TERMINATE است که پیشامد ترک را معرفی می‌کند؛ یعنی نهاد سیستم را ترک می‌کند و سابقه‌اش از بین می‌رود. نماد و عملوند این گره به شرح زیر است:



TERMINATE, TC;

TC ـ شنارشگر موارد پایانی

اگر عملوند TC یک عدد صحیح مثبت باشد، شبیه‌سازی متوقف می‌شود هرگاه تعداد TC نهاد در این گره به پایان رسیده باشد. (اگر چند گره TERMINATE وجود داشته باشد، شبیه‌سازی با رسیدن به نخستین شمارشگر موارد پایانی، TC، تمام می‌شود. اگر TC در یک گره مشخص TERMINATE سفید گذاشته شود، گره مزبور نهادهای واردشونده را از بین می‌برد ولی برای متوقف کردن شبیه‌سازی مورد استفاده قرار نمی‌گیرد.) شمارشگر موارد پایانی در شکل ۳-۲۸ TC = ۱۰۰۰ است، یعنی شبیه‌سازی تا خدمتدهی به ۱۰۰۰ متقاضی اجرا خواهد شد.

مدل شبکه‌ای در شکل ۳-۲۸ به شرح شکل ۳-۲۹ به جملات SLAM ترجمه شده است. توجه داشته باشید که تمام جملات SLAM با یک سمی‌کالن پایان می‌گیرد. علاوه بر جملات شبکه، جملات کنترلی افزوده نیز وجود دارد. سه جملهٔ نخست، اطلاعات عمومی برای راه‌اندازی مدل را فراهم می‌کند. اولین جملهٔ الزامی، ...GEN است و شناسایی و سایر اطلاعات عمومی را تأمین می‌کند. جملهٔ الزامی دوم، LIMITS است و تعداد پرونده‌ها، ماکسیمم تعداد ویژگیهای هر نهاد و ماکسیمم طول تمام پرونده‌ها به‌صورت ترکیبی را مشخص می‌کند. جملهٔ سوم، NETWORK و جملهٔ ماقبل آخر، ENDNETWORK، باید دربرگیرندهٔ تمام جملات شبکهٔ SLAM (گرهها و شاخه‌ها) باشد. جملهٔ FIN همواره آخرین جمله در میان کارتهاست.

آمار خروجی شبیه‌سازی SLAM در شکل ۳-۳۰ نشان داده شده است. براوردهـای زیـر

```
S L A M   S U M M A R Y   R E P O R T

SIMULATION PROJECT SINGLE SERVER QUEUE          BY BANKS AND CARSON

DATE 5/31/1982                                  RUN NUMBER   1 OF

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 ۳-۲۰ گزارش خروجی SLAM برای صف تک خدمت‌دهنده.

به‌دست آمده است:

- ماکسیمم طول صف انتظار = ۸
- بهره‌برداری از خدمت‌دهنده = ۰٫۷۶۹۴

متوسط مدت پاسخ و درصد متقاضیانی که مدت پاسخی بزرگتر از یا مساوی با ۴ دقیقه دارند براورد نشده است. به منظور بررسی این آمار می‌توان جملات اضافی SLAM را به برنامه افزود.

به‌طور خلاصه دیده می‌شود که همانند مدل GPSS، مدل SLAM نیز در مقایسه با مدل FORTRAN به جملات بسیارکمتری نیاز دارد. SLAM از GPSS تواناتر است. زیرا اگر نمایش شبکه‌ای ناکافی باشد، SLAM توانایی ترکیب مدلسازی شبیه به GASP با جملات شبکهٔ دارای گرایش پردازشی را دارد. روی‌هم‌رفته SLAM خصوصیات مطلوب بسیاری را در اختیار تحلیلگر شبیه‌سازی قرار می‌دهد. ■

۳-۳ خلاصه و مقایسهٔ زبانهای شبیه‌سازی[۱]

در این فصل مقدمه‌ای کوتاه بر پنج زبان‌ـ FORTRAN، GASP IV، SIMSCRIPT II.5، GPSS V و SLAMـ که در هر پروژهٔ شبیه‌سازی قابل استفاده‌اند، ارائه کردیم. به هنگام تصمیم‌گیری در این زمینه که کدام زبان در مورد پروژهٔ خاصی به‌کار گرفته شود ضوابط متعددی وجود دارد. برخی از این ضوابط در جدول ۳-۸ که عرضه‌کنندهٔ مقایسه‌ای از پنج زبان مورد بحث در این فصل است ارائه شده است. به‌هر صورت، اگر شبیه‌ساز زبانی را از قبل می‌داند و زبان مزبور نیز از عهدهٔ مدلسازی سیستم مورد نظر برمی‌آید، همین آشنایی می‌تواند ضابطهٔ کنارزنندهٔ بقیهٔ زبانها شود. فراگیری هر زبان به وقت و کوشش قابل‌ملاحظه‌ای نیاز دارد.

به‌طور کلی، GPSS و بخش پردازش‌ـتقابل SLAM از لحاظ فراگیری، به‌ویژه برای غیر برنامه‌نویسان راحت‌ترین است. بهایی که به ازای این امتیاز پرداخت می‌شود این است که زبانهای پردازش‌ـتقابل عموماً از توانایی و انعطاف کمتری برخوردارند. در SLAM و GASP (که بر FORTRAN پایه دارد) و در SIMSCRIPT، شبیه‌ساز نهایت توان و انعطاف یک زبان کامل برنامه‌نویسی را که محاسبات عددی پیچیده و مدلسازی وضعیتهای نامتعارف متنوع را میسر می‌کند، در اختیار دارد. شنون [۱۹۷۵] بحثی تفصیلی در زمینهٔ انتخاب یک زبان شبیه‌سازی و ضوابط گزینش ارائه می‌کند.

---

۱. به منظور مطالعهٔ بیشتر در این زمینه به منبع زیر که ۵۵ بستهٔ نرم‌افزار شبیه‌سازی را مورد ارزیابی قرار می‌دهد مراجعه کنید:

Swain J. James, *Flexible Tools for Modeling, OR/MS Today*, December 1993.

www.matlabdl.com

جدول ۳-۸ مقایسهٔ زبانها برای شبیه‌سازی گسسته پیشامد.

ضابطه	زبان: FORTRAN	GASP	SIMSCRIPT II.5	GPSS V	SLAM
سهولت فراگیری	خوب	خوب	خوب	عالی	عالی
سهولت درک مسأله	ضعیف	متوسط	خوب	عالی(الف)	عالی(الف)
سیستم مورد گرایش رهیافت مدلسازی	هیچ(ب)	همه	همه	صف	همه
زمانبندی پیشامدها	نه(ب)	بله	بله	نه	بله
پردازش‌متقابل	نه(ب)	نه	بله	بله	بله
پیوسته امکانات	نه(ب)	بله	بله	نه	بله
نمونگیری تصادفی درونی	نه(ج)	بله	بله	نه(د)	بله
توان گردآوری آمار	ضعیف	عالی	عالی	خوب(د)	عالی
توان فهرست‌پردازی	ضعیف	خوب	عالی	متوسط	خوب
سهولت دریافت گزارش استاندارد	ضعیف	عالی	متوسط	عالی	عالی
سهولت طراحی گزارش ویژه	متوسط	خوب	عالی	ضعیف(د)	خوب
غلطیابی	متوسط	خوب	عالی	متوسط(د)	خوب
مدت اجرا روی کامپیوتر	عالی(ه)	خوب	خوب	ضعیف(د)	خوب
مستندسازی از لحاظ فراگیری زبان و از دید مراجعه	بسیار خوب	بسیار خوب	متوسط	بسیارخوب	بسیار خوب
خود مستند ساخته بودن رمز	ضعیف	خوب	خوب	عالی	خوب
هزینه	پایین(و)	پایین(و)	بالا	پایین (GPSS/H بالا)	متوسط

الف) فهم دیاگرام بلوکی (شبکه‌ای) در مورد مدلهای صف عالی است.
ب) FORTRAN گرایش شبیه‌سازی سیستم ندارد. برنامه‌نویس گرایش موردنظر را ایجاد می‌کند و رهیافت مدلسازی موردنظر را برمی‌گزیند.
ج) زیربرنامه‌های علمی کتابخانه‌ای متعدد (مثلاً IMSL) به منظور تولید مقادیر تصادفی، برنامه‌های FORTRAN دارد.
د) CPSS/H از این جهات نسبت به GPSS V بسیار بهبود پیدا کرده است.
ه) با این فرض که مدل به کاراترین وجه برنامه‌نویسی شده باشد، FORTRAN سریع خواهد بود.
و) معمولاً در اکثر مراکز محاسبات موجود است.

منابع

CACI, Inc. [1976], *SIMSCRIPT II.5 Reference Handbook*, Los Angeles.

Delfosse, C. M. [1976], *Continuous Simulation and Combined Simulation in SIMSCRIPT II.5*, CACI, Inc., Arlington, Va.

Gordon, Geoffrey [1975], *The Application of GPSS V to Discrete System Simulation*, Prentice-Hall, Englewood Cliffs, N.J.

Gordon, Geoffrey [1978], *System Simulation*, 2nd ed., Prentice-Hall,

Englewood Cliffs, N.J.

Henriksen, J. O. [1979], *The GPSS/H User's Manual*, Wolverine Software, Falls Church, Va.

Kiviat, P. J., R. Villanueva, and H. M. Markowitz [1973], *SIMSCRIPT II.5 Programming Language*, ed. by E. C. Russell, CACI, Inc., Los Angeles.

Law, A. M., and W.D. Kelton [1982], *Simulation Modeling and Analysis*, McGraw-Hill, New york.

Pritsker, A. A. B. [1974], *The GASP IV Simulation Language*, Wiley, New York.

Pritsker, A. A. B., and C. Dennis Pegden [1979], *Introduction to Simulation and SLAM*, Wiley, New York.

Russell, E. C. [1983], *Building Simulation Models with SIMSCRIPT II.5*, CACI, Inc., Los Angeles.

Russell, E. C. [1976], *Simulating with Processes and Resources in SIMSCRIPT, II.5*, CACI, Inc., Los Angeles.

Russell, E. C., and J. S. Annino [1979], *A Quick Look at SIMSCRIPT II.5*, CACI, Inc., Los Angeles.

Schriber, Thomas J. [1974], *Simulation Using GPSS*, Wiley, New York.

Shannon, R. E. [1975], *Systems Simulation: The Art and Science*, Prentice-Hall, Englewood Cliffs, N.J.

تمرینها

دستورالعمل: در مورد اکثر این تمرینها، اولاً باید مدلی ساخت که به‌طور صریح موارد زیر در آن تعریف شود:

۱. حالت سیستم
۲. نهادهای سیستم و ویژگیهای آنها
۳. مجموعه‌ها و نهادهایی که می‌توان آنها را در مجموعه‌ها قرار داد
۴. پیشامدها و فعالیتها
۵. متغیرهای مورد نیاز برای گردآوری آمار تجمعی

ثانیاً، دانشجو باید یا ۱) به منظور تهیهٔ مقدمات استفاده از رهیافت زمانبندی پیشامدها، (همانند

شکلهای ۳-۵ و ۳-۶ در مثال ۳-۲) منطق پیشامد را به وضوح توضیح دهد، یا ۲) به منظور تهیهٔ مقدمات استفاده از رهیافت پردازش-تقابل، (همانند شکل ۳-۴) پردازشهای سیستم را تشریح کند. سرانجام اینکه، تا جایی که جز این در مسأله تصریح نشده است، دانشجو باید مدل را به یک زبان همه‌منظوره (مانند FORTRAN) یا به یک زبان خاص شبیه‌سازی (مثل GPSS، SIMSCRIPT یا SLAM) برنامه‌نویسی یا برخی از مسائل ساده‌تر را با دست شبیه‌سازی کند.

اکثر مسائل فعالیتهایی را دربر می‌گیرد که در فاصلهٔ $[a,b]$ توزیع یکنواخت دارد. هرگاه از یک زبان شبیه‌سازی استفاده می‌کنید، فرض کنید که وقوع تمام مقادیر بین a و b ممکن است؛ یعنی مدت فعالیت متغیری تصادفی و پیوسته است. هرگاه با دست شبیه‌سازی را انجام می‌دهید، تنها مقادیر ممکن را a، $a+1$، $a+2$، ...، b بگیرید؛ یعنی فرض کنید مدت فعالیت متغیری تصادفی و گسسته است. فرض گسستگی شبیه‌سازی دستی را آسان خواهد کرد.

توزیع یکنواخت با نماد $U(a,b)$، مشخص می‌شود که در آن a و b دو نقطهٔ انتهایی فاصله‌اند یا با نماد $m \pm h$ معرفی می‌شود، که m میانگین و h «نیم پهنه» توزیع است. این چهار پارامتر با معادله‌های

$$m = (a+b)/2 \qquad h = (b-a)/2$$

$$a = m - h \qquad b = m + h$$

به هم مرتبط‌اند. برخی از مولدهای مقادیر تصادفی موجود در زبانهای شبیه‌سازی نیاز به مشخص کردن a و b و بقیه نیاز به تعیین m و h دارد.

برخی از مسائل فعالیتهایی را دربر می‌گیرد که فرض می‌کنیم توزیع نرمال دارد. توزیع نرمال با نماد $N[\mu, \sigma^2]$ معرفی می‌شود که μ میانگین و σ^2 واریانس توزیع است. (چون مدتهای فعالیتها غیرمنفی است، توزیع نرمال تنها اگر $\mu \geq k\sigma$ باشد مناسب است. k دستکم ۴ و ترجیحاً ۵ یا بیشتر است. اگر مقداری منفی تولید شود به دور ریخته خواهد شد.) برخی دیگر از مسائل از توزیع نمایی با آهنگی مانند λ یا میانگین $\frac{1}{\lambda}$ استفاده می‌کنند. در فصل ۴ به بررسی این توزیعها پرداخته‌ایم؛ تولید مقادیر تصادفی برخوردار از این توزیعها نیز در فصل ۸ مورد بررسی قرار گرفته است. اکثر زبانها از امکان تولید سادهٔ نمونه‌هایی از این توزیعها برخوردار است. در مورد شبیه‌سازی با زبان FORTRAN، می‌توان از توابع ارائه شده در زیربخش ۳-۲-۱ به منظور تولید نمونه‌های توزیعهای نرمال و نمایی استفاده کرد.

۳-۱ (الف) با استفاده از رهیافت زمانبندی پیشامدها، شبیه‌سازی (دستی) باجهٔ صندوق را که در مثال ۳-۲، جدول ۳-۱ شروع شد ادامه دهید. از همان مدتهای بین ورود و خدمتدهی که قبلاً تولید و در مثال ۲-۱ به‌کارگرفته شد، استفاده کنید. با استفاده از آخرین مدت بین دو ورود، شبیه‌سازی را (بدون مجاز دانستن ورود جدید) ادامه دهید تا سیستم خالی شود. نتایج به‌دست آمده از این قسمت را با پاسخهای به‌دست آمده در مثال ۲-۱ مقایسه کنید. نتایج باید یکسان باشد.

(ب) تمرین ۳-۱ (الف) را با افزودن اجزاء ضروری به منظور براورد میانگین مدت پاسخ و درصد متقاضیانی که ۴ دقیقه یا بیشتر در سیستم می‌مانند بار دیگر انجام دهید. (*راهنمایی:* مثال ۳-۳، جدول ۳-۲ را ببینید.)

(ج) در زمینهٔ مزایای نسبی شبیه‌سازیهای دستی و کامپیوتری در مقایسه با یکدیگر اظهارنظر کنید.

۳-۲ دیاگرامهای منطق پیشامد را برای مسألهٔ کامیونهای کمپرسی، مثال ۳-۴، ایجاد کنید.

۳-۳ در مسألهٔ کامیونهای کمپرسی در مثال ۳-۴، براورد میانگین مدت پاسخ و درصد مدتهای پاسخ بزرگتر از ۳۰ دقیقه موردنظر است. هر مدت پاسخ برای یک کامیون از لحظهٔ ورود کامیون به صف بارگیری آغاز و در لحظه‌ای که توزین پایان می‌یابد تمام می‌شود. اجزاء مدل و آمار تجمعی مورد نیاز به منظور براورد این دو معیار عملکرد سیستم را اضافه کنید و شبیه‌سازی را به مدت ۸ ساعت انجام دهید.

۳-۴ اصلاحات لازم در مدل FORTRAN باجهٔ صندوق (مثال ۳-۶) را به منظور اجرای شبیه‌سازی دقیقاً به مدت ۶۰ ساعت انجام دهید. [توجه: بسته به اینکه چه کامپیوتری مورد استفاده قرارگیرد، RANF(.)، یعنی مولد اعداد تصادفی $U(0,1)$ تغییر داده خواهد شد.]

۳-۵ علاوه بر تغییر موضوع تمرین ۴، این فرض را نیز بکنید که متقاضیانی که در ساعت ۶۰ هنوز در باجه‌اند خدمت خواهند گرفت. ولی پس از زمان ۶۰ هیچ ورودی مجاز نخواهد بود. تغییرات لازم در برنامهٔ FORTRAN را ایجاد و مدل را اجرا کنید.

۳-۶ تغییرات موضوع تمرینهای ۴ و ۵ را در هر زبانی به‌کار گیرید (GASP، SIMSCRIPT، GPSS یا SLAM).

۳-۷ مثال ۲-۲ (مسألهٔ اتو رستوران هابیل و خباز) را با استفاده از رهیافت زمانبندی پیشامدها دوباره با دست شبیه‌سازی کنید.

۳-۸ مثال ۲-۴ [سیستم موجودی (M, N)] را با استفاده از رهیافت زمانبندی پیشامدها دوباره با دست شبیه‌سازی کنید.

۳-۹ مثال ۲-۵ (مسألهٔ تعویض برینگها) را با استفاده از رهیافت زمانبندی پیشامدها دوباره با دست شبیه‌سازی کنید.

۳-۱۰ در یک منطقهٔ وسیع شهری، آمبولانسها با آهنگ یک دستگاه در هر 15 ± 10 دقیقه روانهٔ مأموریت می‌شود. پانزده درصد از موارد درخواست خدمات آمبولانس، ساختگی است که نیازمند 12 ± 2 دقیقه وقت است. بقیهٔ درخواستها یکی از دو نوع زیر است. نوع اول، درخواستهای جدی است و ۱۵٪ از درخواستهای غیرساختگی را تشکیل می‌دهد که نیازمند 25 ± 5 دقیقه وقت برای کامل کردن است. کامل کردن درخواستهای باقیمانده به 20 ± 10 دقیقه وقت احتیاج دارد. فرض کنید تعداد بسیاری آمبولانس موجود است و در هر لحظه هر تعداد از آنها می‌تواند در حال انجام مأموریت باشد. سیستم را تا کامل شدن ۵۰۰ درخواست شبیه‌سازی کنید.

۳-۱۱ در تمرین ۱۰ میانگین مدت کامل کردن هر درخواست با یک آمبولانس را براورد کنید.

۳-۱۲ الف) در تمرین ۱۰ تصور کنید که تنها یک آمبولانس موجود است. هر درخواستی که در خلال بیرون بودن آمبولانس برسد باید به انتظار بماند. آیا یک آمبولانس می‌تواند از عهدهٔ حجم کار برآید؟

ب)شبیه‌سازی را با x آمبولانس ($x = ۱, ۲, ۳,$ یا ۴) انجام دهید و این چهار مورد را بر اساس مدت منتظر ماندن یک درخواست، درصد درخواستهایی که باید انتظار بکشند و درصد مدتی که آمبولانس به خاطر درخواست بیرون است، مقایسه کنید.

۳-۱۳ صیادی در حال شکار پرندگان مهاجر است. او ناگزیر از ماندن در وضعیت کنونی خود است تا با موفقیت ۲۰ پرنده را بکشد. شلیک کردن تفنگ ۱ ± ۲ ثانیه و پر کردن مجدد آن ۱ ± ۳ ثانیه وقت می‌گیرد. صیاد از یک تفنگ دولول استفاده می‌کند و حداکثر دوبار به هر پرنده شلیک و پس از شلیک به هر پرنده، تفنگ خود را پر می‌کند. پرندگان با آهنگ یک پرنده در هر ۲ ± ۱۰ ثانیه از فراز سر او می‌گذرند و صیاد از آهنگ موفقیت ۷۵ درصد در هر شلیک برخوردار است. کشتن ۲۰ پرنده چقدر از وقت صیاد را می‌گیرد؟

۳-۱۴ یک بزرگراه دو منطقهٔ بزرگ شهری را به هم متصل می‌کند. در هر ۱۵ ± ۲۰ ثانیه یک خودرو شهر اول را ترک می‌کند. بیست درصد از خودروها یک سرنشین، ۳۰ درصد دو سرنشین، ۱۰ درصد سه سرنشین و ۱۰ درصد چهار سرنشین دارند. ۳۰ درصد باقیماندهٔ خودروها اتوبوس است و هر اتوبوس ۴۰ نفر را حمل می‌کند. مدت مسافرت بین دو منطقهٔ شهری، ۱۰ ± ۶۰ دقیقه طول می‌کشد. چقدر طول می‌کشد تا ۵۰۰۰ نفر به شهر دوم وارد شوند؟

زمان بین دو ورود

۳-۱۵ افراد با آهنگ یک نفر در هر ۱۰ ± ۲۵ ثانیه به غرفهٔ فروش گوشت وارد می‌شوند. غرفه از دو قسمت تشکیل می‌شود: قسمت فروش گوشت دام و قسمت فروش گوشت ماکیان. نسبت تقاضای جنس توسط افراد به این شرح است: ۵۰ درصد فقط گوشت دام، ۳۰ درصد فقط گوشت ماکیان، ۲۰ درصد گوشت دام و ماکیان. خدمتدهی به هر سفارش یک مشتری، ۲۰ ± ۴۵ ثانیه از وقت یک قصاب را می‌گیرد. همهٔ مشتریها یک سفارش دارند به استثنای مشتریان «دام و ماکیان» که دو سفارش دارند. فرض کنید همواره به تعداد کافی قصاب برای پاسخگویی به تمام مشتریان حاضر وجود دارد. شبیه‌سازی را تا خدمتدهی به ۲۰۰ مشتری ادامه دهید.

۳-۱۶ در تمرین ۱۵، بیشترین تعداد قصابهای مورد نیاز در جریان شبیه‌سازی چند نفر است؟ آیا این تعداد به منظور تضمین اینکه هیچ‌گاه یک مشتری ناچار از انتظار کشیدن نباشد همواره کافی خواهد بود؟

۳-۱۷ تمرین ۱۵ را با x قصاب شبیه‌سازی کنید به‌طوری که x مساوی با ۱، ۲، ۳، و ۴ باشد. هرگاه تمام قصابها مشغول باشند، صفی تشکیل می‌شود. به ازای هر مقدار x میانگین تعداد قصابهای مشغول را براورد کنید.

۳-۱۸ آرایشگاهی با یک صندلی دارای آهنگ ورود یک نفر در هر ۱۵ ± ۲۰ دقیقه است. نیمی از مشتریها نیاز به کوتاه کردن مو دارند، ۳۰ درصد به آرایش و ۲۰ درصد تنها به اصلاح نیاز دارند. کوتاه کردن مو، ۵ ± ۱۵ دقیقه، آرایش ۱۰ ± ۲۵ دقیقه و اصلاح ۳ ± ۱۰ دقیقه وقت می‌گیرد. شبیه‌سازی را برای مراجعهٔ ۵۰ مشتری به آرایشگاه انجام دهید. درصد تنوع تقاضای خدمت برای هر نوع را با نتایج شبیه‌سازی مقایسه کنید. آیا نتایج منطقی است؟

۳-۱۹ فرودگاهی دو کریدور دارد. مسافران کریدور ۱ با آهنگ یک مسافر در هر ۲ ± ۱۵ ثانیه و مسافران کریدور ۲ با آهنگ یک مسافر در هر ۵ ± ۱۰ ثانیه از راه می‌رسند. پیمودن کریدور ۱، ۵ ± ۳۰ ثانیه و پیمودن کریدور ۲، ۱۰ ± ۳۵ ثانیه وقت می‌گیرد. هر دو کریدور به سالن اصلی باز می‌شود که در جنب قسمت دریافت چمدانها قرار دارد. رسیدن از سالن اصلی به قسمت دریافت چمدانها ۳ ± ۱۰ ثانیه وقت می‌گیرد. تنها ۶۰ درصد از مسافران به قسمت دریافت چمدانها می‌روند. گذر ۵۰۰ مسافر از سیستم فرودگاهی را شبیه‌سازی کنید. از این تعداد چند مسافر به قسمت دریافت چمدانها می‌روند؟ تا چه میزان می‌توان انتظار داشت که این براورد شبیه‌سازی به تعداد انتظاری نزدیک باشد؟ [تعداد انتظاری ۳۰۰ = (۵۰۰)(۰٫۶۰) است.]

۳-۲۰ بیماران با آهنگ یک نفر در هر ۲ ± ۵ دقیقه برای معاینات به یک درمانگاه چند مرحله‌ای وارد می‌شوند تا به بخش سنجش شنوایی بروند. معاینه ۱ ± ۳ دقیقه طول می‌کشد. هشتاد درصد بیماران بدون هیچ مسأله‌ای به آزمایش بعدی فرستاده می‌شوند. نیمی از ۲۰ درصد باقیمانده به آزمایشهایی سادهٔ نیاز دارند که ۱ ± ۲ دقیقه طول می‌کشد و متعاقباً برای معاینهٔ مجدد با همان احتمال عدم‌موفقیت فرستاده می‌شوند. نیمهٔ دیگر با تجویز دارو به خانه فرستاده می‌شوند. سیستم را شبیه‌سازی کنید تا معلوم شود چقدر طول می‌کشد تا ۲۰۰ مراجعه‌کننده با موفقیت آزمایشها را بگذرانند. (توجه: اشخاصی که با تجویز دارو به خانه فرستاده می‌شوند در شمار «موفقیتها» محسوب نمی‌شوند.)

۳-۲۱ بانکی را در نظر بگیرید که چهار تحویلدار دارد. تحویلدارهای ۳ و ۴ تنها به حسابهای تجاری می‌پردازند. اما تحویلدارهای ۱ و ۲ امور مربوط به حسابهای عمومی را انجام می‌دهند. مشتریان با آهنگ یک نفر در هر ۱ ± ۳ دقیقه به بانک وارد می‌شوند. ۳۳ درصد از مراجعات مربوط به حسابهای تجاری است. مراجعان هر نوع حساب از میان دو تحویلدار حاضر یکی را به‌طور تصادفی انتخاب می‌کنند. (فرض کنید که هر مشتری یک صف را بدون توجه به طول آن برمی‌گزیند و تغییر صف نمی‌دهد.) کامل کردن تقاضاهای مربوط به حسابهای تجاری ۱۰ ± ۱۵ دقیقه و حسابهای عمومی ۵ ± ۶ دقیقه وقت می‌گیرد. سیستم را تا کامل شدن راه‌اندازی ۵۰۰ تقاضا شبیه‌سازی کنید. هر نوع تحویلدار در چه درصدی از زمان مشغول است؟ متوسط مدتی که هر نوع مشتری در بانک می‌گذراند چقدر است؟

۳-۲۲ تمرین ۲۱ را با این فرض تکرار کنید که مشتریها به کوتاهترین صفی می‌پیوندند که حسابهای مربوط به آنها را انجام می‌دهد.

۳-۲۳ در تمرینهای ۲۱ و ۲۲ میانگین تأخیر مشتریان تجاری و مشتریان عادی را براورد کنید. (تأخیر عبارت از مدت سپری شده در صف انتظار است و مدت خدمتدهی را شامل نمی‌شود.) میانگین طول صف انتظار و میانگین نسبت مشتریانی را که بیش از یک دقیقه با تأخیر روبه‌رو می‌شوند نیز براورد کنید.

۳-۲۴ برای عملیات ماشینکاری روی یک قطعهٔ خاص سه ماشین مختلف به مدت یک ساعت در هر روز دسترسپذیر است. داده‌های مربوط به مدت انجام کار به شرح زیر است:

ماشین	مدت ماشینکاری روی یک قطعه (ثانیه)
۱	۲۰ ± ۴
۲	۱۰ ± ۳
۳	۱۵ ± ۵

فرض کنید قطعات توسط تسمهٔ نقاله با آهنگ یک قطعه در هر ۱۵ ± ۵ ثانیه در سه ساعت اول هر روز از راه می‌رسد. ماشین ۱ در اولین ساعت، ماشین ۲ در ساعت دوم و ماشین ۳ در ساعت سوم هر روز دسترسپذیر است. در یک روز چند قطعه تولید می‌شود؟ برای قطعه‌هایی که منتظر ماشین است انباری با چه وسعت مورد نیاز است؟ آیا این قطعه‌ها در زمانهای معینی روی هم «انباشته» می‌شود؟ چرا؟

۳-۲۵ افراد با آهنگ یک نفر در هر ۳۰ ± ۲۰ ثانیه به یک کافه تریای سلف سرویس وارد می‌شوند. چهل درصد به میز ساندویچ می‌روند که یک نفر آن را اداره و هر ۶۰ ± ۳۰ ثانیه یک ساندویچ درست می‌کند. بقیه به میز اصلی می‌روند که در آن یک خدمت‌دهنده در ۴۵ ± ۳۰ ثانیه غذای از قبل آماده شده را در بشقابی می‌ریزد. تمام مشتریان باید به صندوقدار واحدی پول غذا را بپردازند. خوردن غذا ۲۰ ± ۱۰ دقیقه از وقت تمام مشتریان را می‌گیرد. پس از صرف غذا، ۱۰ درصد از افراد دسر صرف می‌کنند که به این ترتیب ۱۰ ± ۲ دقیقه دیگر نیز در کافه تریا می‌مانند. شبیه‌سازی را تا ترک کافه تریا از سوی ۱۰۰ نفر انجام دهید. در زمانی که شبیه‌سازی متوقف می‌شود چند نفر در کافه تریا باقی می‌مانند و در حال انجام چه کاری هستند؟

۳-۲۶ اجزاء و قطعات یک هواپیمای باری C-5N با ۳۰ کامیون به‌طور همزمان از اتلانتا به مقصد بندر سه‌ونا حمل می‌شود. بر اساس تجارب قبلی می‌دانیم که انجام این سفر با یک کامیون ۶ ± ۲ ساعت وقت می‌گیرد. چهل درصد از رانندگان برای صرف قهوه در راه توقف می‌کنند که این ۱۵ ± ۵ دقیقه دیگر وقت می‌گیرد.

الف) وضعیت را چنین مدلسازی کنید: برای هر راننده ۴۰ درصد احتمال توقف برای صرف قهوه وجود دارد.

ب) وضعیت را چنان مدلسازی کنید که دقیقاً ۴۰ درصد از رانندگان برای صرف قهوه توقف کنند. آخرین کامیون در چه زمانی به سه‌ونا می‌رسد؟

۳-۲۷ مشتریان هر ۴۰ ± ۳۵ ثانیه یک بار به بانک «الف» وارد می‌شوند. در حال حاضر مشتریان به‌طور تصادفی یکی از دو تحویلدار را انتخاب می‌کنند. هر تحویلدار خدمتدهی به مشتری را در ۷۵ ± ۲۵ ثانیه تمام می‌کند. هر مشتری که به صف بپیوندد، تا کامل شدن خدمتگیری خود در آن صف می‌ماند. برخی از مشتریان مایل‌اند که بانک از روش تک صفی که توسط بانک «ب» به‌کارگرفته شده است استفاده کند. مشتریان کدام روش را سریعتر خواهند یافت؟ شبیه‌سازی را پیش از گردآوری هیچ‌گونه آماری به مدت ۱۵ دقیقه انجام دهید و سپس یک دورهٔ ۲ ساعته را شبیه‌سازی کنید. دو نظام صف را بر اساس بهره‌برداری از تحویلدار (درصد مدت اشتغال)، میانگین مدت تأخیر مشتریان و درصد مشتریانی که ناگزیرند (پیش از شروع خدمتدهی) بیش از یک دقیقه و بیش از سه دقیقه در انتظار بمانند مقایسه کنید.

۳-۲۸ شرکت وام‌ابزار در کار اجاره‌دادن اره‌های برقی است. مشتریان با آهنگ یک نفر در هر ۳۰ ± ۳۰ دقیقه برای اجاره کردن ارهٔ برقی از راه می‌رسند. دیوید و بتی مشتریان را راه‌اندازی می‌کنند. دیوید می‌تواند یک ارهٔ برقی را در ۱۴ ± ۴ دقیقه اجاره دهد، اما برای بتی این کار ۱۰ ± ۵ دقیقه طول می‌کشد. مشتریانی که اره‌های برقی را باز می‌گردانند نیز با همان آهنگ وارد می‌شوند که مشتریان اجاره‌کننده از راه می‌رسند. به منظور دریافت ارهٔ برقی بازگردانیده شده، دیوید یا بتی به مدت ۲ دقیقه با مشتری وقت صرف می‌کند. خدمتدهی بر اساس ضابطهٔ «به‌ترتیب ورود» است. هرگاه هیچ مشتری حاضر نباشد یا بتی به تنهایی مشغول باشد دیوید به آماده‌سازی اره‌های برقی بازگردانده شده برای اجاره‌دهی مجدد می‌پردازد. این نوع عملیات نگهداری ۶ ± ۴ دقیقه از وقت او را و تمیز کردن اره‌ها ۱۰ ± ۶ از وقت او را می‌گیرد. هرگاه دیوید مشغول نباشد به نگهداری یا تمیز کردن ارهٔ بعدی می‌پردازد. با به پایان رساندن کارهای نگهداری و تعمیر یا تمیز کردن یک اره اگر مشتری یا مشتریانی در انتظار باشند دیوید خدمتدهی به آنها را شروع می‌کند. بتی همواره برای خدمتدهی به مشتریان آمادگی دارد. عملیات سیستم را تحت این شرایط شبیه‌سازی کنید که در ساعت ۸:۰۰ صبح به‌صورت خالی شروع به‌کار کند، در ساعت ۶:۰۰ بعداز ظهر درها را ببندد ولی تا ساعت ۷:۰۰ بعدازظهر اره‌های برقی را برای اجاره‌دهی مجدد آماده کند. عمل نگهداری و تعمیر و تمیزکردن اره‌ها از ساعت ۶:۰۰ تا ۷:۰۰ بعدازظهر را دیوید و بتی با هم انجام می‌دهند. میانگین مدت تأخیر مشتریان اجاره‌کنندهٔ اره را براورد کنید.

۳-۲۹ در تمرین ۲۸ شیوهٔ معمول کارگاه در مورد نگهداری و تعمیر و تمیزکردن اره‌ها به منظور آماده‌سازی آنها برای اجاره‌دهی مجدد را تغییر دهید به‌طوری که اینک بتی تمام این کار را انجام دهد. به محض پایان تمیز کردن هر اره، در صورت وجود صف انتظار، بتی به کمک دیوید می‌رود. (یعنی، دیوید و بتی هر دو به مشتریان تازه خدمت می‌دهند و اره‌های بازگردانده شده را دریافت می‌کنند تا جایی که تنها دیوید مشغول بماند یا کارگاه خالی شود.) پس از این، بتی وظایف خود در زمینهٔ نگهداری و تعمیر و تمیزکردن را از سر می‌گیرد.

الف) میانگین مدت تأخیر مشتریانی را که ارهٔ برقی اجاره می‌کنند براورد کنید. بر این

اساس دو شیوهٔ عمل کارگاه را مقایسه کنید.

ب) نسبت مشتریانی را که باید بیش از ۵ دقیقه منتظر بمانند براورد کنید. بر این اساس دو شیوهٔ عمل کارگاه را مقایسه کنید.

ج) در مورد دیدگاههای مثبت و منفی مربوط به دو ضابطهٔ مندرج دربندهای (الف) و (ب) به منظور مقایسهٔ دو شیوهٔ عمل کارگاه بحث کنید. ضوابط دیگری را در این باره ارائه دهید.

۳۰-۳ دانشگاهی یک بابانهٔ کامپیوتر دارد. دانشجویان هر ۱۰ ± ۱۵ دقیقه برای استفاده از پایانه به مدت ۶ ± ۱۲ دقیقه به آن وارد می‌شوند. اگر پایانه مشغول باشد، ۶۰٪ از آنها پس از ۱۰ دقیقه برای استفاده از پایانه باز می‌گردند. اگر پایانه هنوز هم مشغول باشد ۵۰٪ (از ۶۰٪) پس از ۱۵ دقیقه باز می‌گردند. در مقابل ۵۰۰ نفری که عملاً از پایانه به‌طور کامل خدمت می‌گیرند چند دانشجو موفق به خدمتگیری از پایانه نمی‌شوند؟ تقاضا و خدمتدهی ۲۴ ساعته صورت می‌گیرد.

۳۱-۳ انباری گنجایش پذیرش ۱۰۰۰ مترمکعب کارتن را دارد. کارتنها سه اندازه دارند: کوچک (یک مترمکعب)، متوسط (۲ مترمکعب) و بزرگ (۳ مترمکعب). آهنگ ورود کارتنها به شرح زیر است: کوچک، هر ۱۰ ± ۱۰ دقیقه، متوسط، هر ۱۵ دقیقه، بزرگ هر ۸ ± ۸ دقیقه. اگر کارتنی از انبار برداشته نشود چقدر طول می‌کشد تا انبار خالی پر شود.

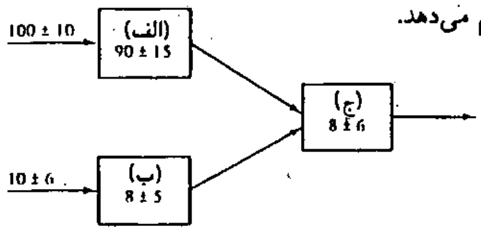
۳۲-۳ شرکتی به منظور انجام نیازهای داده‌پردازی خود یک دستگاه کامپیوتر خریداری کرده است. برنامه‌ها هر ۱۰ ± ۱۰ دقیقه از راه می‌رسد تا به‌صورت دسته‌ای یک به یک پردازش شود. پردازش ۷ ± ۷ دقیقه طول می‌کشد. سیستم کامپیوتر هر ۶۰ ± ۶۰ دقیقه از کار بازمی‌ماند. بازمانی ۴ ± ۸ دقیقه به درازا می‌کشد. پردازش برنامه‌ای که به سبب بازمانی کامپیوتر نیمه کاره می‌ماند، پس از اتمام بازمانی، از جایی که ناتمام مانده پیگیری می‌شود. عملیات این سیستم را برای ۲۴ ساعت شبیه‌سازی کنید. میانگین مدت پاسخ سیستم را براورد کنید. (هر مدت پاسخ سیستم برابر با طول مدت از لحظهٔ ورود تا لحظهٔ کامل شدن پردازش برنامه است.) میانگین مدت تأخیر را نیز برای برنامه‌هایی که وقوع بازمانی سیستم کامپیوتر در حال خدمتگیری‌اند براورد کنید.

۳۳-۳ (الف)، (ب) و (ج) سه خدمت‌دهندهٔ یک اتورستوران‌اند. خودروها هر ۵ ± ۵ دقیقه وارد می‌شوند. خدمت‌دهنده‌ها با آهنگ یک نفر در هر ۶ ± ۱۰ دقیقه به مشتریان خدمت می‌دهند. اما مشتریان، (الف) را به (ب) و (ب) را به (ج) ترجیح می‌دهند. اگر خدمت‌دهندهٔ مورد علاقه مشغول باشد، مشتریان اولین خدمت‌دهندهٔ دسترس‌پذیر را انتخاب می‌کنند. سیستم را تا تکمیل ۲۰۰ خدمتدهی شبیه‌سازی کنید. ضرایب بهره‌برداری (درصد مدت اشتغال) (الف)، (ب) و (ج) را براورد کنید.

۳۴-۳ عملیات یک اتومبیل‌شویی پنج مرحله دارد و هر مرحله ۱ ± ۲ دقیقه وقت می‌گیرد. فضای لازم برای به انتظار ماندن تا شروع عملیات شستشو تنها برای ۶ خودرو وجود دارد.

تأسیسات اتومبیل‌شویی گنجایش ۵ خودرو را دارد که به‌ترتیب سراسر سیستم را می‌پیمایند و هیچ خودروی نمی‌تواند حرکت کند مگر اینکه خودرو جلوی آن حرکت کند. هر ۲ ± ۲٫۵ دقیقه یک خودرو برای شستشو وارد می‌شود. اگر خودروی نتواند به سیستم راه یابد به اتومبیل‌شویی دیگری در آن سوی خیابان می‌رود. آهنگ امتناع در ساعت را براورد کنید، یعنی در ساعت چند خودرو به خاطر عدم امکان راهیابی به سیستم از ورود امتناع می‌کنند؟ شبیه‌سازی را برای یک روز ۱۲ ساعته انجام دهید.

۳۵-۳ سه ماشین (الف)، (ب) و (ج) به تصویر درآمده را در نظر بگیرید. ورود قطعه‌ها و مدتهای انجام کار به شرحی است که نشان داده شده است (مدتها به دقیقه است). ماشین (الف) کار قطعه‌های نوع (الف)، ماشین (ب) کار قطعه‌های نوع (ب) و ماشین (ج) کار هر دو نوع قطعه را انجام می‌دهد.



تمام ماشینها در معرض خطر شکستگی است: ماشین (الف) در هر ۳۵۰ ± ۴۰۰ دقیقه با ۱۴ ± ۱۵ دقیقه از کارماندگی، ماشین (ب) هر ۱۵۰ ± ۲۰۰ دقیقه با ۸ ± ۱۰ دقیقه از کارماندگی کار می‌کند و ماشین (ج) تقریباً هیچ‌گاه از کار نمی‌افتد که بدین ترتیب، از کارماندگی آن نادیده گرفته می‌شود. قطعه‌های رسیده از ماشین (الف) در اسرع وقت و پیش از قطعه‌های رسیده از ماشین (ب) یا ماشین (ج) راه‌اندازی می‌شود. هرگاه ماشین (الف) بشکند، هر قطعهٔ موجود در آن به ماشین (ب) فرستاده می‌شود تا به محض آزاد شدن (ب) راه‌اندازی آن بار دیگر از ابتدا در مدت ۲۰ ± ۱۰۰ دقیقه انجام شود. قطعه‌های رسیده از (الف) پیش از قطعه‌های منتظر در (ب) ولی پس از قطعه‌ای که در حال خدمتگیری است، در (ب) راه‌اندازی می‌شود. هرگاه ماشین (ب) بشکند راه‌اندازی قطعهٔ در حال خدمتگیری به محض آماده شدن (ب) پیگیری می‌شود. هر ماشین در هر لحظه از عهدهٔ راه‌اندازی یک قطعه برمی‌آید. شبیه‌سازی این سیستم را با دو دوباره‌سازی مستقل انجام دهید. هر دوباره‌سازی یک دورهٔ راه‌اندازی ۸ ساعته برای بارگذاری قطعه‌ها در سیستم و سپس یک اجرای ۴۰ ساعتهٔ حالت پایا را دربر می‌گیرد. (دوباره‌سازیهای مستقل بدین معناست که در هر اجرا از رشتهٔ متفاوتی از اعداد تصادفی استفاده شود.) مدیریت به سطح تولید بلندمدت [یعنی تعداد قطعات تولید شده در یک روز ۸ ساعته، از هر نوع ((الف) یا (ب))]، میزان بهره‌برداری از هر ماشین در بلندمدت و وجود تگناها («صفهای» طویل قطعه‌های منتظر) علاقه‌مند است. گزارش داده‌های خروجی را در جدولی به شرح زیر درج کنید:

www.matlabdl.com

	اجرای ۱	اجرای ۲	متوسط دو اجرا
بهره‌برداری از (الف)			
بهره‌برداری از (ب)			
⋮			

درگزارش خود شرح کوتاهی که خلاصهٔ نتایج مهم را دربرگیرد بگنجانید.

۳-۳۶ کارگران به منظور تهیهٔ ابزار با آهنگ یک نفر در هر ۴ ± ۱۰ دقیقه به فروشگاه ابزار مراجعه می‌کنند. هر تقاضا را یک نفر از سه کارمند فروشگاه راه‌اندازی می‌کند؛ هر کارمند برای راه‌اندازی هر تقاضا ۱۰ ± ۲۲ دقیقه وقت صرف می‌کند. سپس تمام تقاضاهای رسیده به یک صندوقدار داده می‌شود که برای هر تقاضا ۶ ± ۷ دقیقه وقت صرف می‌کند. سیستم را به مدت ۵۰ ساعت شبیه‌سازی کنید.

الف) بر اساس شبیه‌سازی ۵۰ ساعته، میزان بهره‌برداری هر کارمند را براورد کنید.

ب) به چند کارگر به‌طور کامل خدمت داده می‌شود؟ سه نفر کارمند به چند نفر خدمت می‌دهند؟ چند کارگر به فروشگاه وارد می‌شوند؟ آیا هیچ‌گاه هر سه کارمند به‌طور همزمان مشغول هستند؟ متوسط تعداد کارمندان مشغول چقدر است؟

۳-۳۷ افراد با آهنگ یک نفر در هر ۴٫۵ دقیقه به آرایشگاهی وارد می‌شوند. اگر آرایشگاه پر باشد (آرایشگاه گنجایش ۵ نفر را دارد)، ۳۰ درصد از مشتریان بالقوه محل را ترک می‌کنند و در مدت ۲۰ ± ۶۰ دقیقه باز می‌گردند. بقیه محل را ترک می‌کنند و باز نمی‌گردند. یک آرایشگر کوتاه کردن مو را در ۲ ± ۸ دقیقه به انجام می‌رساند، اما آرایشگر دوم بسیار حرف می‌زند و ۴ ± ۱۲ دقیقه وقت صرف می‌کند. اگر هر دو آرایشگر بیکار باشند مشتری آرایشگر اول را ترجیح می‌دهد. (با مشتریانی که سعی در ورود مجدد به آرایشگاه دارند به مانند مشتریان جدید رفتار کنید.) سیستم را تا زمان کوتاه کردن موی ۳۰۰ مشتری شبیه‌سازی کنید.

الف) آهنگ امتناع، یعنی تعداد کسانی را که در هر دقیقه به سبب ممکن نبودن راهیابی از ورود امتناع می‌کنند براورد کنید.

ب) تعداد امتناع کنندگان در دقیقه را که مجدداً باز نمی‌گردند براورد کنید.

ج) متوسط مدت سپری‌شده در آرایشگاه چقدر است؟

د) متوسط مدت سپری شده برای کوتاه کردن مو (بدون در نظرگرفتن مدت تأخیر) چقدر است؟

ه) متوسط تعداد مشتریان حاضر در آرایشگاه چقدر است؟

۳-۳۸ مردم با آهنگ یک نفر در هر ۲ ± ۸ دقیقه به یک نمایشگاه میکروسکوپی وارد می‌شوند. در هر لحظه تنها یک نفر می‌تواند نمایش را ببیند. دیدار از نمایش ۲ ± ۵ دقیقه وقت می‌گیرد. هر شخص می‌تواند یک بلیط ترجیحی به قیمت یک واحد پول بخرد که در صف

انتظار او را نسبت به کسانی‌که در مصرف یک واحد پول امساک کرده‌اند اولویت می‌دهد. در حدود ۵۰ درصد از بازدیدکنندگان این کار را می‌کنند اما تنها وقتی چنین تصمیمی را می‌گیرند که به هنگام ورودشان یک نفر یا بیشتر در صف باشد. نمایش از ساعت ۱۰ صبح تا ۴ بعدازظهر به‌طور پیوسته جریان دارد. عملیات سیستم را به مدت یک روز کامل شبیه‌سازی کنید. از محل فروش بلیطهای ترجیحی چه مقدار پول فراهم می‌آید؟

۳-۳۹ پیامها از اتاق اورژانس یک بیمارستان به انبار مرکزی فرستاده می‌شود و پاسخها به اتاق اورژانس باز می‌گردد. پیامها هر ۳ ± ۶ دقیقه تهیه می‌شود و از طریق یک لوله و به کمک هوای فشرده در عرض ۲ دقیقه به مقصد می‌رسد. در هر لحظه تنها یک پیام از طریق لوله قابل ارسال است. ۷۰ درصد از این پیامها نیاز به پاسخ دارد. آماده‌سازی هر پاسخ ۱ ± ۲ دقیقه وقت می‌گیرد. پیامهای اتاق اورژانس از اولویت برخوردار است. شبیه‌سازی را تا دریافت ۱۰۰ پاسخ از سوی اتاق اورژانس انجام دهید. میزان بهره‌برداری از لولهٔ هوای فشرده را براورد کنید. به‌طور متوسط چند پیام (و پاسخ) در انتظار ارسال می‌ماند؟

۳-۴۰ دو ماشین برای سوراخ کردن قطعه‌ها (نوع (الف) و نوع (ب)) وجود دارد. قطعه‌های نوع (الف) با آهنگ یک قطعه در هر ۳ ± ۱۰ دقیقه و قطعه‌های نوع (ب) با آهنگ یک قطعه در هر ۲ ± ۳ دقیقه از راه می‌رسد. در مورد قطعه‌های نوع (ب) کارگران ماشینی بیکار را انتخاب می‌کنند، یا اگر هر دو دستگاه مته (دویی و ترومن) مشغول باشد ماشینی را به‌طور تصادفی انتخاب می‌کنند و تا به آخر پای انتخاب خود می‌ایستند. قطعه‌های نوع (الف) باید در اسرع وقت سوراخ شود. بنابراین، اگر ماشینی (ترجیحاً دویی) آماده باشد به‌کار گرفته می‌شود؛ در غیر این‌صورت، قطعه به سر صف متهٔ دویی می‌رود. کامل کردن تمام تقاضاها ۳ ± ۴ دقیقه وقت می‌گیرد. شبیه‌سازی پایان صد سوراخکاری قطعه‌های نوع (الف) را انجام دهید. متوسط تعداد قطعه‌های نوع (الف) منتظر مته‌کاری را براورد کنید.

۳-۴۱ از یک خط تلفن در یک کلانتری هم برای تماسهای اضطراری و هم برای تماسهای شخصی استفاده می‌شود. تماسهای شخصی بر اساس ضابطهٔ به‌ترتیب ورود برقرار می‌شود و با آهنگ یک تماس در هر ۱ ± ۵ دقیقه وارد می‌شود. تماسهای اضطراری از اولویت برخوردار است و قطع سایر تماسها را ایجاب می‌کند. این نوع تماسها با آهنگ یک تماس در هر ۵ ± ۱۵ دقیقه انجام می‌شود. کامل کردن تماسهای اضطراری ۱ ± ۳ دقیقه وقت می‌گیرد اما تماسهای شخصی به ۲ ± ۲ دقیقه وقت نیاز دارد. بیست درصد از افرادی که از تلفن به‌صورت غیر اضطراری استفاده می‌کنند تمایل دارند در اولین زمان ممکن تماس تلفنی دیگری نیز بگیرند که به این افراد برای تماس دومشان پایین‌ترین اولویت داده می‌شود. تا کامل شدن ۲۰۰ تماس از همهٔ انواع، شبیه‌سازی را انجام دهید. ضریب بهره‌برداری از تلفن را براورد کنید.

۳-۴۲ یک مرکز فرعی کامپیوتر دو پایانه دارد. دانشجویان با آهنگ یک نفر در هر ۲ ± ۸ دقیقه از راه می‌رسند. استادانی که با آهنگ ورود یک نفر در هر ۲ ± ۱۲ دقیقه وارد می‌شوند،

می‌توانند موجب قطع کار دانشجویان شوند. یک تحلیلگر سیستم نیز وجود دارد که کار هر کسی را می‌تواند قطع کند هر چند که کار دانشجویان پیش از استادان قطع می‌شود. تحلیلگر سیستم روی پایانه ۴ ± ۶ دقیقه وقت صرف می‌کند و در ظرف ۵ ± ۲۰ دقیقه باز می‌گردد. استادان و دانشجویان به مدت ۲ ± ۴ دقیقه از پایانه استفاده می‌کنند. اگر کار شخصی قطع شود آن شخص به سر صف می‌پیوندد و در اولین زمان ممکن خدمتگیری را دنبال می‌کند. برای ۵۰ تقاضای استاد یا تحلیلگر سیستم شبیه‌سازی را انجام دهید. آهنگ قطع کار در ساعت و میانگین صف انتظار دانشجویان را براورد کنید.

۳-۴۳ قطعه‌ها با یک دستگاه متهٔ فشاری ماشین می‌شود. آنها با آهنگ یکی در هر ۳ ± ۵ دقیقه وارد می‌شود و ماشینکاری آنها ۲ ± ۳ دقیقه طول می‌کشد. هر یک ساعت، یک تقاضای تعجیلی وارد می‌شود که کامل کردن آن ۳ ± ۱۲ دقیقه وقت می‌گیرد. با رسیدن تقاضای تعجیلی راه‌اندازی تقاضای در دست انجام قطع می‌شود و در نوبت تقاضاهای عادی بعدی ماشین قرار می‌گیرد (ماشین فرایند ماشینکاری را از ابتدا انجام می‌دهد). ماشینکاری ۱۰ تقاضای تعجیلی را شبیه‌سازی کنید. میانگین مدت پاسخ سیستم برای هر نوع قطعه را براورد کنید. (هر مدت پاسخ، برابر با مجموع مدتی است که یک قطعه در سیستم سپری می‌کند.)

۳-۴۴ پیامها یک به یک با آهنگ یک پیام در هر ۱۰ ± ۳۵ ثانیه برای مخابره تولید می‌شود. مخابره ۵ ± ۲۰ ثانیه طول می‌کشد. در فواصل ۳ ± ۶ دقیقه، پیامهای فوری که ۳ ± ۱۰ ثانیه طول می‌کشد، خط مخابراتی را اشغال می‌کند. پیامهای در دست تولید باید به مدت ۲ دقیقه پردازش شود تا آمادهٔ مخابره شود. پیام آمادهٔ مخابره به سر صف می‌رود. به مدت ۹۰ دقیقه سیستم را شبیه‌سازی کنید. درصد مدتی را براورد کنید که خط در اشغال پیامهای عادی است.

۳-۴۵ کارگری به پر کردن جعبه‌هایی اشتغال دارد که با آهنگ یکی در هر ۳ ± ۱۵ دقیقه وارد می‌شود. پر کردن یک جعبه ۳ ± ۸ دقیقه وقت می‌گیرد. هر یک ساعت یک بار کار کارگر با وقفه روبه‌رو می‌شود تا او بسته‌بندی سفارشهای مخصوصی را انجام دهد که کار آن ۳ ± ۱۶ دقیقه طول می‌کشد. سپس سفارشی که بسته‌بندی آن دچار وقفه می‌شود بقیهٔ خدمت خود را دریافت می‌کند. شبیه‌سازی را برای مدت ۴۰ ساعت انجام دهید. درصد مدتی را براورد کنید که تعداد جعبه‌های منتظر برای پر شدن بیش از پنج عدد است.

۳-۴۶ هر ۱۹ ± ۴۰ دقیقه یک بیمار به اتاق اورژانس یک بیمارستان وارد می‌شود تا یکی از دو پزشک این بخش او را معاینه کند. بیست درصد از بیماران کسانی هستند که نیاز به رسیدگی فوری دارند، اما بقیه می‌توانند صبر کنند. به بیماران دستهٔ اول بالاترین اولویت، یعنی اولویت ۳ داده می‌شود تا در اسرع وقت پزشکی را به مدت ۳۷ ± ۴۰ دقیقه ببینند ولی بعد اولویت آنها به ۲ کاهش می‌یابد و به انتظار می‌مانند تا دوباره یک پزشک آزاد شود تا این بار به مدت ۲۵ ± ۳۰ دقیقه مداوا و سپس مرخص شوند. بیماران دستهٔ دوم ابتدا اولویت ۱ می‌گیرند و (وقتی نوبتشان برسد) به مدت ۱۴ ± ۱۵ دقیقه درمان می‌شوند.

سپس اولویتشان به ۲ افزایش می‌یابد و به انتظار می‌مانند تا یک پزشک آزاد شود و به مدت ۸ ± ۱۰ دقیقه آنها را به‌طور نهایی درمان کند و سرانجام مرخص شوند. شبیه‌سازی را برای مدت ۲۰ روز عملیات بدون وقفه (۲۴ ساعت در روز) انجام دهید. به منظور ایجاد باری از بیماران در سیستم، پیش از اجرای ۲۰ روزه، یک دورهٔ راه‌اندازی ۲ روزه در نظر بگیرید. شرایط را در صفر روز، ۲ روز و ۲۲ روز گزارش کنید. آیا یک دورهٔ راه‌اندازی دو روزه به اندازهٔ کافی بلند هست تا سطح باری که در سیستم ایجاد می‌کند به‌طور معقولی به شرایط حالت پایا نزدیک باشد؟

الف) متوسط و ماکسیمم طول صف از زمان ورود تا زمان دیدن اولین پزشک را برای بیماران دستهٔ اول اندازه‌گیری کنید. چند درصد اصلاً مجبور به انتظارکشیدن نیستند؟ مقادیر مدت انتظار اولیه را همراه با تصویر توزیع مقادیر نامبرده برای بیماران دستهٔ اول ارائه کنید. چند درصد این بیماران پیش از دیدن پزشک کمتر از پنج دقیقه انتظار می‌کشند؟

ب) مقادیر مربوط به جمع مدت سیستم برای تمام بیماران را همراه با نمودار توزیع آنها ارائه کنید. چندک ۹۰ درصد بیمارانی را که کمتر از x واحد زمان در سیستم می‌مانند براورد کنید. مقدار x را براورد کنید.

ج) مقادیر مربوط به بقیهٔ مدت ماندن در سیستم را (از پایان درمان اولیه تا مرخص شدن) همراه با نمودار توزیع آنها برای تمام بیماران ارائه دهید. چندک ۹۰ درصد را براورد کنید.

(تذکر: اکثر زبانهای شبیه‌سازی امکان ارائهٔ خودکار توزیع هر متغیر مشخص را فراهم می‌آورند.)

۳-۴۷ مردم به دکهٔ روزنامه‌فروشی چنان وارد می‌شوند که مدت بین ورودشان توزیع نمایی منفی با میانگین ۰٫۵ دقیقه داشته باشد. پنجاه و پنج درصد مردم تنها روزنامهٔ صبح را می‌خرند. ۲۵ درصد روزنامهٔ صبح و یک روزنامهٔ اقتصادی را خریداری می‌کنند. بقیه، تنها روزنامهٔ اقتصادی را می‌خرند. یک فروشنده مسئول فروش روزنامهٔ اقتصادی و فروشندهٔ دیگری مسئول فروش روزنامهٔ صبح است. شخصی که هر دو روزنامه را می‌خرد به فروشندهٔ روزنامهٔ اقتصادی مراجعه می‌کند. مدت خدمتدهی به هر مشتری برای هر خرید توزیع نرمال با میانگین ۴۰ ثانیه و انحراف معیار ۴ ثانیه دارد. در مورد صفهای هر نوع خرید آمار گردآوری کنید. راههایی را برای کاراتر شدن سیستم پیشنهاد دهید. شبیه‌سازی را برای مدت ۴ ساعت انجام دهید.

۳-۴۸ شخصی به کار ایجاد تغییر در مدل خانه‌ها و اضافه کردن اتاق به آنها اشتغال دارد. مدت زمان لازم برای انجام هر سفارش توزیع نرمال با میانگین ۱۷ روز و انحراف معیار ۳ روز دارد. فواصل زمانی بین امضای قرارداد انجام کار از سوی مالکین خانه‌ها توزیع نمایی منفی با میانگین ۲۰ روز دارد. شخص مورد بحث تنها یک گروه کارگر در اختیار دارد. میانگین مدت انتظار (از امضای قرارداد تا شروع کار) را برای سفارشهایی که انتظار مثبت دارد براورد کنید. درصد مدت بیکاری گروه کارگران را نیز براورد کنید. شبیه‌سازی را تا کامل شدن ۱۰۰ سفارش اجرا کنید.

۳-۴۹ قطعه‌ها به‌طور تصادفی با مدتهای بین ورود نمایی منفی و با میانگین ۶۰ ثانیه به ماشینی

می‌رسند. تمام قطعه‌ها به مدت ۵ ثانیه نیاز به آماده‌سازی و موازنه برای ماشینکاری دارد. قطعه‌ها به شرح درصدهای زیر بر سه نوع است. مدتهای ماشینکاری هر نوع قطعه توزیع نرمال با میانگین و انحراف معیاری به شرح زیر دارد:

نوع قطعه	درصد	میانگین(ثانیه)	σ(ثانیه)
۱	۵۰	۴۸	۸
۲	۳۰	۵۵	۹
۳	۲۰	۸۵	۱۲

توزیع مجموع مدت لازم برای کامل کردن ماشینکاری هر سه نوع قطعه را پیدا کنید. چند درصد از قطعه‌ها به بیش از ۶۰ ثانیه وقت برای کامل کردن ماشینکاری نیاز دارند؟ قطعه‌ها به‌طور متوسط چقدر باید در انتظار بمانند؟ شبیه‌سازی را برای یک روز ۸ ساعته انجام دهید.

۳-۵۰ معلوم شده است که مدتهای خرید در یک فروشگاه بزرگ دارای توزیع زیر است:

مدت خرید (دقیقه)	تعداد خریدکننده‌ها
۰-۱۰	۹۰
۱۰-۲۰	۱۲۰
۲۰-۳۰	۲۷۰
۳۰-۴۰	۱۴۵
۴۰-۵۰	۸۸
۵۰-۶۰	۲۸

مشتریان پس از خرید یکی از شش باجهٔ صندوق را انتخاب می‌کنند. مدتهای پرداخت پول به صندوق توزیع نرمال با میانگین ۵/۱ دقیقه و انحراف معیار ۰/۷ دقیقه دارد. مدتهای بین دو ورود یا توزیع نمایی منفی و میانگین یک دقیقه توزیع می‌شود. برای هر باجهٔ صندوق آمار (شامل آمار صف) گردآوری کنید. توزیع مدت کامل کردن خرید و مدت کامل کردن خرید و پرداخت پول به صندوق را در قالب جدولی ارائه کنید. چه درصدی از مشتریان بیش از ۴۵ دقیقه در فروشگاه می‌مانند؟ شبیه‌سازی را برای یک روز ۱۶ ساعته انجام دهید.

۳-۵۱ مدت بین ورود قطعاتی که نیاز دارند تا روی آنها کار شود به شرح زیر داده شده است:

مدت بین دو ورود (ثانیه)	درصد
۱۰-۲۰	۰/۲۰
۲۰-۳۰	۰/۳۰
۳۰-۴۰	۰/۵۰

قطعه‌ها بر سه نوع است: (الف)، (ب) و (ج). درصد هر قطعه و میانگین و انحراف معیار مدتهای نرمال انجام کار به شرح زیر است:

نوع قطعه	درصد	میانگین	انحراف معیار
(الف)	۰/۵	۳۰ ثانیه	۳ ثانیه
(ب)	۰/۳	۴۰ ثانیه	۴ ثانیه
(ج)	۰/۲	۵۰ ثانیه	۷ ثانیه

هر ماشین می‌تواند روی هر نوع قطعه‌ای به‌صورت انفرادی کار کند. از شبیه‌سازی به منظور مقایسهٔ عملیات یک ماشین با دو و با سه ماشین موازی استفاده کنید. برای چنین مقایسه‌ای چه ضوابطی مناسبت خواهد داشت؟

۳-۵۲ سفارشهایی برای یکی از چهار نوع قطعه دریافت می‌شود. مدت بین دو ورود سفارشها توزیع نمایی با میانگین ۱۰ دقیقه دارد. جدولی که در پی می‌آید درصد قطعات برحسب نوع و مدت لازم برای پاسخگویی یک کارمند به هر نوع سفارش را ارائه می‌دهد.

نوع قطعه	درصد	مدت خدمتدهی (دقیقه)
(الف)	۴۰	N[۶/۱, ۱/۳]
(ب)	۳۰	N[۹/۱, ۲/۹]
(ج)	۲۰	N[۱۱/۸, ۴/۱]
(د)	۱۰	N[۱۵/۱, ۴/۵]

سفارشهای نوع (الف) و (ب) پس از پیچیده شدن فوراً ارسال می‌شود، اما سفارشهای نوع (ج) و (د) باید ۵ ± ۱۰ دقیقه پیش از ارسال به انتظار بماند. توزیع مدت کامل کردن تحویل برای تمام انواع سفارش را در قالب جدولی ارائه دهید. چه درصدی از این توزیع کمتر از ۱۵ دقیقه وقت می‌گیرد؟ چه درصدی از آن کمتر از ۲۵ دقیقه طول می‌کشد؟ شبیه‌سازی را برای یک دورهٔ ۸ ساعتهٔ راه‌اندازی و در پی آن با یک اجرای ۴۰ ساعته انجام دهید. در خلال دورهٔ ۸ ساعتهٔ راه‌اندازی از گردآوری داده‌ها خودداری کنید.

۳-۵۳ هر سه دستگاه ماشین تولید نوعی ابزار کوچک به یک قطعهٔ اساسی نیاز دارد و باید در فواصل کوتاه زمانی عملیات نگهداری و تعمیر روی آنها انجام شود. به منظور افزایش تولید، تصمیم گرفته شده است که دو قطعهٔ یدکی تهیه شود (جمعاً ۲ + ۳ = ۵ قطعه). دو ساعت پس از استفاده، قطعه را از ماشین برداشته و به تکنیسین منحصر به‌فردی می‌دهیم که عملیات لازم نگهداری و

تعمیر را در ۳۰ ± ۲۰ دقیقه به انجام می‌رساند. پس از انجام عملیات نگهداری و تعمیر، قطعه در انبار قطعات یدکی قرار می‌گیرد تا روی اولین ماشینی که به آن نیاز دارد نصب شود. تکنیسین وظایف دیگری نیز دارد، مثل تعمیر سایر اقلامی که از اولویت بالاتری برخوردار است که هر ۶۰ ± ۲۰ دقیقه وارد می‌شود و نیاز به ۱۵ ± ۱۵ دقیقه خدمت دارد. در هر دورهٔ دو ساعته، تکنیسین یک استراحت ۱۵ دقیقه‌ای نیز دارد، یعنی او یک ساعت و ۴۵ دقیقه کار و ۱۵ دقیقه استراحت می‌کند. دوباره یک ساعت و ۴۵ دقیقه کار و ۱۵ دقیقه استراحت می‌کند و . . .

الف) شرایط اولیهٔ مدل چیست؟ به عبارت دیگر، در زمان صفر، قطعه‌ها کجایند و شرایطشان چیست؟ آیا این شرایط همانند شرایط «حالت پایا» است؟

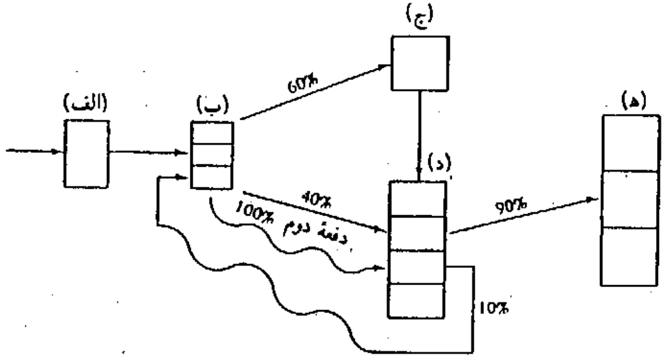
ب) هر دوباره‌سازی این تجربه را چنان انجام دهید که یک مرحلهٔ ۸ ساعتهٔ راه‌اندازی و به دنبال آن یک مرحلهٔ ۴۰ ساعتهٔ گردآوری داده‌ها را دربر گیرد. چهار دوباره‌سازی مستقل از این تجربه را تماماً در یک اجرای کامپیوتری انجام دهید. (یعنی چهار اجرا انجام دهید که هر یک از مجموعهٔ متفاوتی از اعداد تصادفی استفاده کند.)

ج) میانگین تعداد ماشینهای مشغول و درصد مدت اشتغال تکنیسین را برآورد کنید.

د) برآورد می‌شود که هزینهٔ قطعات برای شرکت ۴۰ واحد پول برای هر قطعه در هر روز ۸ ساعته (صرفنظر از مدت استفاده از آنها) باشد. هزینهٔ تکنیسین هر ساعت ۱۰ واحد پول است. هر ماشین در حال کار در هر ساعت تولید ابزاری به ارزش ۸۰ واحد پول تولید‌می‌کند. رابطه‌ای ارائه دهید که معرف هزینهٔ کل تولید ابزار در ساعت باشد (در واقع، تمام وقت تکنیسین صرف تولید ابزار نمی‌شود). این رابطه را بر اساس نتایج شبیه‌سازی ارزیابی کنید.

۳-۵۴ یک کارگاه انواع و اقسام ماشین‌آلات کوچک را تعمیر می‌کند. کارگاه از پنج ایستگاه کاری تشکیل می‌شود و جریان سفارشها در داخل کارگاه مطابق شکل صفحه بعد است. سفارشهای معمولی با آهنگ هر سفارش ۱۵ ± ۱۳ دقیقه به ایستگاه الف می‌رسد. سفارشهای تعجیلی هر ۴ ± ۳ ساعت وارد شده و بجز در ایستگاه ج که همراه همهٔ سفارشهای دیگر روی تسمهٔ نقاله قرارگرفته و عملیات تمیزکاری و چربی‌گیری روی آنها انجام می‌شود، در بقیه ایستگاهها از اولویت بالاتری برخوردار است. مدت رسیدگی به سفارشها و انجام تعمیرات در اولین بار ورود هر سفارش به هر ایستگاه به شرح صفحه بعد است.

مدتهای فوق در مورد تمام سفارشهایی که یکی از دو توالی (الف) ← (ب) ← (ج) ← (د) ← (ه) یا (الف) ← (ب) ← (د) ← (ه) را طی می‌کند درست است. اما، حدود ۱۰ درصد از سفارشهای خروجی از ایستگاه (د) به منظور انجام کارهای بیشتر (که ۳۰ ± ۱۰ دقیقه طول می‌کشد) به ایستگاه (ب) پس فرستاده می‌شود که از آنجا به (د) و سرانجام به (ه) می‌رود.



ایستگاه	تعداد ماشینها یا کارکنان	مدتهای رسیدگی یاتعمیر (دقیقه)	شرح
(الف)	۱	۱۲ ± ۲	دریافت کردن سفارش
(ب)	۳	۴۰ ± ۲۰	بازکردن ماشین و تعویض قطعه
(ج)	۱	۲۰	پاک کردن چربی
(د)	۴	۵۰ ± ۴۰	بستن قطعه‌ها و تنظیم
(ه)	۳	۴۰ ± ۵	بسته‌بندی و ارسال

مسیر این سفارشها به شرح زیر است:

(ج)

(الف) ← (ب) ← (د) ← (ب) ← (د) ← (ه)

ایستگاه (چربی‌گیری) (ج) هر دو ساعت یک بار، از یک ساعت پس از گشودن ایستگاه، به منظور انجام کارهای روزمرهٔ نگهداری و تعمیر بسته می‌شود که این کار ۱۰ ± ۱ دقیقه طول می‌کشد. اما این کارهای عادی نگهداری و تعمیر تا کامل شدن رسیدگی به ماشین احتمالی موجود در ایستگاه ج آغاز نمی‌شود.

الف) مدل شبیه‌سازی را مستقلاً ده دفعه دوباره‌سازی کنید به طوری که هر دوباره‌سازی معادل یک اجرای شبیه‌سازی ۸ ساعته از پی یک اجرای ۲ ساعتهٔ راه‌اندازی باشد. ده مجموعهٔ خروجی معرف ماندن یک سفارش در کارگاه است. معیار اصلی عملکرد موردنظر میانگین مدت پاسخ یعنی جمع مدت زمان ماندن یک سفارش درکارگاه است. هیچ‌گاه کارگاه در صبح خالی نیست ولی مدل در ابتدای دورهٔ راه‌اندازی خالی خواهد بود. بنابراین مدل را برای ۲ ساعت دوره راه‌اندازی اجرا و از ساعت ۲ تا ساعت ۱۰ اطلاعات گردآوری کنید. این دورهٔ گرم شدن از اریبی براورد میانگین مدت پاسخ به سمت پایین می‌کاهد. توجه کنید که دورهٔ ۲ ساعتهٔ گرم شدن،

قسمت دوم

مدلهای ریاضی و آماری

ابزاری برای دادن بار به مدل شبیه‌سازی در سطحی واقعی‌تر از سطح خالی است. برای هر یک از ده دوباره‌سازی مستقل براوردی از میانگین مدت پاسخ به‌دست آورید. همچنین با یافتن میانگین نمونهٔ ده‌تایی، براوردی کلی به‌دست آورید و همراه با براوردهای فاصله‌ای آن را ارائه کنید.

ب) مدیریت در صدد است که در مشغول‌ترین ایستگاه ((الف)، (ب)، (د) یا (هـ)) یک کارگر دیگر اضافه کند. آیا انجام این کار به‌طور قابل‌توجهی میانگین مدت پاسخ را بهبود می‌بخشد؟

ج) به‌عنوان گزینه‌ای دیگر در مقابل گزینهٔ ب)، مدیریت درصدد جایگزین کردن ماشین (ج) با ماشین سریعتری است که هر ماشین خدمت‌گیرنده را در ظرف ۱۶ دقیقه راه می‌اندازد. آیا این اقدام به‌طور قابل‌توجهی میانگین مدت پاسخ را بهبود می‌بخشد؟

۳-۵۵ یک بنگاه مصالح ساختمانی کامیونها را با دو دستگاه تراکتور بارگیری می‌کند. توزیع مدتهای بارگیری کامیونها مشخص شده که نمایی منفی با میانگین ۶ دقیقه است. مدتهای بین ورود کامیونها توزیع نمایی منفی با آهنگ ۱۶ ورود در ساعت دارد. براورد می‌شود که مدت انتظار یک کامیون و راننده ۴۰ واحد پول هزینه بردارد. اگر یک سیستم بارگیری متحرک سقفی نصب شود که هر کامیون را در مدت ثابت ۲ دقیقه پر کند، بنگاه (در هر روز ده ساعته) چقدر صرفه‌جویی می‌کند (اگر اصلاً چنین باشد)؟ (فرض کنید که تراکتورهای موجود به گونه‌ای رضایت‌بخش می‌توانند به نقاله‌های تغذیه‌کنندهٔ مخازن سیستم بارگیری سقفی خدمت بدهند.)

۳-۵۶ بخش ماشینهای فرز دارای ۱۰ ماشین است. مدت کارکرد تا بازمانی هر ماشین توزیع نمایی با میانگین ۲۰ ساعت دارد. مدتهای تعمیر توزیع یکنواخت بین ۳ و ۷ ساعت دارد. طول مناسب اجرا و شرایط مناسب شروع را انتخاب کنید.

الف) به منظور تضمین اینکه میانگین تعداد ماشینهای در حال کار بیش از ۸ است، به چند تعمیرکار نیاز است؟

ب) اگر دو تعمیرکار وجود داشته باشد، امید ریاضی تعداد ماشینهایی را که در حال کار یا در دست تعمیر است براورد کنید.

www.matlabdl.com

الف) اعداد تصادفی چگونه تولید می‌شود؟ آزمایش کنید و مطمئن شوید که مولد مطابق شیوه عمل می‌کند.

ب) به منظور انجام آزمایشهای تشریح شده در این فصل، برنامه‌هایی به زبان BASIC بنویسید. صد مجموعهٔ اعداد تصادفی تولید کنید به‌طوری که هر مجموعه ۱۰۰ عدد تصادفی داشته باشد. هر آزمایش را در مورد هر مجموعه از اعداد تصادفی اجرا و نتیجه‌گیری کنید.

ضمیمهٔ فصل ۷

کدهای کامپیوتری قابل حمل

در اینجا کدهای کامپیوتری مربوط به پیاده‌سازی روش همنهشتی ضربی با پیمانهٔ اول (PMMLCG) را که در انتهای زیربخش ۷-۳-۴ مورد بحث قرار گرفت به FORTRAN، پاسکال و C ارائه می‌کنیم. مولد مورد بحث دارای پیمانهٔ $m = m^* = ۲^{۳۱} - ۱ = ۲'۱۴۷'۴۸۳'۶۴۷$ و ضریب $a = ۶۳۰'۳۶۰'۰۱۶$ است. کدهایی که ارائه می‌کنیم همگی به‌طور نزدیکی مبتنی بر کد FORTRAN مارز و رابرتس[1] [۱۹۸۳] است که ایجاب می‌کند اعداد صحیح بین $-m^*$ و m^* به‌طور صحیح معرفی و محاسبه شود. این امر اغلب (ولی نه همیشه) در به‌کارگیری این زبانها روی اکثر ماشینها، از جمله ریز پردازنده‌ها میسر است. تابعی به زبان FORTRAN به نام NXSEED از مارز و رابرتس نیز نشان داده می‌شود، که هستهٔ شروع‌کنندهٔ هر یک از رشته‌هایی را با این فرض که همهٔ رشته‌ها دارای طول ۱۰۰۰۰۰ است، برمی‌گرداند.

اگر این دامنهٔ اعداد صحیح دسترسپذیر نیست، چند نحوهٔ به‌کارگیری این مولد با همان پیمانهٔ m^* ولی با ضریب (نه چندان مطلوب) $a_۱ = ۱۶۸۰۷$ نیز وجود دارد؛ برای نگارش DOUBLE PRECISION به FORTRAN، اسکریج[2] [۱۹۷۹] را ببینید و برای دستیابی به چند مورد به‌کارگیری مولد به زبان پاسکال، به اثر پارک و میلر[3] [۱۹۸۸] مراجعه کنید. چون این کدها به جای ریاضی مبتنی بر اعداد صحیح از اعداد اعشاری استفاده می‌کند، طبیعتاً کندتر است.

1. Marse, K, and S.D. Roberts [1983], "Implementing a Portable FORTRAN Uniform (0,1) Generator," Simulation, 41: pp. 135-139.
2. Schrage, L.[1979], "A More Portable Random Number Generator," Assoc, Comput. Mach. Trans. Math. Software, 5: pp. 132-138.
3. Park, S.K., and K.W. Miller [1988], "Random Number Generators: Good Ones Are Hard to Find," Commun. Assoc. Comput. Mach., 31: pp. 1192-1201.

۷ـ۱ـض FORTRAN

شکل ۷ـ۱ـض کد FORTRAN مارز و رابرتس [۱۹۸۳] را با اندکی اصلاح از لحاظ کنترل هسته از طریق نقاط ENTRY، به‌اضافهٔ هسته‌های پیش فرض تعبیه شده برای ۱۰۰۰ رشته با فاصلهٔ ۱۰۰۰۰۰ عدد نشان می‌دهد. توضیحات موجود درکد، طرز استفاده از آن را تشریح می‌کند. این برنامه را روی ریزپردازنده‌ها و همگردانهای زیر مورد آزمایش قرار داده‌ایم: IBM PC با FORTRAN حرفه‌ای IBM (نگارش ۱٫۰)، IBM PS/2 مدل ۵۰Z با FORTRAN مایکروسافت (نگارش ۴٫۰۱) و اپل‌مک‌اینتاش SE با FORTRAN ابسافت (نگارش ۲٫۴، با یک مورد عیب که در پی می‌آید). این برنامه روی ماشینهای زیر نیز مورد استفاده قرار گرفته است: DEC VAX ۸۶۵۰ با VAX FORTRAN، آنکور ملتیماکس ۳۲۰ با UMAX FORTRAN. همگردان ابسافت مورد استفاده روی مک اینتاش در زمینهٔ به‌کارگیری نقاط ENTRY عیب داشت که تغییر خط اول کد به REAL FUNCTION RAND (DUMMY, ISTRM) و آخرین نقطهٔ ENTRY به ENTRY IRANDG (DUMMY, ISTRM) و به تبع آن، تغییری در کاربرد، یعنی U = RAND(DUMMY, ۱)، و ... را ایجاب می‌کرد.

```
      REAL FUNCTION RAND(ISTRM)

*     Prime modulus multiplicative linear congruential generator
*     Z(I) = (630360016 * Z(I - 1)) (MOD(2**31 - 1)), based on Marse
*     and Roberts' portable random-number generator UNIRAN.  Multiple
*     (100) streams are supported, with seeds spaced 100,000 apart.
*     Throughout, input argument ISTRM must be an INTEGER giving the
*     desired stream number.

*     Usage: (Three options)
*
*     1. To obtain the next U(0,1) random number from stream ISTRM,
*        execute
*           U = RAND(ISTRM)
*        The REAL variable U will contain the next random number.
*
*     2. To set the seed for stream ISTRM to a desired value IZSET,
*        execute
*           CALL RANDST(IZSET,ISTRM)

*        where IZSET must be an INTEGER constant or variable set to the
*        desired seed, a number between 1 and 2147483646 (inclusive).
*        Default seeds for all 100 streams are given in the code.
*
*     3. To get the current (most recently used) integer in the
*        sequence being generated for stream ISTRM into the INTEGER
*        variable IZGET, execute
*           IZGET = IRANDG(ISTRM)

      INTEGER B2E15,B2E16,HI15,HI31,ISTRM,IZGET,IZSET,LOW15,LOWPRD,
     &        MODLUS,MULT1,MULT2,OVFLOW,ZI,ZRNG(100)
      INTEGER IRANDG,RANDST

*     Force saving of ZRNG between calls.

      SAVE ZRNG
```

شکل ۷ـ۱ـض کد FORTRAN مولد همنهشتی ضربی با پیمانهٔ اول با پارامترهای $m = 2^{31} - 1$ و $a = 630{,}360{,}016$ از مارز و رابرتس [۱۹۸۳].

```
*     Define the constants.

      DATA MULT1,MULT2/24112,26143/
      DATA B2E15,B2E16,MODLUS/32768,65536,2147483647/

*     Set the default seeds for all 100 streams.

      DATA ZRNG/1973272912, 281629770,  20006270,1280689831,2096730329,
     &          1933576050, 913566091, 246780520,1363774876, 604901985,
     &          1511192140,1259851944, 824064364, 150493284, 242708531,
     &            75253171,1964472944,1202299975, 233217322,1911216000,
     &           726370533, 403498145, 993232223,1103205531, 762430696,
     &          1922803170,1385516923,  76271663, 413682397, 726466604,
     &           336157058,1432650381,1120463904, 595778810, 877722890,
     &          1046574445,  68911991,2088367019, 748545416, 622401386,
     &          2122378830, 640690903,1774806513,2132545692,2079249579,
     &            78130110, 852776735,1187867272,1351423507,1645973084,
     &          1997049139, 922510944,2045512870, 898585771, 243649545;
     &          1004818771, 773686062, 403188473, 372279877,1901633463,
     &           498067494,2087759558, 493157915, 597104727,1530940798,
     &          1814496276, 536444882,1663153658, 855503735,  67784357,
     &          1432404475, 619691088, 119025595, 880802310, 176192644,
     &          1116780070, 277854671,1366580350,1142483975,2026948561,
     &          1053920743, 786262391,1792203830,1494667770,1923011392,
     &          1433700034,1244184613,1147297105, 539712780,1545929719,
     &           190641742,1645390429, 264907697, 620389253,1502074852,
     &           927711160, 364849192,2049576050, 638580085, 547070247/

*     Generate the next random number.

      ZI     = ZRNG(ISTRM)
      HI15   = ZI / B2E16
      LOWPRD = (ZI - HI15 * B2E16) * MULT1
      LOW15  = LOWPRD / B2E16
      HI31   = HI15 * MULT1 + LOW15
      OVFLOW = HI31 / B2E15
      ZI     = (((LOWPRD - LOW15 * B2E16) - MODLUS) +
     &         (HI31 - OVFLOW * B2E15) * B2E16) + OVFLOW
      IF (ZI .LT. 0) ZI = ZI + MODLUS
      HI15   = ZI / B2E16
      LOWPRD = (ZI - HI15 * B2E16) * MULT2
      LOW15  = LOWPRD / B2E16
      HI31   = HI15 * MULT2 + LOW15
      OVFLOW = HI31 / B2E15
      ZI     = (((LOWPRD - LOW15 * B2E16) - MODLUS) +
     &         (HI31 - OVFLOW * B2E15) * B2E16) + OVFLOW
      IF (ZI .LT. 0) ZI = ZI + MODLUS
      ZRNG(ISTRM) = ZI
      RAND   = (2 * (ZI / 256) + 1) / 16777216.0
      RETURN

*     Set the current ZRNG for stream ISTRM to IZSET.

      ENTRY RANDST(IZSET,ISTRM)
      ZRNG(ISTRM) = IZSET
      RETURN

*     Return the current ZRNG for stream ISTRM.

      ENTRY IRANDG(ISTRM)
      IRANDG = ZRNG(ISTRM)
      RETURN
```

www.matlabdl.com

۷ـ۲ـض پاسکال

شکل ۷ـ۲ـض کد پاسکال متشکل از چهار PROCEDURE جداگانه را برای این مولد ارائه می‌کند. توضیحات برنامه، دستورهای مشخص کاربرد آن را عرضه می‌دارد. در این شیوهٔ به‌کارگیری، PROCEDURE RANDDF باید پیش از استفاده از مولد RAND برای شروع کردن هسته‌های ۱۰۰ رشته فعال شود. افزودن تعریف VAR برای ZRNG که در توضیحات تذکر داده شده است نیز الزامی است (به این ترتیب، از نقطه‌نظر استفاده‌کننده، ZRNG به‌صورت کلمه‌ای ذخیره شده درمی‌آید). تعریف FUNCTION و FUNCTION FORWARD برای RANDDF، RAND، RANDST و RANDGT نیز لازم است. سپس لازم است این چهار PROCEDURE برای شبیه‌سازی در برنامهٔ پاسکال قرارگیرد خواه به‌طور فیزیکی با یک ویرایشگر، یا بافرمان include که به کامپایلر وابسته است. این کد را روی DEC VAX ۸۶۵۰ با VAX پاسکال و روی CRAY-2 با UNICOS پاسکال به‌کار برده‌ایم.

۷ـ۳ـض C

شکل ۷ـ۳ـض، کد این مولد را به ANSI C با سه تابع به شرح توضیحات درون برنامه ارائه می‌کند. شکل ۷ـ۴ـض نیز header file (rand.h) را ارائه می‌کند که کاربر باید برای اعلام توابع include# کند. این کد را ما روی دستگاه IBM PS/2 با توربو C (نگارش ۱/۵)، روی اپل‌مک‌اینتاش IIcx با دستگاه THINK C 4.0 (نام تابع rand موجود در کتابخانهٔ ANSI به منظور پرهیز از تضاد با کاربرد این نام توسط ما، باید تغییر داده می‌شد)، و روی DEC VAX ۸۶۵۰ با VAX C مورد استفاده قرار دادیم. در زمانی که این سطور نوشته می‌شود، برخی از همگردانهای C فاقد ANSI function prototyping است، در نتیجه، باید کد را اصلاح کرد تا با خارج کردن prototyping، با C «قدیمی» کار کند.

۷ـ۴ـض به‌دست آوردن هسته‌های شروع برای رشته‌ها

شکل ۷ـ۵ـض تابع NXSEED مارز و رابرتس [۱۹۸۳] را به FORTRAN ارائه می‌دهد که به‌عنوان ورودی، ISTRM را که معرف شمارهٔ رشتهٔ مورد نظر است، به‌صورت INTEGER دریافت و با همان نام آن، هستهٔ این رشته را باز می‌گرداند. فرض بر این است که رشته‌های مجاور، هر یک بلوکی به طول ۱۰۰۰۰۰ عدد تصادفی است. به این ترتیب، مثلاً (۳)NXSEED مقدار ۲۰۰۰۶۲۷۰ را باز می‌گرداند. فرض بر این است که هستهٔ رشتهٔ اول، ۱۹۷۳۲۷۲۹۱۲ است. چون طول توالی $۲۱۴۷۴۸۳۶۴۷ = ۲^{۳۱} - ۱$ است، ۲۱۴۷۴ رشته از این نوع وجود دارد که NXSEED هستهٔ شروع هر یک از آنها را پیدا می‌کند. به طوری که نوشته شده است، NXSEED هسته‌های شروع هر یک از رشته‌های ۱، ۲، ...، ISTRM را تعیین می‌کند ولی تنها آخرین آنها را باز می‌گرداند؛ البته می‌توان این برنامه را دوباره چنان نوشت که تمام آنها را در برداری INTEGER باز گرداند.

```
{ Prime modulus multiplicative linear congruential generator
  Z[I] = (630360016 * Z[I - 1]) (MOD 2147483647), based on Marse and
  Roberts' portable FORTRAN random-number generator UNIRAN.  Multiple
  (100) streams are supported, with seeds spaced 100,000 apart.
  Throughout, input argument Stream must be an Integer giving the
  desired stream number.  The initialization procedure Randdf described
  below must be invoked before using the generator, in order to set the
  seeds for the 100 predefined streams.

  The following declarations must appear in the program using this
  generator:

     VAR
        Zrng : ARRAY [1..100] OF Integer;

     PROCEDURE Randdf;                                          FORWARD;
     FUNCTION  Rand(Stream : Integer) : Real;                   FORWARD;
     PROCEDURE Randst(Zset : Integer; Stream : Integer);        FORWARD;
     FUNCTION  Randgt(Stream : Integer) : Integer;              FORWARD;

  Note that the name Zrng is thus reserved and cannot be used for any
  other purpose.

  Usage: (Four procedures)

  1. Before using the generator, it is required to initialize the
     routines by executing
        Randdf;
     This sets the initial seed values for all 100 streams in the array
     Zrng.

  2. To obtain the next U(0,1) random number from stream Stream,
     execute
        U := Rand(Stream)
     The Real variable U will contain the next random number.

  3. To set the seed for stream Stream to a desired value Zset, execute
        Randst(Zset, Stream)
     where Zset must be an Integer constant or variable set to the
     desired seed, a number between 1 and 2147483646 (inclusive).
     Seeds for all 100 streams are given in the code, and must be
     initialized by invoking Randdf.

  4. To get the current (most recently used) integer in the sequence
     being generated for stream Stream into the Integer variable Zget,
     execute
        Zget = Randgt(Stream);
}

PROCEDURE Randdf;

   BEGIN { Randdf }

      { Set the seeds for all 100 streams. }

      Zrng[ 1]:=1973272912; Zrng[ 2]:= 281629770; Zrng[ 3]:=  20006270;
      Zrng[ 4]:=1280689831; Zrng[ 5]:=2096730329; Zrng[ 6]:=1933576050;
      Zrng[ 7]:= 913566091; Zrng[ 8]:= 246780520; Zrng[ 9]:=1363774876;
      Zrng[10]:= 604901985; Zrng[11]:=1511192140; Zrng[12]:=1259851944;
      Zrng[13]:= 824064364; Zrng[14]:= 150493284; Zrng[15]:= 242708531;
      Zrng[16]:=  75253171; Zrng[17]:=1964472944; Zrng[18]:=1202299975;
      Zrng[19]:= 233217322; Zrng[20]:=1911216000; Zrng[21]:= 726370533;
      Zrng[22]:= 403498145; Zrng[23]:= 993232223; Zrng[24]:=1103205531;
      Zrng[25]:= 762430696; Zrng[26]:=1922803170; Zrng[27]:=1385516923;
      Zrng[28]:=  76271663; Zrng[29]:= 413682397; Zrng[30]:= 726466604;
      Zrng[31]:= 336157058; Zrng[32]:=1432650381; Zrng[33]:=1120463904;
      Zrng[34]:= 595778810; Zrng[35]:= 877722890; Zrng[36]:=1046574445;
      Zrng[37]:=  68911991; Zrng[38]:=2088367019; Zrng[39]:= 748545416;
      Zrng[40]:= 622401386; Zrng[41]:=2122378830; Zrng[42]:= 640690903;
```

شکل ۷ـ۲ـض کد پاسکال مولد همنهشتی ضربی با پیمانهٔ اول و پارامترهای $m = ۲^{۳۱} - ۱$ و $a = ۶۳۰۳۶۰۰۱۶$ از مارز و رابرتس [۱۹۸۳].

```
    Zrng[43]:=1774806513; Zrng[44]:=2132545692; Zrng[45]:=2079249579;
    Zrng[46]:=  78130110; Zrng[47]:= 852776735; Zrng[48]:=1187867272;
    Zrng[49]:=1351423507; Zrng[50]:=1645973084; Zrng[51]:=1997049139;
    Zrng[52]:= 922510944; Zrng[53]:=2045512870; Zrng[54]:= 898585771;
    Zrng[55]:= 243649545; Zrng[56]:=1004818771; Zrng[57]:= 773686062;
    Zrng[58]:= 403188473; Zrng[59]:= 372279877; Zrng[60]:=1901633463;
    Zrng[61]:= 498067494; Zrng[62]:=2087759558; Zrng[63]:= 493157915;
    Zrng[64]:= 597104727; Zrng[65]:=1530940798; Zrng[66]:=1814496276;
    Zrng[67]:= 536444882; Zrng[68]:=1663153658; Zrng[69]:= 855503735;
    Zrng[70]:=  67784357; Zrng[71]:=1432404475; Zrng[72]:= 619691088;
    Zrng[73]:= 119025595; Zrng[74]:= 880802310; Zrng[75]:= 176192644;
    Zrng[76]:=1116780070; Zrng[77]:= 277854671; Zrng[78]:=1366580350;
    Zrng[79]:=1142483975; Zrng[80]:=2026948561; Zrng[81]:=1053920743;
    Zrng[82]:= 786262391; Zrng[83]:=1792203830; Zrng[84]:=1494667770;
    Zrng[85]:=1923011392; Zrng[86]:=1433700034; Zrng[87]:=1244184613;
    Zrng[88]:=1147297105; Zrng[89]:= 539712780; Zrng[90]:=1545929719;
    Zrng[91]:= 190641742; Zrng[92]:=1645390429; Zrng[93]:= 264907697;
    Zrng[94]:= 620389253; Zrng[95]:=1502074852; Zrng[96]:= 927711160;
    Zrng[97]:= 364849192; Zrng[98]:=2049576050; Zrng[99]:= 638580085;
    Zrng[100]:= 547070247
END; { Randdf }

FUNCTION Rand; { Generate the next random number. }

  { Define the constants. }

  CONST
    B2E15  =      32768;
    B2E16  =      65536;
    Modlus = 2147483647;
    Mult1  =      24112;
    Mult2  =      26143;

  VAR
    Hi15, Hi31, Low15, Lowprd, Ovflow, Zi : Integer;

  BEGIN { Rand }

    { Generate the next random number. }

    Zi     := Zrng[Stream];
    Hi15   := Zi DIV B2E16;
    Lowprd := (Zi - Hi15 * B2E16) * Mult1;
    Low15  := Lowprd DIV B2E16;
    Hi31   := Hi15 * Mult1 + Low15;
    Ovflow := Hi31 DIV B2E15;
    Zi     := (((Lowprd - Low15 * B2E16) - Modlus) +
               (Hi31 - Ovflow * B2E15) * B2E16) + Ovflow;
    IF Zi < 0 THEN Zi := Zi + Modlus;
    Hi15   := Zi DIV B2E16;
    Lowprd := (Zi - Hi15 * B2E16) * Mult2;
    Low15  := Lowprd DIV B2E16;
    Hi31   := Hi15 * Mult2 + Low15;
    Ovflow := Hi31 DIV B2E15;
    Zi     := (((Lowprd - Low15 * B2E16) - Modlus) +
               (Hi31 - Ovflow * B2E15) * B2E16) + Ovflow;
    IF Zi < 0 THEN Zi := Zi + Modlus;
    Zrng[Stream] := Zi;
    Rand   := (2 * (Zi DIV 256) + 1) / 16777216.0

  END; { Rand }

PROCEDURE Randst;

  BEGIN { Randst }

    { Set the current Zrng for stream Stream to Zset. }

    Zrng[Stream] := Zset

  END; { Randst }
```

شکل ۷ـ۲ـض (ادامه).

```
/* Prime modulus multiplicative linear congruential generator
   Z[i] = (630360016 * Z[i-1]) (mod(pow(2,31) - 1)), based on Marse and
   Roberts' portable FORTRAN random-number generator UNIRAN.  Multiple
   (100) streams are supported, with seeds spaced 100,000 apart.
   Throughout, input argument "stream" must be an int giving the
   desired stream number.  The header file rand.h must be included in
   the calling program (#include "rand.h") before using these
   functions.

   Usage: (Three functions)

   1. To obtain the next U(0,1) random number from stream "stream,"
      execute
          u = rand(stream);
      where rand is a float function.  The float variable u will
      contain the next random number.

   2. To set the seed for stream "stream" to a desired value zset,
      execute
          randst(zset, stream);
      where randst is a void function and zset must be a long set to
      the desired seed, a number between 1 and 2147483646 (inclusive).
      Default seeds for all 100 streams are given in the code.

   3. To get the current (most recently used) integer in the sequence
      being generated for stream "stream" into the long variable zget,
      execute
          zget = randgt(stream);
      where randgt is a long function.
/* Define the constants. */

#define MODLUS 2147483647
#define MULT1       24112
#define MULT2       26143

/* Set the default seeds for all 100 streams. */

static long zrng[] =
{         0,
 1973272912, 281629770,  20006270,1280689831,2096730329,1933576050,
  913566091, 246780520,1363774876, 604901985,1511192140,1259851944,
  824064364, 150493284, 242708531,  75253171,1964472944,1202299975,
  233217322,1911216000, 726370533, 403498145, 993232223,1103205531,
  762430696,1922803170,1385516923,  76271663, 413682397, 726466604,
  336157058,1432650381,1120463904, 595778810, 877722890,1046574445,
   68911991,2088367019, 748545416, 622401386,2122378830, 640690903,
 1774806513,2132545692,2079249579,  78130110, 852776735,1187867272,
 1351423507,1645973084,1997049139, 922510944,2045512870, 898585771,
  243649545,1004818771, 773686062, 403188473, 372279877,1901633463,
  498067494,2087759558, 493157915, 597104727,1530940798,1814496276,
  536444882,1663153658, 855503735,  67784357,1432404475, 619691088,
  119025595, 880802310, 176192644,1116780070, 277854671,1366580350,
 1142483975,2026948561,1053920743, 786262391,1792203830,1494667770,
 1923011392,1433700034,1244184613,1147297105, 539712780,1545929719,
  190641742,1645390429, 264907697, 620389253,1502074852, 927711160,
  364849192,2049576050, 638580085, 547070247 };

/* Generate the next random number. */

float rand(int stream)
{
    long zi, lowprd, hi31;

    zi     = zrng[stream];
    lowprd = (zi & 65535) * MULT1;
    hi31   = (zi >> 16) * MULT1 + (lowprd >> 16);
    zi     = ((lowprd & 65535) - MODLUS) +
             ((hi31 & 32767) << 16) + (hi31 >> 15);
    if (zi < 0) zi += MODLUS;
    lowprd = (zi & 65535) * MULT2;
    hi31   = (zi >> 16) * MULT2 + (lowprd >> 16);
    zi     = ((lowprd & 65535) - MODLUS) +
```

شکل ۷ـ۳ـض کد C برای مولد همنهشتی ضربی با پیمانهٔ اول و پارامترهای $m = 2^{31} - 1$ و $a = 630{,}360{,}016$ از مارز و رابرتس [۱۹۸۳].

www.matlabdl.com

```
                ((hi31 & 32767) << 16) + (hi31 >> 15);
        if (zi < 0) zi += MODLUS;
        zrng[stream] = zi;
        return ((zi >> 7 | 1) + 1)/ 16777216.0;
    }

    /* Set the current zrng for stream "stream" to zset. */

    void randst (long zset, int stream)
    {
        zrng[stream] = zset;
    }

    /* Return the current zrng for stream "stream". */

    long randgt (int stream)
    {
        return zrng[stream];
    }
```

شکل ۷ـ۳ـض (ادامه).

```
/* The following 3 declarations are for use of the random-number
   generator rand and the associated functions randst and randgt for
   seed management.  This file (named rand.h) should be included in any
   program using these functions by executing
       #include "rand.h"
   before referencing the functions.
*/

float rand(int stream);
void randst(long zset, int stream);
long randgt(int stream);
```

شکل ۷ـ۴ـض header file به زبان C (rand.h) برای انضمام به کد C شکل ۷ـ۳ـض.

```
      INTEGER FUNCTION NXSEED(ISTRM)

*     Function from Marse and Roberts to return in its name the beginning
*     seed for stream ISTRM in the generator of Figs. 7.5, 7.6, and 7.7.
*     All streams are assumed to be of length 100,000 random numbers.

*     Usage:  To get the beginning seed for stream ISTRM into INTEGER
*     variable ISEED, execute
*        ISEED = NXSEED(ISTRM)
*     Input argument ISTRM is an INTEGER between 1 and 21,474.

      INTEGER I,SEED,ISTRM
      DOUBLE PRECISION Z
      Z = 1973272912
      DO 10 I = 1, ISTRM
         Z = DMOD( 715.D0*Z, 2147483647.D0)
         Z = DMOD(1058.D0*Z, 2147483647.D0)
         Z = DMOD(1385.D0*Z, 2147483647.D0)
10    CONTINUE
      NXSEED = IDINT(Z)
      RETURN
      END
```

شکل ۷ـ۵ـض تابع NXSEED به FORTRAN که برای مولد شکلهای ۷ـ۱ـض، ۷ـ۲ـض، و ۷ـ۳ـض هستهٔ شروع‌کنندهٔ هر یک از رشته‌ها (ی ۱۰۰۰۰۰تایی) را باز می‌گرداند.

۸

تولید مقدار تصادفی

این فصل به شیوه‌هایی برای نمونه‌گیری از انواع توزیعهای پیوسته وگسسته‌ای می‌پردازد که به طور گسترده‌ای مورد استفاده قرار می‌گیرد. بحثها و مثالهای پیشین از سیستمهای صف و موجودی، بر فایدهٔ توزیعهای آماری برای مدلسازی فعالیتهایی دلالت داشت که عموماً غیرقابل پیش‌بینی یا ناقطعی است. مثلاً، مدتهای بین دو ورود و مدتهای خدمتدهی در صفها و تقاضا برای یک محصول، دست‌کم تا حد معینی، اغلب ماهیتی غیرقابل پیش‌بینی دارند. معمولاً، چنین متغیرهایی به‌صورت متغیرهایی تصادفی با توزیع آماری مشخص مدلسازی می‌شود و برای براورد پارامترهای توزیع فرضی و آزمودن اعتبار مدل آماری مفروض، شیوه‌های استاندارد آماری وجود دارد. این شیوه‌ها در فصل ۹ مورد بحث قرارگرفته است.

در این فصل فرض می‌کنیم که توزیعی به‌طور کامل مشخص شده است و ما در جستجوی راههایی به منظور تولید نمونه‌هایی از این توزیع برای استفاده به‌عنوان ورودی به مدل شبیه‌سازی هستیم. هدف این فصل تشریح و نمایش متداولترین روشهای تولید مقادیر تصادفی، و نه ارائهٔ یک بررسی کاملاً جدید در زمینهٔ کاراترین روشهاست. در عمل، اکثر مدلسازان در شبیه سازی از برنامه‌های موجود دسترسپذیر به زبان FORTRAN (مثلاً، کتابخانهٔ IMSL) یا برنامه‌های موجود در زبان در دست استفاده، مانند برنامه‌های موجود در SIMSCRIPT، GASP و SLAM استفاده خواهند کرد. برخی از زبانها از قبیل GPSS از برنامه‌های مورد بحث بی‌بهره است و برخی مراکز محاسبات مولدهای مقادیر تصادفی به زبان FORTRAN را ندارد، به‌طوری که مدلساز باید برنامه‌ای مورد قبول را خود ایجاد کند. این فصل روش تبدیل معکوس، روش پیچش و به اختصار، روش رد و قبول را مورد بحث قرار می‌دهد.[1] روش دیگری به نام روش ترکیب را فیشمن

۱. برای مطالعهٔ بیشتر در زمینهٔ روشهای تولید مقادیر تصادفی، می‌توان به ضمیمه فصل ۸ رجوع کرد.م

www.matlabdl.com

[۱۹۷۸] مورد بحث قرار داده است. مشخصاً، نشان خواهیم داد که چگونه از تمام توزیعهای مورد بحث در فصل ۴ نمونه تولید می‌کنیم.

در همهٔ روشهای این فصل فرض می‌کنیم که یک منبع اعداد تصادفی یکنواخت (۱ ,۰)، R_1، R_2، . . . ، دسترس‌پذیر است که هر R_i دارای pdf

$$f_R(x)=\begin{cases}1, & 0\le x\le 1\\ 0, & \text{در غیر این صورت}\end{cases}$$

و cdf

$$F_R(x)=\begin{cases}0, & x<0\\ x, & 0\le x\le 1\\ 1, & 1<x\end{cases}$$

است. در سراسر این فصل، R و R_1، R_2، . . . معرف اعداد تصادفی برخوردار از توزیع یکنواخت [۱ ,۰] است و طبق یکی از روشهای فصل ۷ تولید یا از یک جدول اعداد تصادفی مانند جدول پ-۱ گرفته می‌شود. استفاده از جدول پ-۱ را در فصل ۲ تشریح کردیم.

۸-۱ روش تبدیل معکوس

روش تبدیل معکوس را می‌توان به منظور نمونه‌گیری از توزیعهای نمایی، ویبول و یکنواخت و توزیعهای تجربی مورد استفاده قرار داد. به علاوه، در نمونه‌گیری از انواع بسیار توزیعهای گسسته این روش اصل اساسی محسوب می‌شود. این روش به تفصیل برای توزیع نمایی توضیح داده و سپس بر توزیعهای دیگر اعمال می‌شود. این روش مستقیم‌ترین روش است ولی از لحاظ محاسباتی همواره کاراترین نیست.

۸-۱-۱ توزیع نمایی

توزیع نمایی که در بخش ۴-۴ مورد بحث قرار گرفت، دارای تابع چگالی احتمال (pdf)

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

و تابع توزیع تجمعی (cdf)

$$F(x)=\int_{-\infty}^{x} f(t)dt=\begin{cases}1-e^{-\lambda x}, & 0\le x\\ 0, & x<0\end{cases}$$

است. پارامتر λ را می‌توان به‌عنوان میانگین تعداد رخدادها در واحد زمان تعبیر کرد. مثلاً، اگر مدتهای بین ورود X_1، X_2، X_3، . . . توزیع نمایی با آهنگ λ داشته باشد، λ را می‌توان میانگین تعداد موارد ورود در واحد زمان یا آهنگ ورود تعبیر کرد. توجه داشته باشید که به ازای هر i داریم:

$$E(X_i)=\frac{1}{\lambda}$$

به‌طوری که $\frac{1}{\lambda}$ میانگین مدت بین دو ورود است. هدف در اینجا ارائهٔ شیوه‌ای برای تولید مقادیر X_1، X_2، X_3، . . . به گونه‌ای است که توزیع نمایی داشته باشد.

زمانی می‌توان روش تبدیل معکوس را به‌کار برد که شکل cdf، $F(x)$، چنان ساده باشد که معکوس آن، F^{-1}، از راه تحلیلی صریحاً قابل محاسبه باشد. شیوه‌ای گام به گام برای روش تبدیل معکوس که بر پایهٔ توزیع نمایی تشریح می‌شود به شرح زیر است:

گام ۱. cdf متغیر تصادفی مورد نظر، X، را محاسبه کنید.

برای توزیع نمایی، cdf عبارت از $F(x)=1-e^{-\lambda x}, 0\le x$ است.

گام ۲. فرض کنید $R=F(X)$ در دامنهٔ X برقرار است.

برای توزیع نمایی، این رابطه در دامنهٔ $x\ge 0$ به‌صورت $1-e^{-\lambda X}=R$ در می‌آید. چون X متغیری تصادفی (در این مورد با توزیع نمایی) است، نتیجه می‌شود که $1-e^{-\lambda X}$ نیز متغیری تصادفی، در اینجا به نام R، است. همان‌طور که بعداً نشان خواهیم داد، R در فاصلهٔ (۱ ,۰) توزیع یکنواخت دارد.

گام ۳. معادلهٔ $F(X)=R$ را حل کنید تا X برحسب R به‌دست آید.

جواب در مورد توزیع نمایی به شرح زیر به‌دست می‌آید:

$$\begin{aligned}1-e^{-\lambda X}&=R\\ e^{-\lambda X}&=1-R\\ -\lambda X&=\ln(1-R)\\ X&=-\frac{1}{\lambda}\ln(1-R)\end{aligned}\qquad (۸\text{-}۱)$$

معادلهٔ (۸-۱) را مولد مقدار تصادفی برای توزیع نمایی می‌نامند. به‌طور کلی، معادلهٔ (۸-۱) به صورت $X=F^{-1}(R)$ نوشته می‌شود. تولید دنباله‌ای از مقادیر طبق گام ۴ صورت می‌گیرد.

گام ۴. اعداد تصادفی یکنواخت R_1، R_2، R_3، . . . را (در صورت نیاز) تولید و مقادیر مورد نظر را طبق رابطهٔ

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

محاسبه کنید. در مورد توزیع نمایی، طبق معادلهٔ (۸-۱) داریم $F^{-1}(R) = (-\frac{1}{\lambda})\ell n(1 - R)$ به‌طوری که به ازای $i = 1, 2, 3, \ldots$

$$X_i = -\frac{1}{\lambda}\ell n(1 - R_i) \qquad \text{(۸-۲(الف))}$$

یک نوع از ساده‌سازی که معمولاً در معادلهٔ (۸-۲(الف)) انجام می‌شود، قرار دادن R_i به‌جای $1 - R_i$ است که رابطهٔ

$$X_i = -\frac{1}{\lambda}\ell n R_i \qquad \text{(۸-۲(ب))}$$

از این کار حاصل می‌شود. چون R_i و $1 - R_i$ هر دو در فاصلهٔ $(0, 1)$ توزیع یکنواخت دارد، این جانشینی موجه است.

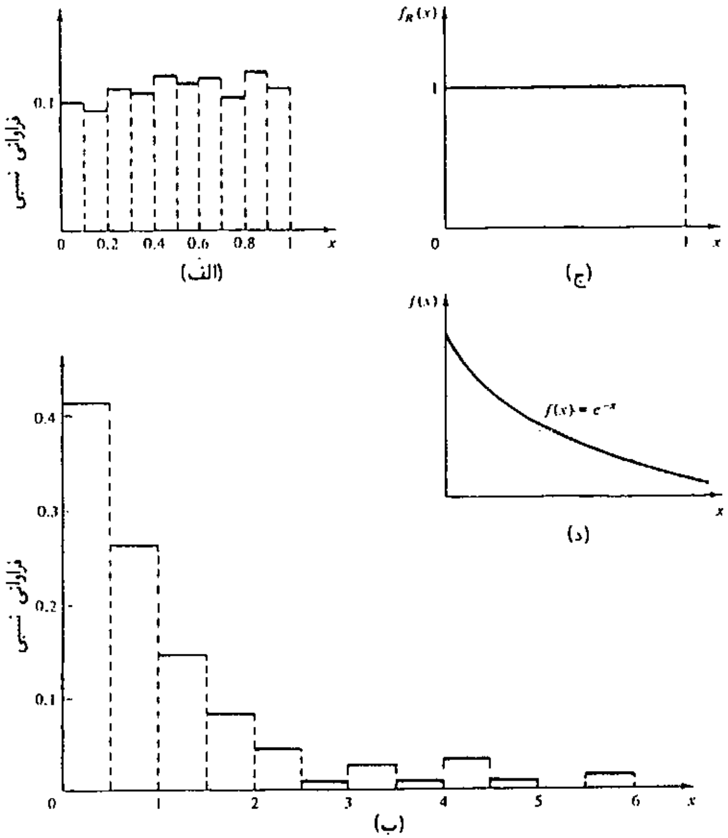
■ **مثال ۸-۱**

جدول ۸-۱ دنباله‌ای از اعداد تصادفی از جدول پ-۱ و مقادیر محاسبه شدهٔ نمایی را که به ازای مقدار $\lambda = 1$ از معادلهٔ (۸-۲(الف)) به‌دست آمده ارائه می‌دهد. شکل ۸-۱ (الف) هیستوگرم ۲۰۰ مقدار $R_1, R_2, \ldots, R_{200}$ از توزیع یکنواخت و شکل ۸-۱ (ب)، هیستوگرم ۲۰۰ مقدار $X_1, X_2, \ldots, X_{200}$ است که طبق معادلهٔ (۸-۲(الف)) محاسبه شده است. این هیستوگرمهای تجربی را با توابع چگالی نظری در شکلهای ۸-۱ (ج) و (د) مقایسه کنید. چنانکه در اینجا به تصویر کشیده شده است، هر هیستوگرم براوردی از تابع چگالی مبناست. (در فصل ۹ این واقعیت به‌عنوان راهی برای شناسایی توزیع به‌کارگرفته شده است.)

شکل ۸-۲ تعبیری تصویری از روش تبدیل معکوس ارائه می‌کند. cdf نشان داده شده، $F(x) = 1 - e^{-x}$، توزیعی نمایی با آهنگ $\lambda = 1$ است. به منظور تولید مقدار X_1 با تابع تجمعی $F(x)$، ابتدا عدد تصادفی R_1 را بین ۰ و ۱ تولید می‌کنیم و از R_1 خطی افقی به شکل cdf می‌کشیم، سپس خطی عمودی بر محور x فرود می‌آوریم تا نتیجهٔ دلخواه، یعنی X_1

جدول ۸-۱ تولید مقادیر تصادفی نمایی X_i با میانگین ۱ به ازای اعداد تصادفی R_i.

i	۱	۲	۳	۴	۵
R_i	۰٫۱۳۰۶	۰٫۰۴۲۲	۰٫۶۵۹۷	۰٫۷۹۶۵	۰٫۷۶۹۶
X_i	۰٫۱۴۰۰	۰٫۰۴۳۱	۱٫۰۷۸	۱٫۵۹۲	۱٫۴۶۸



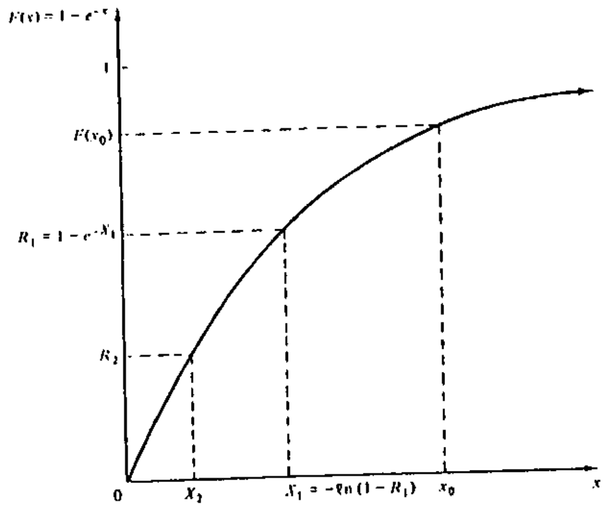
شکل ۸-۱ (الف) هیستوگرم تجربی ۲۰۰ عدد تصادفی یکنواخت؛ (ب) هیستوگرم تجربی ۲۰۰مقدار نمایی؛ (ج) چگالی نظری یکنواخت در فاصلهٔ $(0, 1)$؛ (د) چگالی نظری نمایی با میانگین ۱.

را به‌دست آوریم. به رابطهٔ معکوس بین R_1 و X_1، یعنی

$$R_1 = 1 - e^{-X_1}$$

و

$$X_1 = -\ell n(1 - R_1)$$



شکل ۸-۲ نمای ترسیمی روش تبدیل معکوس.

توجه کنید. به‌طور کلی، رابطه به صورت

$$R_1 = F(X_1)$$

و

$$X_1 = F^{-1}(R_1)$$

نوشته می‌شود. چرا متغیر تصادفی X_1 که با این شیوه تولید می‌شود، از توزیع مورد نظر برخوردار است؟ مقداری مانند X_0 را انتخاب و احتمال تجمعی

$$P(X_1 \leq x_0) = P(R_1 \leq F(x_0)) = F(X_0) \qquad (۸\text{-}۳)$$

را محاسبه کنید. برای دیدن تساوی اول در معادلهٔ (۸-۳) به شکل ۸-۲ مراجعه کنید که در آن اعداد ثابت x_0 و $F(x_0)$ به محورهای نظیر خود رسم شده است. می‌توان دید که رابطهٔ $X_1 \leq x_0$ وقتی و فقط وقتی درست است که رابطهٔ $R_1 \leq F(x_0)$ درست باشد. چون $0 \leq F(x_0) \leq 1$، تساوی دوم در معادلهٔ (۸-۳) پیامد فوری این واقعیت است که R_1 در فاصلهٔ (۰،۱) توزیع یکنواخت دارد.
■

۸-۱-۲ توزیع یکنواخت

یک متغیر تصادفی مانند X را در نظر بگیرید که در فاصلهٔ $[a, b]$ به‌طور یکنواخت توزیع شده است. حدسی معقول برای تولید X عبارت است از

$$X = a + (b - a)R \qquad (۸\text{-}۴)$$

[به یاد آورید که R همواره عددی تصادفی در فاصلهٔ (۰، ۱) است.] تابع چگالی X به‌صورت زیر ارائه می‌شود

$$f(x) = \begin{cases} \dfrac{1}{b-a}, & a \leq x \leq b \\ 0, & \text{در غیر این صورت} \end{cases}$$

تعیین معادلهٔ (۸-۴) پیامد برداشتن گامهای ۱ تا ۳ از زیر بخش ۸-۱-۱ است:

گام ۱. cdf به‌صورت زیر ارائه می‌شود

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

گام ۲. تساوی $F(X) = (X - a)/(b - a) = R$ را بنویسید.

گام ۳. حل X برحسب R به رابطهٔ $X = a + (b - a)R$ می‌انجامد که همان معادلهٔ (۸-۴) است.

۸-۱-۳ توزیع ویبول

توزیع ویبول به‌عنوان مدلی برای «مدت تا بازمانی» ماشین‌آلات یا قطعه‌های الکترونیک را در بخش ۴-۴ معرفی کردیم. هرگاه پارامتر موقعیت، ν، مساوی با صفر قرار داده شود، pdf آن طبق معادلهٔ (۴-۴۵) به‌صورت

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

در می‌آید که $\alpha > 0$ و $\beta > 0$ پارامترهای مقیاس و شکل توزیع است. به منظور تولید هر مقدار تصادفی ویبول، از گامهای ۱ تا ۳ از زیربخش ۸-۱-۱ پیروی کنید:

گام ۱. cdf به‌صورت $F(x) = 1 - e^{-(x/\alpha)^{\beta}}, x \ge 0$ ارائه می‌شود.
گام ۲. فرض کنید $R = F(X) = 1 - e^{-(X/\alpha)^{\beta}}$ باشد.
گام ۳. حل X برحسب R به نتیجهٔ زیر می‌انجامد

$$X = \alpha[-\ell n(1 - R)]^{\frac{1}{\beta}} \qquad (۸\text{-}۵)$$

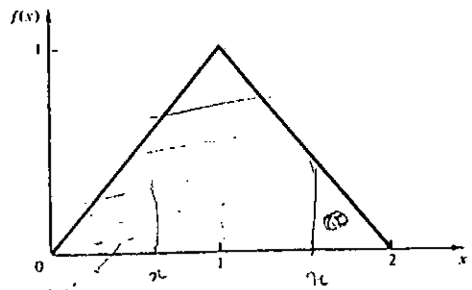
تعیین معادلهٔ (۸-۵) را به‌عنوان تمرینی به دانشجو وامی‌گذاریم. با مقایسهٔ معادله‌های (۸-۵) و (۸-۱) می‌توان دید که اگر X یک مقدار تصادفی ویبول باشد، در این صورت X^{β} یک مقدار تصادفی نمایی با میانگین α^{β} است. برعکس، اگر Y یک مقدار تصادفی نمایی با میانگین μ باشد، در این صورت $Y^{1/\beta}$ یک مقدار تصادفی ویبول با پارامتر شکل β و پارامتر مقیاس $\alpha = \mu^{1/\beta}$ است.

۸-۱-۴ توزیع مثلثی

متغیری تصادفی مانند X را در نظر بگیرید که دارای pdf

$$f(x) = \begin{cases} x, & 0 \le x \le 1 \\ 2 - x, & 1 < x \le 2 \\ 0, & \text{در غیر این صورت} \end{cases}$$

به گونهٔ نشان داده شده در شکل ۸-۳ باشد. این توزیع را توزیع مثلثی با نقاط انتهایی (۰،۲) و مُد



شکل ۸-۳ تابع چگالی مربوط به یک توزیع مثلثی.

یک می‌نامند. cdf توزیع به‌صورت

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

است. به ازای $0 \le X \le 1$ داریم

$$R = \frac{X^2}{2} \qquad (۸\text{-}۶)$$

و به ازای $1 \le X \le 2$ داریم

$$R = 1 - \frac{(2-X)^2}{2} \qquad (۸\text{-}۷)$$

به موجب معادلهٔ (۸-۶)، از $0 \le X \le 1$ چنین برمی‌آید که $0 \le R \le \frac{1}{2}$، که در این صورت، $X = \sqrt{2R}$ است. به موجب معادلهٔ (۸-۷)، از $1 \le X \le 2$ رابطهٔ $\frac{1}{2} \le R \le 1$ نتیجه می‌شود، که در این صورت داریم: $X = 2 - \sqrt{2(1-R)}$. بنابراین، X طبق رابطهٔ

$$X = \begin{cases} \sqrt{2R}, & 0 \le R \le \frac{1}{2} \\ 2 - \sqrt{2(1-R)}, & \frac{1}{2} < R \le 1 \end{cases} \qquad (۸\text{-}۸)$$

تولید می‌شود. تمرینهای ۲، ۳ و ۴ فرصت کار با سایر توزیعهای مثلثی را به دانشجو می‌دهد. توجه داشته باشید که اگر pdf و cdf متغیر تصادفی X چند تکه‌ای باشد (یعنی، نیاز به فرمولهای متفاوت در بخشهای مختلف دامنهٔ X داشته باشد)، در این صورت کاربرد روش تبدیل معکوس در مورد تولید X، در ابتدا بخشهای مختلف دامنهٔ R، همانند معادلهٔ (۸-۸)، به فرمولهای مجزا می‌انجامد. در بخش ۴-۴ شکلی کلی از توزیع مثلثی مورد بحث قرار گرفت.

۸-۱-۵ توزیعهای تجربی پیوسته

اگر مدلساز ناتوان از یافتن توزیعی نظری به منظور ارائهٔ مدل مناسبی برای داده‌های ورودی باشد، ممکن است استفاده از توزیع تجربی داده‌ها لازم شود.

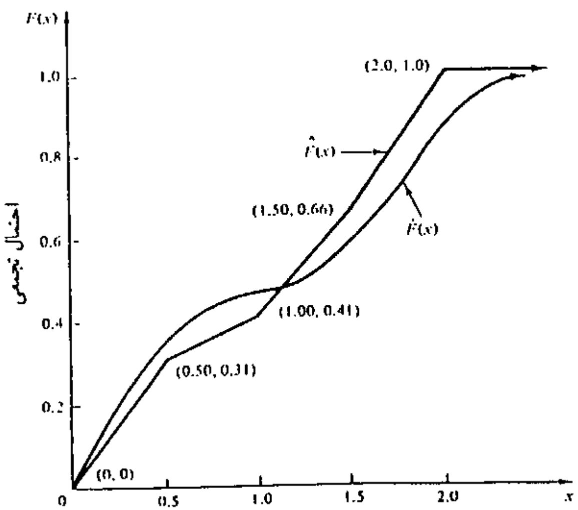
■ **مثال ۸-۲**

تصور کنید ۱۰۰ مورد مدت تعمیر نوعی ابزار شکسته گردآوری شده و داده‌ها برحسب تعداد مشاهده‌ها در فواصل مختلف، در جدول ۸-۲ خلاصه شده است. مثلاً، ۳۱ مشاهده بین صفر و ۰٫۵ ساعت، ۱۰ مشاهده بین ۰٫۵ و یک ساعت، و ... وجود دارد.

توزیع واقعی، $F(x)$، مدتهای تعمیر (خط با انحنا در شکل ۸-۴) را می‌توان به‌وسیلهٔ cdf

جدول ۸-۲ خلاصهٔ داده‌های مدت تعمیر.

فاصله (ساعت)	فراوانی	فراوانی نسبی	فراوانی تجمعی
۰ ≤ x ≤ ۰٫۵	۳۱	۰٫۳۱	۰٫۳۱
۰٫۵ < x ≤ ۱٫۰	۱۰	۰٫۱۰	۰٫۴۱
۰٫۰ < x ≤ ۱٫۰	۲۵	۰٫۲۵	۰٫۶۶
۱٫۵ < x ≤ ۲٫۰	۳۴	۰٫۳۴	۱٫۰۰



شکل ۸-۴ توابع توزیع تجربی و نظری، برای داده‌های مدت تعمیر ($X \geq ۰$).

تجربی، $\hat{F}(x)$ (منحنی پاره‌خطی در شکل ۸-۴) براورد کرد. شکل حقیقی $F(x)$ مجهول است و همواره در عمل مجهول خواهد ماند، مگر در صورتی که مقدار نامحدودی داده دسترسپذیر باشد. منحنی مشخص شده در شکل ۸-۴ یک شکل ممکن این توزیع مبنا را نمایش می‌دهد و همچنین، $\hat{F}(x)$ براوردی از $F(x)$ است.

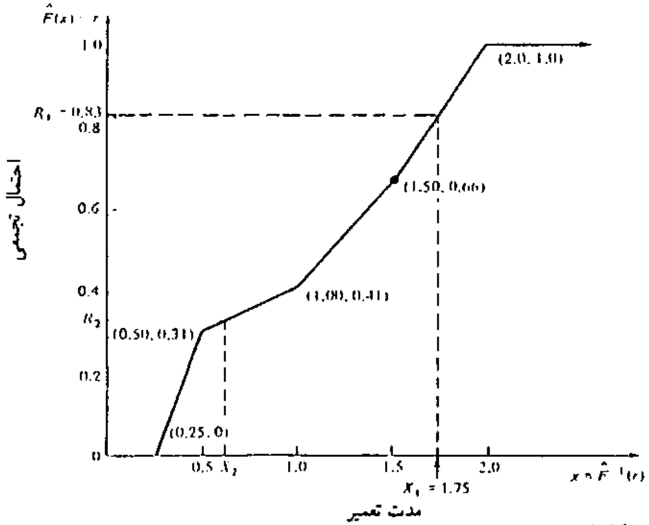
cdf تجربی $\hat{F}(x)$ با استفاده از اطلاعات جدول ۸-۲ تعریف می‌شود. هر فاصله، دو نقطه را بر نمودار تعریف می‌کند که با خط مستقیمی به هم وصل شده است. (این درونیابی خطی تنها امکان موجود نیست ولی ساده‌ترین امکان است.) توجه کنید که چهار فاصله به پنج زوج نقطه برای تعریف چهار پاره‌خط می‌انجامد. در مثال ۸-۲ می‌توان دید که

$$\hat{F}(x) = ۰, \qquad x < ۰$$

و

$$\hat{F}(x) = ۱, \qquad x > ۲$$

ولی از نقطه نظر تولید مقدار، این واقعیت اهمیتی ندارد. فرض بر این است که متغیر تصادفی، X، در مورد مدتهای تعمیر در رابطهٔ $X \geq ۰$ صدق می‌کند و هر مقدار، هر چه هم کوچک باشد میسر است. این فرض به نقطهٔ (۰, ۰) بر نمودار موجود در شکل ۸-۴ می‌انجامد. از سوی دیگر، تصور کنید معلوم است که همهٔ تعمیرها دستکم ۱۵ دقیقه طول می‌کشد و همواره، $X \geq ۰٫۲۵$ است. در این‌صورت، نقطهٔ (۰, ۰) باید به گونهٔ نشان داده شده در شکل ۸-۵ با (۰٫۲۵, ۰) جانشین



شکل ۸-۵ تولید مقادیر از تابع توزیع تجربی برای داده‌های مدت تعمیر ($X \geq ۰٫۲۵$).

www.matlabdl.com

شود. شکل ۸-۵ به منظور نمایش شیوهٔ تولید مورد استفاده قرار خواهد گرفت.

روش تبدیل معکوس مستقیماً برای تولید مقادیر مدت تعمیر، X، کاربردپذیر است. با به یاد آوردن تعبیر نموداری این روش، ابتدا یک عدد تصادفی، R_1، مثلاً $R_1 = ۰٫۸۳$ را تولید کنید و X_1 را از نمودار شکل ۸-۵ بخوانید. این مقدار را به‌صورت نمادین، می‌توان طبق رابطهٔ

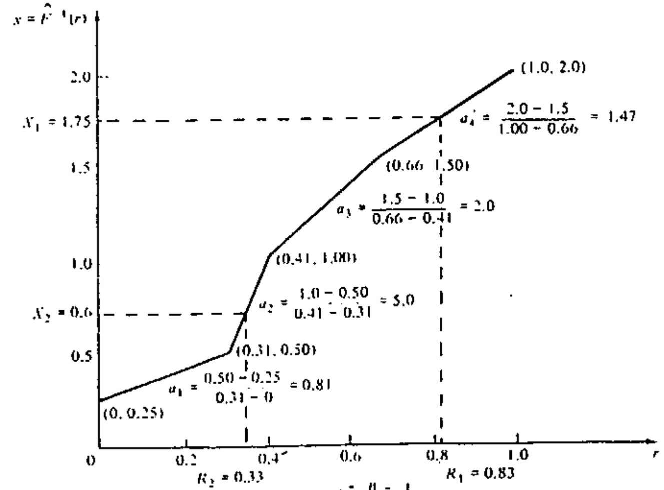
$$X_1 = F^{-1}(R_1)$$

بیان کرد، ولی از نظر جبری، چون مقدار R_1 بین ۰٫۶۶ و ۱٫۰۰ است، X_1 با درونیابی خطی مقداری بین ۱٫۵ و ۲٫۰ می‌شود. یعنی

$$X_1 = ۱٫۵ + \left[\frac{R_1 - ۰٫۶۶}{۱٫۰۰ - ۰٫۶۶}\right](۲٫۰ - ۱٫۵) = ۱٫۷۵ \qquad (۹\text{-}۸)$$

وقتی $R_1 = ۰٫۸۳$ است، توجه کنید که $(R_1 - ۰٫۶۶)/(۱٫۰۰ - ۰٫۶۶) = ۰٫۵$ می‌شود، که X_1 نصف فاصله بین ۱٫۵ و ۲٫۰ است زیرا R_1 در نیمه راه بین ۰٫۶۶ و ۱٫۰۰ است.

توجه کنید که به ازای همهٔ مقادیر R_1 در فاصلهٔ (۰٫۶۶، ۱٫۰۰)، به منظور محاسبهٔ X_1 به مقدار $a_4 = \frac{۲٫۰-۱٫۵}{۱٫۰۰-۰٫۶۶} = ۱٫۴۷$ نیاز داریم. مقدار a_4 شیب، $\frac{\Delta x}{\Delta r}$، تابع $x = \hat{F}^{-1}(r)$ است که صرفاً تصویر تابع $r = \hat{F}(x)$ از شکل ۸-۵ حول خط $r = x$ است. تابع معکوس نقش ورودی و خروجی را، همان‌طور که شکل ۸-۶ نشان می‌دهد معکوس می‌کند. شیبهای چهار پاره‌خط نیز



شکل ۸-۶ معکوس cdf مدتهای تعمیر.

جدول ۸-۳ فواصل و شیبها برای تولید مدتهای تعمیر، X.

i	ورودی r_i	خروجی x_i	شیب a_i
۱	۰	۰٫۲۵	۰٫۸۱
۲	۰٫۳۱	۰٫۵	۵٫۰
۳	۰٫۴۱	۱٫۰	۲٫۰
۴	۰٫۶۶	۱٫۵	۱٫۴۷
۵	۱٫۰۰	۲٫۰	—

در شکل ۸-۶ و در جدول ۸-۳ ارائه شده است که می‌توان به شرح زیر آنها را برای تولید مقادیر X مورد استفاده قرار داد:

گام ۱. R را تولید کنید.

گام ۲. فاصلهٔ i را که R در آن قرار دارد بیابید؛ یعنی i را چنان بیابید که $r_i \le R \le r_{i+1}$ باشد.

گام ۳. X را به‌صورت

$$X = x_i + a_i(R - r_i) \qquad (۱۰\text{-}۸)$$

محاسبه کنید.

برای داده‌های مدت مربوط به مدت تعمیر، نقاط انتهایی (r_i, x_i)، $i = ۱، \ldots، ۵$، و شیبهای a_i، $i = ۱, \cdots, ۴$ در جدول ۸-۳ ارائه شده است. اگر تولید تعداد بسیاری X لازم باشد، محاسبهٔ قبل از وقت a_iها و ذخیرهٔ آنها در جدول ۸-۳ برای استفادهٔ آتی سودمند خواهد بود. توجه داشته باشید که معادلهٔ (۸-۹) کاربردی از فرمول کلی درونیابی است که در معادلهٔ (۸-۱۰) ارائه شد. توضیحی دیگر می‌دهیم. تصور کنید $R_2 = ۰٫۳۳$ است. چون طبق جدول ۸-۳ رابطهٔ $r_2 = ۰٫۳۱ \le R_2 = ۰٫۳۳ < r_3 = ۰٫۴۱$ درست است، R_2 در فاصلهٔ $i = ۲$ قرار دارد و بنابراین

$$X_2 = x_2 + a_2(R_2 - r_2) = ۰٫۵ + ۵٫۰(۰٫۳۳ - ۰٫۳۱) = ۰٫۶$$

نقطهٔ $(R_2 = ۰٫۳۳، X_2 = ۰٫۶)$ را نیز در شکلهای ۸-۵ و ۸-۶ نشان داده‌ایم.

اینک، شکل ۸-۴ و داده‌های جدول ۸-۲ را دوباره در نظر بگیرید. داده‌ها محدود به دامنهٔ $۰ \le X \le ۲٫۰$ است. اما توزیع مبنا ممکن است دامنه‌ای وسیعتر داشته باشد. این، توجیه مهمی بر این ضرورت است که یک توزیع آماری نظری (مانند گاما یا ویبول) برای داده‌ها بیابیم، زیرا

این توزیعها دامنه‌ای وسیعتر، یعنی $۰ \leq X < \infty$ را فراهم می‌کند. به‌طور کلی، توصیه می‌کنیم که از cdf تجربی فقط به‌عنوان آخرین چاره استفاده شود.

علاوه بر این، توصیه می‌شود که نقاط مربوط به داده‌های منفرد گردآوری شود نه فقط داده‌های فاصله‌ای خلاصه به گونه‌ای که در جدول ۸-۲ مورد عمل قرار گرفت. اگر داده‌ها برحسب فراوانی در رده‌ها تلخیص شود، توصیه می‌شود که فواصل نسبتاً کوتاه مورد استفاده قرار گیرد، زیرا این کار به نمایش دقیقتر cdf مبنا می‌انجامد. مثلاً، برای داده‌ها مربوط به مدت تعمیر جدول ۸-۲، که در آن $n = ۱۰۰$ مشاهده بود، به‌جای چهار فاصلهٔ بسیار وسیع که عملاً به قصد نمایش مطلب انتخاب شد، با به کارگیری ۱۰ تا ۲۰ فاصله، که قطعاً تعداد چندان بزرگی نیست، می‌توانستیم براورد بسیار دقیقتری به‌دست آوریم.
■

اینک، مثالی را در نظر بگیرید که در آن همهٔ داده‌های خام در دست است. به منظور توضیح مطلب، تعداد مشاهده‌ها را کم می‌گیریم، اما می‌توانیم این روش را برای هر تعداد داده به کار ببریم.

■ **مثال ۸-۳**

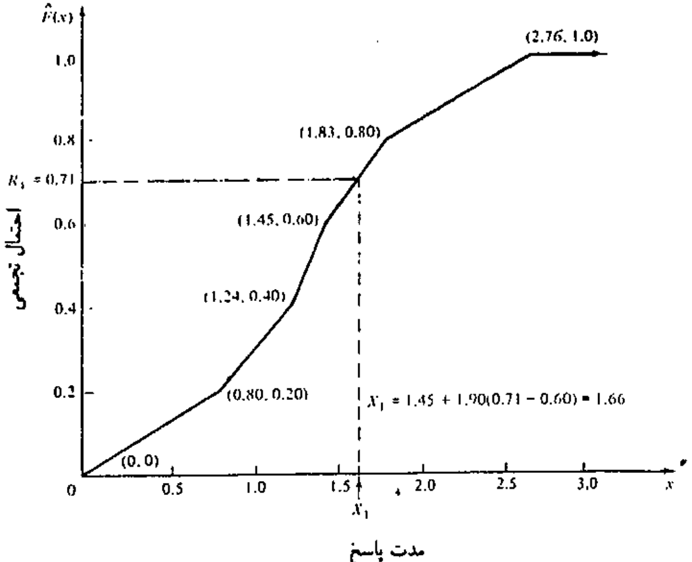
پنج مشاهده در مورد مدت پاسخ گروه آتش‌نشانان به درخواستهای کمک (برحسب دقیقه) گردآوری شده است تا در شبیه‌سازی مربوط به تحقیق دربارهٔ تأمین نیروی انسانی و خط‌مشیهای زمانبندی گروههای آتش‌نشان مورد استفاده قرار گیرد. داده‌ها عبارت است از

$$۱٫۲۴ \quad ۱٫۴۵ \quad ۰٫۸۰ \quad ۱٫۸۳ \quad ۲٫۷۶$$

پیش از گردآوری داده‌های بیشتر، علاقه‌مند به ایجاد یک مدل شبیه‌سازی مقدماتی هستیم که بر اساس این پنج مشاهده از توزیع مدت پاسخ استفاده می‌کند. بنابراین، به روشی برای تولید مقادیر تصادفی از توزیع مدت پاسخ نیاز داریم. نخست، فرض می‌کنیم که مدتهای پاسخ، X، دارای دامنهٔ $۰ \leq X \leq c$ است، که مقدار c نامعلوم است اما به وسیلهٔ $\hat{c} = \max\{X_i : i = ۱, \cdots, n\} = ۲٫۷۶$ براورد می‌شود، که $\{X_i, i = ۱, \cdots, n\}$ داده‌های خام و $n = ۵$ تعداد مشاهده‌هاست.

داده‌ها را مانند جدول ۸-۴ از کوچکترین به بزرگترین مرتب کنید و به هر فاصله احتمال $\frac{۱}{n} = \frac{۱}{۵}$ را نسبت دهید. cdf تجربی به‌دست آمده و $\hat{F}(x)$ در شکل ۸-۷ نمایش داده شده است و شیبهای، $\frac{\Delta x}{\Delta r}$، معکوس cdf، $x = \hat{F}^{-۱}(r)$، که برای تولید مدتهای پاسخ، X، مورد نیاز است در جدول ۸-۴ ارائه شده است. به‌عنوان مثال، اگر عدد تصادفی $R_۱ = ۰٫۷۱$ تولید شود، می‌بینیم که $R_۱$ در فاصلهٔ چهارم (بین $r_۴ = ۰٫۶۰$ و $r_۵ = ۰٫۸۰$) قرار می‌گیرد به‌طوری که طبق معادلهٔ (۸-۱۰) داریم

$$X_۱ = x_۴ + a_۴(R_۱ - r_۴) = ۱٫۴۵ + ۱٫۹۰(۰٫۷۱ - ۰٫۶۰) = ۱٫۶۵$$



شکل ۸-۷ cdf تجربی مدتهای پاسخ گروه آتش‌نشانان.

جدول ۸-۴ خلاصهٔ داده‌های مربوط به مدت پاسخ گروه آتش‌نشانان.

فاصله (دقیقه)	احتمال	احتمال تجمعی	شیب $\frac{\Delta x}{\Delta r}$
$۰٫۰ < x \leq ۰٫۸۰$	۰٫۲	۰٫۲	۴٫۰۰
$۰٫۸۰ < x \leq ۱٫۲۴$	۰٫۲	۰٫۴	۲٫۲۰
$۱٫۲۴ < x \leq ۱٫۴۵$	۰٫۲	۰٫۶	۱٫۰۵
$۱٫۴۵ < x \leq ۱٫۸۳$	۰٫۲	۰٫۸	۱٫۹۰
$۱٫۸۳ < x \leq ۲٫۷۶$	۰٫۲	۱٫۰	۴٫۶۵

برای نمای تصویری شیوهٔ تولید، خواننده را به شکل ۸-۷ ارجاع می‌دهیم. و همچنین به جدول ۸-۵ که اطلاعات جدول ۸-۴ را که در شکل قبلی به منظور تولید مورد استفاده قرار دادیم، تلخیص می‌کند.

جدول ۸-۵ معکوس cdf تجربی مدتهای پاسخ گروه آتش‌نشانان.

i	ورودی r_i	خروجی x_i	شیب a_i
۱	۰	۰	۴٫۰۰
۲	۰٫۲	۰٫۸	۲٫۲۰
۳	۰٫۴	۱٫۲۴	۱٫۰۵
۴	۰٫۶	۱٫۴۵	۱٫۹۰
۵	۰٫۸	۱٫۸۳	۴٫۶۵
۶	۱٫۰	۲٫۷۶	—

■

توضیح چند نکته لازم به نظر می‌رسد:

۱. به هنگام استفاده از این شکل روش تبدیل معکوس در مورد داده‌های تجربی، همچنانکه تعداد مشاهده‌ها، n، افزایش پیدا می‌کند، شکل کامپیوتری شیوه ناکاراتر می‌شود. هر شکل سیستماتیک کامپیوتری را اغلب طرح تولید جدولگرد می‌نامند زیرا به ازای مقدار مفروضی برای R، برنامهٔ کامپیوتر باید آرایه‌ای از ورودیها مانند جدول ۸-۵ را به منظور یافتن فاصلهٔ i که R در آن قرار دارد، و در مورد آن رابطهٔ

$$r_i \le R \le r_{i+۱}$$

صادق است، جستجو کند. هر چه تعداد فاصله‌ها بیشتر باشد، جستجو به‌طور متوسط وقت بیشتری می‌گیرد. تحلیلگر باید موازنهٔ بین دقت براورد cdf و کارایی محاسباتی را به هنگام نوشتن برنامهٔ شیوه مدنظر داشته باشد. اگر تعداد کثیری مشاهده در دست باشد، تحلیلگر می‌تواند مشاهده‌ها را در (مثلاً) ۲۰ تا ۵۰ فاصله گروهبندی کند و سپس شیوهٔ مثال ۸-۲ را به‌کار گیرد.

۲. در مثال ۸-۳ فرض کرده بودیم که مدتهای پاسخ، X، در رابطهٔ $۰ \le X \le ۲٫۷۶$ صدق می‌کند. این فرض به احتساب نقطه‌های $(x_۱, r_۱) = (۰, ۰)$ و $(x_۶, r_۶) = (۲٫۷۶, ۱٫۰۰)$ در شکل ۸-۷ و جدول ۸-۵ انجامید. اگر از قبل معلوم باشد که X در دامنه‌ای واقع است، مثلاً، اگر معلوم باشد که مدتهای پاسخ همواره بین ۱۵ ثانیه و ۳ دقیقه، یعنی

$$۰٫۲۵ \le X \le ۳٫۰$$

است، نقطه‌های $(x_۱, r_۱) = (۰٫۲۵, ۰)$، $(x_۶, r_۶) = (۲٫۷۶, ۰٫۸۳)$، و $(x_۷, r_۷) = (۳٫۰, ۱٫۰)$ برای براورد cdf تجربی مدتهای پاسخ در شکل ۸-۷ و جدول ۸-۵ به‌کار گرفته می‌شود. توجه کنید که به علت احتساب نقطهٔ جدید $(۳٫۰, ۱٫۰)$، اینک شش نقطه به‌جای پنج

نقطه وجود دارد و به هر فاصله احتمال $۰٫۱۶۷ = \frac{۱}{۶}$ نسبت داده می‌شود. تمرین ۱۲ به‌کارگیری این فرضهای اضافی را تشریح می‌کند.

۸-۱-۶ شیوه‌های جدولگرد در زمینهٔ تقریب تولید مقادیر نمایی و نرمال

در زبان شبیه‌سازی گسسته پیشامد GPSS V، امکان (مستقیم) محاسبهٔ لگاریتم، سینوس، کسینوس یا ریشه وجود ندارد و بدین ترتیب، شیوه‌های دقیق تولید که در زیر بخش ۸-۱-۱ برای توزیع نمایی و در بخش ۸-۲ برای توزیع نرمال شرح داده شده است، کاربردپذیر نیست. به منظور تولید مقادیر نمایی و نرمال، دستکم به‌طور تقریبی، تقریبهای پاره‌خطی استاندارد برای استفاده در GPSS ایجاد شده است. از لحاظ تصویری، این تقریبها ناظر به جانشین کردن cdf واقعی، یعنی

$$F(x) = ۱ - e^{-x}, \qquad ۰ \le x \qquad (۸\text{-}۱۱)$$

در مورد توزیع نمایی دارای میانگین یک و

$$\Phi(x) = \int_{-\infty}^{x} \sqrt{۱/۲\pi}\, e^{-\frac{۱}{۲}t^۲} dt \qquad (۸\text{-}۱۲)$$

در مورد توزیع نرمال استاندارد دارای میانگین صفر و واریانس یک، با دنباله‌ای از پاره‌خطها (یعنی تابعی پاره‌خطی)، و سپس استفاده از روش تبدیل معکوس در مورد این تقریب پاره‌خطی است. همان‌طور که در زیر بخش ۸-۱-۵ دیده می‌شود، هرگاه cdf پاره‌خطی باشد، روش تبدیل معکوس شیوه‌ای جدولگرد می‌شود. از لحاظ مفهومی، ایده همانند آن چیزی است که در شکل ۸-۴ به نمایش گذاشته شد، با این تفاوت که اینک cdf حقیقی، $F(x)$، معلوم [یا معادلهٔ (۸-۱۱) یا معادلهٔ (۸-۱۲)] است و نقاط انتهایی پاره‌خطهای مستقیم از داده‌ها براورد نمی‌شود، بلکه به گونه‌ای محاسبه می‌شود که پاره‌خطهای منحنی را به طریقی هر چه نزدیکتر تقریب می‌زند.

تقریبهای استاندارد برای تولید مقادیر تصادفی نمایی و نرمال، به ترتیب، در جدولهای ۸-۶ و ۸-۷ ارائه شده است. به منظور تولید مقداری مانند X از یکی از این توزیعها، ابتدا عددی تصادفی مانند R تولید کنید و سپس فاصله‌ای را بیابید که R در آن قرار دارد و سرانجام طبق فرمول درونیابی خطی داده شده در معادلهٔ (۸-۱۰)، و با اخذ مقدار مناسب r_i، x_i، و a_i از جدول ۸-۶ یا ۸-۷، X را محاسبه کنید.

■ **مثال ۸-۴**

با استفاده از روش دقیق معادلهٔ (۸-۲(الف)) و شیوهٔ تقریبی جدولگرد جدول ۸-۶، شش مقدار از یک توزیع نمایی با میانگین ۴۰ تولید کنید. برای روش تقریبی، ابتدا مقدار تصادفی نمایی، X،

www.matlabdl.com

جدول ۸-۶ جدول مربوط به تولید متغیر تصادفی با توزیع نمایی (۱ =میانگین).

ورودی r_i	خروجی x_i	شیب a_i	ورودی r_i	خروجی x_i	شیب a_i
۰	۰	۱٫۰۴	۰٫۹۰	۲٫۳۰	۱۱٫۰
۰٫۱	۰٫۱۰۴	۱٫۱۸	۰٫۹۲	۲٫۵۲	۱۴٫۵
۰٫۲	۰٫۲۲۲	۱٫۳۳	۰٫۹۴	۲٫۸۱	۱۸٫۰
۰٫۳	۰٫۳۵۵	۱٫۵۴	۰٫۹۵	۲٫۹۹	۲۱٫۰
۰٫۴	۰٫۵۰۹	۱٫۸۱	۰٫۹۶	۳٫۲۰	۳۰٫۰
۰٫۵	۰٫۶۹۰	۲٫۲۵	۰٫۹۷	۳٫۵۰	۴۰٫۰
۰٫۶	۰٫۹۱۵	۲٫۸۵	۰٫۹۸	۳٫۹۰	۷۰٫۰
۰٫۷	۱٫۲۰	۳٫۶۰	۰٫۹۹	۴٫۶۰	۱۴۰
۰٫۷۵	۱٫۳۸	۴٫۴۰	۰٫۹۹۵	۵٫۳۰	۳۰۰
۰٫۸۰	۱٫۶۰	۵٫۷۵	۰٫۹۹۸	۶٫۲۰	۸۰۰
۰٫۸۴	۱٫۸۳	۷٫۲۵	۰٫۹۹۹	۷٫۰	۲۳۳۳
۰٫۸۸	۲٫۱۲	۹٫۰۰	۰٫۹۹۹۷	۸٫۰	—

جدول ۸-۷ جدول مربوط به تولید متغیر تصادفی نرمال استاندارد.

ورودی r_i	خروجی x_i	شیب a_i	ورودی r_i	خروجی x_i	شیب a_i
۰	−۵٫۰	۳۳,۳۳۳	۰٫۵۷۹۲۶	۰٫۲	۲٫۶۳
۰٫۰۰۰۰۳	−۴٫۰	۷۵۶	۰٫۶۵۵۴۲	۰٫۴	۲٫۸۴
۰٫۰۰۱۳۵	−۳٫۰	۲۰۶	۰٫۷۲۵۷۵	۰٫۶	۳٫۲۱
۰٫۰۰۶۲۱	−۲٫۵	۳۰٫۲	۰٫۷۸۸۱۴	۰٫۸	۳٫۷۶
۰٫۰۲۲۷۵	−۲٫۰	۱۱٫۳	۰٫۸۴۱۳۴	۱٫۰	۴٫۵۹
۰٫۰۶۶۸۱	−۱٫۵	۶٫۲۲	۰٫۸۸۴۹۳	۱٫۲	۶٫۲۲
۰٫۱۱۵۰۷	−۱٫۲	۴٫۵۹	۰٫۹۲۳۱۹	۱٫۵	۱۱٫۳
۰٫۱۵۸۶۶	−۱٫۰	۳٫۷۶	۰٫۹۷۷۲۵	۲٫۰	۳۰٫۲
۰٫۲۱۱۸۶	−۰٫۸	۳٫۲۱	۰٫۹۹۳۷۹	۲٫۵	۲۰۶
۰٫۲۷۴۲۵	−۰٫۶	۲٫۸۴	۰٫۹۹۸۶۵	۳٫۰	۷۵۶
۰٫۳۴۴۵۸	−۰٫۴	۲٫۶۳	۰٫۹۹۹۹۷	۴٫۰	۳۳,۳۳۳
۰٫۴۲۰۷۴	−۰٫۲	۲٫۵۲	۱٫۰	۵٫۰	—
۰٫۵۰۰۰۰	۰	۲٫۵۲			

با میانگین یک تولید کنید و سپس رابطهٔ

$$Y = \beta X \qquad (۱۳-۸)$$

را به کار ببرید و مقدار نمایی Y با میانگین دلخواه β را محاسبه کنید. معادلهٔ (۸-۱۳) به‌طور کلی برای سایر توزیعها برقرار نیست. ضرب کردن متغیری تصادفی با میانگین یک در مقداری مانند β، متغیر تصادفی تازه‌ای با میانگین β می‌دهد ولی معمولاً شکل توزیع نیز دچار تغییر می‌شود. معادلهٔ (۸-۱۳) در مورد خانوادهٔ توزیعهای گاما که توزیعهای نمایی و ارلنگ را دربر دارد قابل استفاده است.

تصور کنید $R_1 = ۰٫۱۶۳۶$ است؛ پس طبق روش دقیق ارائه شده در معادلهٔ (۸-۲(الف)) داریم

$$Y_1 = -\beta \ell n(1 - R_1) = -40 \ell n(1 - 0.1636) = 7.15$$

با استفاده از روش تقریبی، ابتدا توجه کنید که $R_1 = ۰٫۱۶۳۶$ در فاصلهٔ $i = ۲$ قرار دارد، یعنی $r_2 = 0.1 \le R_1 = 0.1636 < r_3 = 0.2$، بدین ترتیب با استفاده از جدول ۸-۶ داریم

$$X_1 = x_2 + a_2(R_1 - r_2) = 0.104 + 1.18(0.1636 - 0.1) = 0.179$$

و با استفاده از معادلهٔ (۸-۱۳) به ازای $\beta = ۴۰$ داریم

$$Y_1 = 40 X_1 = 7.16$$

این مقدار به‌اضافهٔ پنج مقدار دیگر در جدول ۸-۸ نشان داده شده است.

پنج مقدار تصادفی یکنواخت اول از جدول پ-۱ برگزیده شد، ولی آخرین مقدار، $R_6 = ۰٫۰۵$، به‌طور اختیاری برگزیده شد تا نشان داده شود که در این تقریب پاره‌خطی خاص، خطای نسبی

جدول ۸-۸ تولید مقادیر تصادفی نمایی با روشهای دقیق و تقریبی.

i	R_i	Y_i (دقیق)	Y_i (تقریبی)	درصد خطا
۱	۰٫۱۶۳۶	۷٫۱۵	۷٫۱۶	۰٫۱۴
۲	۰٫۹۰۴۰	۹۳٫۷۴	۹۳٫۷۶	۰٫۰۲
۳	۰٫۱۸۷۱	۸٫۲۹	۸٫۲۷	۰٫۲۴
۴	۰٫۷۸۲۴	۶۱٫۰۰	۶۰٫۹۰	۰٫۱۶
۵	۰٫۵۹۰۵	۳۵٫۷۱	۳۵٫۷۵	۰٫۱۱
۶	۰٫۰۵۰۰	۲٫۰۵	۲٫۰۸	۱٫۳۸

می‌تواند به بزرگی ۱٫۴ درصد باشد. انتقاد دیگر بر تقریب این است که دامنهٔ متغیر تولیدشده محدود به $۰ \le X \le ۸$، یعنی بزرگترین مقدار x_i در جدول ۸-۶ است، حال آنکه دامنهٔ هر نمایی تمام مقادیر غیر منفی، $۰ \le X < \infty$ است. برای یک متغیر تصادفی نمایی، X، با میانگین یک، احتمال بزرگتر شدن X از ۸، یعنی

$$\Pr(X > ۸) = e^{-۸} = ۰٫۰۰۰۳۴$$

ممکن است بسیار کوچک به نظر برسد و در واقع، چنین مقادیری به‌ندرت (حدود ۳۴ مقدار در هر ۱۰۰۰۰۰ مقدار تولید شده) طبق روش دقیق ارائه شده در معادلهٔ (۸-۲) تولید می‌شود، ولی اگر تحت شرایطی تعداد بسیاری از مقادیر تولید شود و اگر مقداری بزرگ برای X تأثیری بسیار چشمگیر بر سیستم بگذارد، محدودیتهای تقریب استاندارد جدول ۸-۶ اهمیت بیشتری پیدا می‌کند. انتقادهای همانندی نسبت به تقریب cdf نرمال در جدول ۸-۷ صورت گرفته است. به‌طور کلی، توصیه می‌شود در صورت امکان از روشی دقیق، مانند آنهایی که در این فصل مورد بحث قرار گرفت، استفاده شود. (GPSS V تنها زبان اصلی شبیه‌سازی گسسته پیشامد است که منحصراً بر روش جدولگرد و تقریبهای عددی متکی است. اما، GPSS V از توانایی فراخوانی برنامه‌های FORTRAN و PL/1 از طریق بلوک HELP برخوردار است که این توانایی می‌تواند به منظور استفاده از روشهای دقیق تولید مقادیر از توزیعهای آماری استاندارد به کار گرفته شود.) ■

استفاده از جدول ۸-۷ به منظور تولید مقادیر تقریبی نرمال را به تمرین ۲۴ وامی‌گذاریم. فصل ۹ گوردون [۱۹۷۵] را برای بررسی بیشتر شیوهٔ جدولگرد در GPSS V و به ویژه روش معمول GPSS برای تولید مقادیر نمایی و نرمال توصیه می‌کنیم.

۸-۱-۷ توزیعهای گسسته

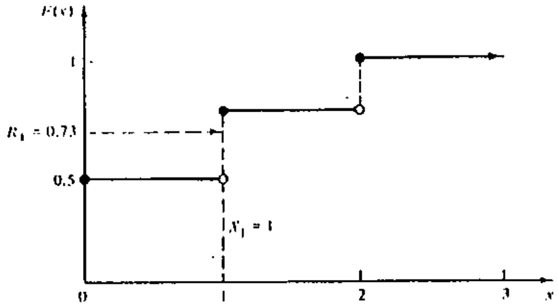
مقدار تصادفی برای همهٔ توزیعهای گسسته با استفاده از روش تبدیل معکوس، یا به‌صورت عددی از طریق شیوهٔ جدولگرد، یا در بعضی موارد به‌صورت جبری که طرح نهایی تولید در قالب فرمولی ارائه می‌شود قابل تولید است. گاهی سایر روشها برای توزیعهای مشخصی مورد استفاده قرار می‌گیرد، مثل روش پیچش برای توزیع دو جمله‌ای. برخی از این روشها را در بخشهای بعد مورد بحث قرار می‌دهیم. این زیربخش مثالهایی شامل توزیعهای تجربی و دو توزیع گسستهٔ استاندارد، یعنی یکنواخت (گسسته) و هندسی را عرضه می‌کند.

■ مثال ۸-۵ یک توزیع گسستهٔ تجربی

در پایان روز، تعداد محموله‌های موجود بر سکوی بارگیری یک شرکت (که محصول اصلی آن ابزار

جدول ۸-۹ توزیع تعداد محموله‌ها، X.

x	$p(x)$	$F(x)$
۰	۰٫۵۰	۰٫۵۰
۱	۰٫۳۰	۰٫۸۰
۲	۰٫۲۰	۱٫۰۰



شکل ۸-۸ cdf تعداد محموله‌ها، X.

بسیار بزرگی است)، صفر، یک، یا دو با فراوانی نسبی مشهود، به ترتیب، ۰٫۵۰، ۰٫۳۰ و ۰٫۲۰ است. از مشاوران داخلی خواسته شده است تا به منظور بهبود کارایی عملیات بارگیری و حمل مدلی ایجاد کنند؛ آنها به‌عنوان بخشی از مدل نیاز دارند که بتوانند مقادیر X را برای معرفی تعداد محموله‌ها بر سکوی بارگیری در پایان روز تولید کنند. مشاوران تصمیم می‌گیرند که X را به‌صورت متغیر تصادفی گسسته‌ای با توزیع ارائه شده در جدول ۸-۹ و نشان داده شده در شکل ۸-۸ مدلسازی کنند. تابع جرم احتمال، $p(x)$، به‌صورت

$$p(۰) = P(X = ۰) = ۰٫۵۰$$

$$p(۱) = P(X = ۱) = ۰٫۳۰$$

$$p(۲) = P(X = ۲) = ۰٫۲۰$$

www.matlabdl.com

و cdf، $F(x) = P(X \le x)$ به‌صورت

$$F(x) = \begin{cases} ۰ & x < ۰ \\ ۰٫۵ & ۰ \le x < ۱ \\ ۰٫۸ & ۱ \le x < ۲ \\ ۱٫۰ & ۲ \le x \end{cases}$$

ارائه می‌شود. به یاد دارید که cdf هر متغیر تصادفی گسسته همواره از پاره‌خطهای افقی با جهشهایی به اندازهٔ $p(x)$ در نقاطی که متغیر تصادفی مقدار می‌پذیرد تشکیل می‌شود. مثلاً، در شکل ۸-۸، جهشی به اندازه $p(x) = ۰٫۵$ در $x = ۰$، جهشی به اندازهٔ $p(x) = ۰٫۳$ در $x = ۱$ و جهشی به اندازهٔ $p(x) = ۰٫۲$ در $x = ۲$ وجود دارد.

به منظور تولید مقادیر تصادفی گسسته، روش تبدیل معکوس، با شیوهٔ جدولگرد جایگزین می‌شود، ولی برخلاف مورد متغیرهای پیوسته، نیازی به درونیابی نیست. برای تشریح شیوه، تصور کنید $R_1 = ۰٫۷۳$ تولید می‌شود. از لحاظ تصویری، به گونهٔ نشان داده شده در شکل ۸-۸، ابتدا $R_1 = ۰٫۷۳$ را روی محور عمودی مکانیابی کنید، سپس پاره خطی افقی رسم کنید تا به یکی از «جهشهای» cdf برخورد کند و سپس عمودی بر محور افقی فرود آورید تا مقدار تولیدشده را تعیین کند. در اینجا، $R_1 = ۰٫۷۳$ به $X_1 = ۱$ تبدیل می‌شود. این شیوه شبیه به شیوهٔ مورد استفاده برای توزیعهای پیوسته در زیر بخش ۸-۱-۵ و به نمایش گذاشته شده در شکل ۸-۵ است. با این تفاوت که گام نهایی درونیابی خطی حذف شده است.

شیوهٔ جدولگرد از طریق ایجاد جدولی همانند جدول ۸-۱۰ تسهیل می‌شود. وقتی $R_1 = ۰٫۷۳$ تولید می‌شود، ابتدا فاصله‌ای را بیابید که R_1 در آن قرار دارد. به‌طور کلی، به ازای $R = R_1$، اگر

$$F(x_{i-1}) = r_{i-1} < R \le r_i = F(x_i) \qquad (۸\text{-}۱۴)$$

باشد، X_1 را مساوی با x_i قرار دهید. در اینجا $r_0 = ۰$، $x_0 = -\infty$ است و $x_1, x_2, \cdots, x_n$

جدول ۸-۱۰ جدول برای تولید مقدار گسستهٔ x.

i	ورودی r_i	خروجی x_i
۱	٫۵۰	۰
۲	۰٫۸۰	۱
۳	۱٫۰۰	۲

مقادیر ممکن متغیر تصادفی و $r_k = p(x_1) + \cdots + p(x_k), k = ۱, ۲, \cdots, n$ است. (توجه کنید که در همهٔ موارد $r_n = ۱٫۰$ است.)

چون $r_1 = ۰٫۵ < R_1 = ۰٫۷۳ \le r_2 = ۰٫۸$ است، X_1 را مساوی با $x_2 = ۱$ قرار دهید. طرح تولید به‌صورت زیر تلخیص می‌شود

$$X = \begin{cases} ۰, & R \le ۰٫۵ \\ ۱, & ۰٫۵ < R \le ۰٫۸ \\ ۲, & ۰٫۸ < R \le ۱٫۰ \end{cases}$$

■

مثال ۸-۵ شیوهٔ جدولگرد را نمایش می‌دهد، حال آنکه مثال بعد رهیافتی جبری را نشان می‌دهد که برای برخی از توزیعها قابل استفاده است.

■ مثال ۸-۶ یک توزیع یکنواخت گسسته

توزیع یکنواخت گسسته روی نقاط $\{۱, ۲, \cdots, k\}$ را با pmf و cdf ارائه شده در زیر در نظر بگیرید:

$$p(x) = \frac{۱}{k}, \quad x = ۱, ۲, \cdots, k$$

$$F(x) = \begin{cases} ۰, & x < ۱ \\ \frac{۱}{k}, & ۱ \le x < ۲ \\ \frac{۲}{k}, & ۲ \le x < ۳ \\ \vdots & \vdots \\ \frac{k-۱}{k}, & k-۱ \le x < k \\ ۱ & k \le x \end{cases}$$

بگذارید $x_i = i$ و به ازای $i = ۱, ۲, \cdots, k$، $r_i = p(۱) + \cdots + p(x_i) = F(x_i) = i/k$ باشد. در این صورت، با استفاده از نامساوی (۸-۱۴) می‌توان دید که اگر اعداد تصادفی تولیدشده در رابطهٔ

$$r_{i-1} = \frac{i-۱}{k} < R \le r_i = \frac{i}{k} \qquad (۸\text{-}۱۵)$$

صدق کند، X با نوشتن تساوی $X = i$ تولید می‌شود. اینک، می‌توان نامساوی (۸-۱۵) را

به ازای i حل کرد:

$$i - ۱ < Rk \le i$$

$$Rk \le i < Rk + ۱ \qquad (۸\text{-}۱۶)$$

بگذارید $\lceil y \rceil$ معرف کوچکترین عدد صحیح بزرگتر یا مساوی با y باشد. مثلاً $\lceil ۷٫۸۲ \rceil = ۸$، $\lceil ۵٫۱۳ \rceil = ۶$ و $\lceil -۱٫۳۲ \rceil = -۱$ است. $\lceil y \rceil$ به ازای $y \ge ۰$ تابعی است که به بالا گرد می‌شود. این نمادگذاری و نامساوی (۸-۱۶) فرمولی برای تولید X عرضه می‌کند، یعنی

$$X = \lceil Rk \rceil \qquad (۸\text{-}۱۷)$$

مثلاً، تولید مقدار تصادفی X با توزیع یکنواخت روی نقاط $\{۱, ۲, \cdots, ۱۰\}$ را در نظر بگیرید. مقدار X ممکن است معرف تعداد بالتهایی باشد که باید درکامیون بار شود. با استفاده از جدول پ-۱ به‌عنوان منبع اعداد تصادفی، R، و معادلهٔ (۸-۱۷) به ازای $k = ۱۰$، داریم

$$R_۱ = ۰٫۷۸ \qquad X_۱ = \lceil ۷٫۸ \rceil = ۸$$

$$R_۲ = ۰٫۰۳ \qquad X_۲ = \lceil ۰٫۳ \rceil = ۱$$

$$R_۳ = ۰٫۲۳ \qquad X_۳ = \lceil ۲٫۳ \rceil = ۳$$

$$R_۴ = ۰٫۹۷ \qquad X_۴ = \lceil ۹٫۷ \rceil = ۱۰$$

■

شیوهٔ مورد بحث در اینجا را می‌توان به منظور تولید مقداری تصادفی از توزیع یکنواخت گسسته در هر دامنهٔ متشکل از اعداد صحیح متوالی اصلاح کرد. در تمرین ۱۳ از دانشجو می‌خواهیم که شیوه‌ای برای این مورد طراحی کند.

■ مثال ۸-۷

توزیع گسستهٔ دارای pmf ارائه شده در زیر را در نظر بگیرید

$$p(x) = \frac{۲x}{k(k + ۱)}, \qquad x = ۱, ۲, \cdots, k$$

(این مثال برگرفته از اشمید و تیلور [۱۹۷۰] است.) به ازای مقادیر صحیح x در دامنهٔ

$\{۱, ۲, \cdots, k\}$، cdf به‌صورت زیر است

$$F(x) = \sum_{i=۱}^{x} \frac{۲i}{k(k + ۱)} = \frac{۲}{k(k + ۱)} \sum_{i=۱}^{x} i = \frac{۲}{k(k + ۱)} \frac{x(x + ۱)}{۲} = \frac{x(x + ۱)}{k(k + ۱)}$$

R را تولید و از نامساوی (۸-۱۴) استفاده کنید تا هرگاه

$$F(x - ۱) = \frac{(x - ۱)x}{k(k + ۱)} < R \le \frac{x(x + ۱)}{k(k + ۱)} = F(x)$$

یا هرگاه

$$(x - ۱)x < k(k + ۱)R \le x(x + ۱)$$

باشد نتیجه بگیرید که $X = x$ است. به منظور حل نامساوی اخیر برحسب R، ابتدا مقدار x صادق در رابطهٔ

$$(x - ۱)x = k(k + ۱)R$$

یا

$$x^۲ - x - k(k + ۱)R = ۰$$

را پیدا کنید. سپس، با گرد کردن به بالا، متوجه می‌شوید جواب $X = \lceil x - ۱ \rceil$ است. طبق فرمول معادلهٔ درجهٔ دوم، یعنی

$$x = \frac{-b \pm \sqrt{b^۲ - ۴ac}}{۲a}$$

به ازای $a = ۱, b = -۱$ و $c = -k(k + ۱)R$، جواب معادلهٔ درجهٔ دو عبارت است از

$$x = \frac{۱ \pm \sqrt{۱ + ۴k(k + ۱)R}}{۲} \qquad (۸\text{-}۱۸)$$

ریشهٔ صحیح برای استفاده، ریشهٔ مثبت در معادلهٔ (۸-۱۸) است (چرا؟)، پس X از رابطهٔ

$$X = \left\lceil \frac{۱ + \sqrt{۱ + ۴k(k + ۱)R}}{۲} - ۱ \right\rceil \qquad (۸\text{-}۱۹)$$

تولید می‌شود. در تمرین ۱۴ از دانشجو می‌خواهیم که چند مقدار از این توزیع تولید کند.
■

■ **مثال ۸-۸ توزیع هندسی**

توزیع هندسی با pmf

$$p(x) = p(1-p)^x, \quad x = 0, 1, 2, \cdots$$

را در نظر بگیرید که $0 < p < 1$ است. cdf این توزیع به ازای $x = 0, 1, 2, \cdots$ عبارت است از

$$F(x) = \sum_{j=0}^{x} p(1-p)^j = \frac{p\{1-(1-p)^{x+1}\}}{1-(1-p)} = 1-(1-p)^{x+1}$$

از روش تبدیل معکوس [یعنی، نامساوی (۱۴.۸)] استفاده کنید و به یاد داشته باشید که متغیر تصادفی هندسی، X، مقدار x را می‌گیرد هر گاه که

$$F(x-1) = 1-(1-p)^x < R \le 1-(1-p)^{x+1} = F(x) \qquad (۸-۲۰)$$

باشد، که R یک عدد تصادفی تولید شده است که $0 < R < 1$ فرض می‌شود. جواب نامساوی (۸-۲۰) به ازای x به شرح زیر یافته می‌شود

$$(1-p)^{x+1} \le 1-R < (1-p)^x$$
$$(x+1)\ell n(1-p) \le \ell n(1-R) < x\ell n(1-p)$$

اما از رابطهٔ $0 < 1-p < 1$، رابطهٔ $\ell n(1-p) < 0$ نتیجه می‌شود، که داریم

$$\frac{\ell n(1-R)}{\ell n(1-p)} - 1 \le x < \frac{\ell n(1-R)}{\ell n(1-p)} \qquad (۸-۲۱)$$

بنابراین، به ازای آن مقدار صحیح x که در نامساوی (۸-۲۱) صدق کند، داریم $X = x$، یا به اختصار، با استفاده از تابع گرد کردن به بالا $\lceil \cdot \rceil$، داریم

$$X = \left\lceil \frac{\ell n(1-R)}{\ell n(1-p)} - 1 \right\rceil \qquad (۸-۲۲)$$

چون p پارامتری ثابت است، فرض کنید $\beta = -1/\ell n(1-p)$ باشد، پس $\beta > 0$ و به موجب معادلهٔ (۸-۲۲)، $X = \lceil -\beta\ell n(1-R) - 1\rceil$ است. به موجب معادلهٔ (۸-۱)، $-\beta\ell n(1-R)$ مقداری تصادفی با توزیع نمایی و میانگین β است، که یک راه تولید مقدار تصادفی هندسی با پارامتر p، تولید یک مقدار نمایی با پارامتر $\beta^{-1} = -\ell n(1-p)$ (با هر روشی)، کم کردن یک از آن و گرد کردن نتیجه به بالاست.

گهگاه، مقداری هندسی، X، مورد نیاز است که مقادیر $\{q, q+1, q+2, \cdots\}$ را با تابع احتمال $p(x) = p(1-p)^{x-q}$ به ازای $x = q, q+1, \cdots$ می‌گیرد. این مقدار X را می‌توان با استفاده از معادلهٔ (۸-۲۲)، از رابطهٔ

$$X = q + \left\lceil \frac{\ell n(1-R)}{\ell n(1-p)} - 1 \right\rceil \qquad (۸-۲۳)$$

تولید کرد. یکی از عادیترین موارد، $q = 1$ است. ■

■ **مثال ۸-۹**

از توزیع هندسی با دامنهٔ $\{X \ge 1\}$ و میانگین ۲، سه مقدار تولید کنید. یک چنین توزیع هندسی دارای تابع احتمال $p(x) = p(1-p)^{x-1}$ به ازای $x = 1, 2, \cdots$ و میانگین $2 = \frac{1}{p}$ یا $p = \frac{1}{2}$ است. پس، به ازای $q = 1$، $p = \frac{1}{2}$، و $1/\ell n(1-p) \doteq -1.443$، X را می‌توان طبق معادلهٔ (۸-۲۳) تولید کرد. استفاده از جدول پ-۱، $R_1 = 0.932$، $R_2 = 0.105$ و $R_3 = 0.687$ به نتیجهٔ زیر می‌رسد

$$X_1 = 1 + \lceil -1.443\ell n(1-0.932) - 1\rceil = 1 + \lceil 3.878 - 1\rceil = 4$$
$$X_2 = 1 + \lceil -1.443\ell n(1-0.105) - 1\rceil = 1$$
$$X_3 = 1 + \lceil -1.443\ell n(1-0.687 - 1\rceil = 2$$

تمرین ۱۵ به کاربرد توزیع هندسی مربوط می‌شود. ■

۸-۲ تبدیل مستقیم در مورد توزیع نرمال

روشهای بسیاری برای تولید مقادیر تصادفی با توزیع نرمال به وجود آمده است. اما، روش تبدیل معکوس کاربردپذیر نیست، زیرا نمی‌توان معکوس cdf را از طریق تحلیلی محاسبه کرد. (روش تبدیل معکوس در زیر بخش ۸-۱-۶ در مورد تقریب پاره‌خطی cdf نرمال به‌کار گرفته شد.) cdf نرمال استاندارد به صورت

$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}t^2} dt, \qquad -\infty < x < \infty$$

است. به منظور استفاده از روش تبدیل معکوس، لازم است بتوان (به شکل بسته) معادلهٔ $\Phi(X) = R$ را برحسب R حل کرد (آزمایش کنید-این امری غیر ممکن است!). روشهای دیگری که مورد استفاده قرار گرفته است، شامل روش پیچش (بخش ۸-۳) و روشهای رد و قبول (بخش ۸-۴) است. این بخش تبدیلی مستقیم و گیرا را شرح می‌دهد که یک زوج مقدار تصادفی نرمال با میانگین صفر و

واریانس یک تولید می‌کند. روش از باکس و مولر [۱۹۵۸] است. هر چند این روش به اثربخشی بسیاری از روشهای جدید نیست، ولی برنامه‌نویسی آن در زبانی علمی مانند FORTRAN آسان است.

دو متغیر تصادفی نرمال استاندارد، Z_1 و Z_2، را در نظر بگیرید که به گونهٔ نشان داده شده در شکل ۸-۹ به‌صورت یک نقطه در صفحه رسم و به‌صورت

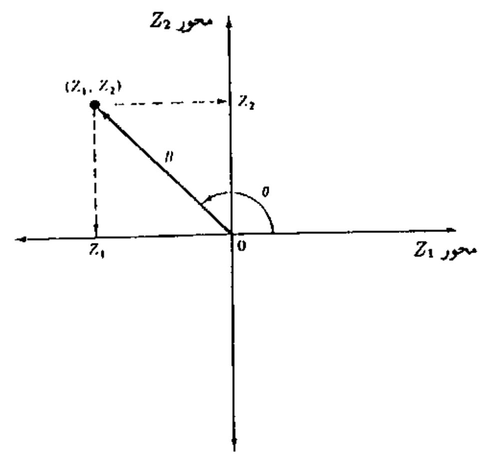
$$Z_1 = B\cos\theta$$

$$Z_2 = B\sin\theta \qquad (۲۴-۸)$$

با مختصات قطبی نمایش داده شده است. معلوم است که $B^2 = Z_1^2 + Z_2^2$ توزیع مربع‌کای با دو درجهٔ آزادی دارد، که همان توزیع نمایی با میانگین ۲ است. بنابراین، شعاع B را با استفاده از معادلهٔ (۸-۲(ب)) می‌توان تولید کرد

$$B = (-2\ell n R)^{1/2} \qquad (۲۵-۸)$$

بر اساس تقارن توزیع نرمال، این فرض که زاویهٔ θ بین صفر و 2π به‌طور یکنواخت توزیع می‌شود، منطقی به نظر می‌رسد که واقعاً چنین نیز هست. علاوه بر این، شعاع B و زاویهٔ



شکل ۸-۹ نمایش قطبی یک زوج متغیر نرمال استاندارد.

θ مستقل نیز هستند. با تلفیق معادله‌های (۸-۲۴) و (۸-۲۵)، روش مستقیمی برای تولید دو مقدار مستقل نرمال استاندارد، Z_1 و Z_2، از دو عدد تصادفی مستقل R_1 و R_2 به‌دست می‌آید:

$$Z_1 = (-2\ell n R_1)^{1/2}\cos(2\pi R_2)$$

$$Z_2 = (-2\ell n R_1)^{1/2}\sin(2\pi R_2) \qquad (۲۶-۸)$$

مثالی از کاربرد معادله‌های (۸-۲۶) در زیر بخش ۸-۳-۲ ارائه می‌شود.

۸-۳ روش پیچش

توزیع احتمال جمع دو یا چند متغیر تصادفی مستقل را پیچش توزیعهای متغیرهای اصلی می‌نامند. به این ترتیب، روش پیچش به افزودن دو یا چند متغیر تصادفی به منظور به‌دست آوردن متغیر تصادفی تازه‌ای با توزیع موردنظر اشاره دارد. با استفاده از این روش می‌توان مقادیر ارلنگ، مقادیری با توزیع تقریباً نرمال و مقادیر دو جمله‌ای را به‌دست آورد. آنچه اهمیت دارد، cdf متغیر تصادفی موردنظر نیست، بلکه رابطهٔ آن با سایر مقادیری است که تولید آنها آسان است.

۸-۳-۱ توزیع ارلنگ

به گونه‌ای که در بخش ۴-۴ بحث شد، می‌توان نشان داد که هر متغیر تصادفی ارلنگ X با پارامترهای (K,θ) جمع K متغیر تصادفی مستقل نمایی، $X_i (i = 1,\cdots,K)$، هر یک با میانگین $\frac{1}{K\theta}$ است، یعنی،

$$X = \sum_{i=1}^{K} X_i$$

چون می‌توان هر X_i را طبق معادلهٔ (۸-۲ (ب)) به ازای $\frac{1}{\lambda} = \frac{1}{K\theta}$ تولید کرد، مقدار تصادفی ارلنگ را می‌توان به‌صورت زیر تولید کرد

$$X = \sum_{i=1}^{K} -\frac{1}{K\theta}\ell n R_i = -\frac{1}{K\theta}\ell n\left(\prod_{i=1}^{K} R_i\right) \qquad (۲۷-۸)$$

در معادلهٔ (۸-۲۷)، Π معرف حاصلضرب است. از لحاظ محاسباتی کاراتر است که ابتدا تمام اعداد تصادفی را در هم ضرب کنیم و سپس فقط یک لگاریتم بگیریم.

■ **مثال ۸-۱۰**

کامیونهایی به‌طور کاملاً تصادفی به انبار وسیعی وارد می‌شوند؛ ورود به‌صورت یک فرایند پواسون با آهنگ $\lambda = 10$ کامیون در ساعت مدلسازی می‌شود. نگهبان درب ورود، کامیونها را به‌طور متناوب به سکوهای شمالی و جنوبی می‌فرستد. تحلیلگری به منظور بررسی فرایند بارگیری و تخلیه در سکوهای جنوبی، مدلی ایجاد کرده است و به مدل فرایند ورود فقط به سکوهای جنوبی نیاز دارد. هر مدت بین ورودهای دو کامیون متوالی در سکوهای جنوبی، X، برابر با جمع دو مدت بین دو ورود به انبار است و به این ترتیب، جمع دو متغیر تصادفی نمایی هر یک با میانگین ۰٫۱ ساعت، یا ۶ دقیقه است. پس X توزیع ارلنگ با $K = 2$ و میانگین $0.2 = \frac{2}{\lambda} = \frac{1}{\theta}$ ساعت دارد. به منظور تولید مقدار X، ابتدا $K = 2$ عدد تصادفی، مثلاً $R_1 = 0.937$ و $R_2 = 0.217$ را از جدول پ-۱ به‌دست آورید. سپس به موجب معادلهٔ (۸-۲۷) چنین بنویسید

$$X = -0.1\ln[0.937(0.217)] = 0.159 \text{ ساعت } = 9.56 \text{ دقیقه}$$

به‌طور کلی، از معادلهٔ (۸-۲۷) چنین بر می‌آید که برای تولید هر مقدار ارلنگ به K عدد تصادفی نیاز داریم. اگر K بزرگ باشد، تولید مقادیر ارلنگ با روشهای دیگر، از قبیل یکی از روشهای فراوان رد و قبول برای توزیع گاما که فیشمن [۱۹۷۸] آنها را عرضه کرده، کاراتر است. ■

۸-۳-۲ تولید مقادیر تقریباً نرمال

قضیهٔ حد مرکزی چنین می‌گوید که جمع n متغیر تصادفی مستقل و هم توزیع X_1، X_2، $\cdots$، X_n، هر یک با میانگین μ_X و واریانس محدود σ_X^2، تقریباً توزیع نرمال با میانگین $n\mu_X$ و واریانس $n\sigma_X^2$ دارد. با به‌کارگیری این قضیه در مورد متغیرهای تصادفی یکنواخت در فاصلهٔ (۰،۱) که میانگین $\mu = 0.5$ و واریانس $\sigma^2 = \frac{1}{12}$ دارد، نتیجه می‌گیریم که

$$Z = \frac{\sum_{i=1}^{n} R_i - 0.5n}{(n/12)^{1/2}} \qquad (۸-۲۸)$$

تقریباً توزیع نرمال با میانگین صفر و واریانس یک دارد. هر چه مقدار n بزرگتر شود، تقریب مناسبتر می‌شود، اما بسیاری از نویسندگان کتابهای درسی عنوان می‌کنند که $n = 12$ به‌عنوان تقریبی مناسب برای نرمال بودن کافی است. علاوه بر این، استفاده از $n = 12$ برای برنامهٔ کامپیوتری کاراترین شق ممکن است زیرا در آن از گرفتن ریشه دوم و یک عمل تقسیم به گونه‌ای که می‌بینیم

اجتناب می‌شود. با قرار دادن $n = 12$ در معادلهٔ (۸-۲۸)، طرح تولید

$$Z = \sum_{i=1}^{12} R_i - 6 \qquad (۸-۲۹)$$

برای یک متغیر تصادفی تقریباً نرمال با میانگین صفر و واریانس یک به‌دست می‌آید. اگر تولید مقدار نرمالی مانند Y با میانگین μ_Y و واریانس σ_Y^2 مدنظر باشد، ابتدا Z را طبق معادلهٔ (۸-۲۹) تولید و سپس از تبدیل

$$Y = \mu_Y + \sigma_Y Z \qquad (۸-۳۰)$$

استفاده می‌کنیم. به تفاوت معادلهٔ (۸-۳۰) برای تبدیل یک مقدار نرمال استاندارد به مقدار نرمال موردنظر، با معادلهٔ (۸-۱۳) برای تبدیل مقدار نمایی استاندارد به مقدار نمایی مورد نظر توجه کنید. در مورد اول، مقدار استاندارد در انحراف معیار σ ضرب می‌شود، حال آنکه در مورد دوم، این مقدار در میانگین ضرب می‌شود.

■ **مثال ۸-۱۱**

مدتهای خدمتدهی در یک باجهٔ صندوق توزیع نرمال با میانگین $\mu = 7.3$ دقیقه و واریانس $\sigma^2 = 11.7$ دقیقه² دارد. به منظور تولید مدت نمونه‌وار خدمتدهی، ابتدا ۱۲ عدد تصادفی از جدول پ-۱ به‌دست آورید

۰٫۱۰۵۲	۰٫۹۸۱۳	۰٫۶۰۳۳	۰٫۲۷۷۴	۰٫۱۴۸۹	۰٫۱۷۵۸
۰٫۸۸۰۳	۰٫۶۴۳۰	۰٫۷۳۵۰	۰٫۱۶۹۹	۰٫۷۴۸۴	۰٫۱۸۱۶

سپس، معادله‌های (۸-۲۹) و (۸-۳۰) را به کار ببرید تا نتیجهٔ زیر به‌دست آید:

$$Y = 7.3 + \sqrt{11.7}\left(\sum_{i=1}^{12} R_i - 6\right) = 6.10$$

بسیاری از نویسندگان کتابهای شبیه‌سازی به منظور تولید مقادیر تصادفی تقریباً نرمال، این روش را توصیه می‌کنند اما روشی دقیق مانند روش بخش ۸-۲، همواره بر هر روش تقریبی ترجیح داده می‌شود. روشهای دقیق بسیاری شناخته شده است و برخی، هم به آسانی مورد استفاده واقع می‌شود

www.matlabdl.com

و هم در برنامهٔ کامپیوتری کاراست. (خوانندهٔ علاقه‌مند را به فیشمن [۱۹۷۸] ارجاع می‌دهیم.) برای نمایش یک طرح تولید دقیق، معادلهٔ (۸-۲۶) با $R_۱ = ۰٫۱۷۵۸$ و $R_۲ = ۰٫۱۴۸۹$ را در نظر بگیرید. دو مقدار تصادفی نرمال به شرح زیر تولید می‌شود

$$Z_۱ = [-۲\ell n(۰٫۱۷۵۸)]^{\frac{۱}{۲}} \cos ۲\pi(۰٫۱۴۸۹) = ۱٫۱۱$$
$$Z_۲ = [-۲\ell n(۰٫۱۷۵۸)]^{\frac{۱}{۲}} \sin ۲\pi(۰٫۱۴۸۹) = ۱٫۵۰$$

■

این روش به یک دوازدهم اعداد تصادفی مورد نیاز روش تقریبی احتیاج دارد، ولی محاسبات سینوس، کسینوس و لگاریتم روی کامپیوتر نسبتاً ناکاراست. روشهای کاراتر از سوی فیشمن [۱۹۷۸] و اشمایزر [۱۹۸۱] مورد بحث قرار گرفته است.

۸-۴ روش رد و قبول

تصور کنید تحلیلگری نیاز به ایجاد روشی برای تولید مقادیر تصادفی X با توزیع یکنواخت بین $\frac{۱}{۴}$ و ۱ دارد. یک راه انجام کار، برداشتن گامهای زیر است:

گام ۱. عدد تصادفی R را تولید کنید.

گام ۲ (الف). اگر $R \geq \frac{۱}{۴}$ است، $X = R$ را قبول کنید و سپس به گام ۳ بروید.

گام ۲ (ب). اگر $R < \frac{۱}{۴}$ است، R را رد کنید و به گام ۱ برگردید.

گام ۳. اگر مقدار تصادفی یکنواخت دیگری در $[\frac{۱}{۴}, ۱]$ مورد نیاز است، با شروع در گام ۱، شیوه را تکرار کنید. وگرنه، توقف کنید.

هر گاه گام ۱ اجرا شود، عدد تصادفی جدیدی مانند R باید تولید شود. در این روش رد و قبول، گام ۲ (الف) یک «قبول» و گام ۲ (ب) یک «رد» است. به منظور تلخیص روش، مقادیر تصادفی (R) با توزیعی (در اینجا یکنواخت در فاصلهٔ $[۰, ۱]$) تولید می‌شود تا شرطی ($R \geq \frac{۱}{۴}$) برقرار شود. وقتی که سرانجام این شرط برقرار شود می‌توان مقدار تصادفی موردنظر X (در اینجا یکنواخت در فاصلهٔ $[\frac{۱}{۴}, ۱]$) را محاسبه کرد ($X = R$). با تشخیص اینکه مقادیر قبول شدهٔ R مقادیری مشروط است، می‌توان نشان داد که این شیوه صحیح است، یعنی خود R توزیع موردنظر را ندارد ولی R به شرط پیشامد $\{R \geq \frac{۱}{۴}\}$، از توزیع موردنظر برخوردار است. به منظور نشان دادن این مطلب، رابطهٔ $\frac{۱}{۴} \leq a < b \leq ۱$ را در نظر بگیرید؛ در این‌صورت، داریم

$$P\left(a < R \leq b \,\middle|\, \frac{۱}{۴} \leq R \leq ۱\right) = \frac{P(a < R \leq b)}{P(\frac{۱}{۴} \leq R \leq ۱)} = \frac{b-a}{\frac{۳}{۴}} \quad (۸\text{-}۳۱)$$

که احتمالی صحیح برای توزیع یکنواخت در فاصلهٔ $[\frac{۱}{۴}, ۱]$ است. معادلهٔ (۸-۳۱) می‌گوید که توزیع احتمال R، به‌شرط بودن R بین $\frac{۱}{۴}$ و ۱ (همهٔ مقادیر دیگر R دور ریخته می‌شود)، توزیع موردنظر است. بنابراین، اگر $\frac{۱}{۴} \leq R \leq ۱$ باشد، X را با R مساوی قرار دهید.

کارایی هر روش رد و قبول قویاً به توانایی می‌نیمم کردن تعداد ردیها بستگی دارد. احتمال ردی در این مثال $P(R < \frac{۱}{۴}) = \frac{۱}{۴}$ است، که تعداد ردیها متغیری تصادفی با توزیع هندسی با احتمال «موفقیت» $p = \frac{۳}{۴}$ و میانگین تعداد ردی $(\frac{۱}{p} - ۱) = \frac{۴}{۳} - ۱ = \frac{۱}{۳}$ است. (توزیع هندسی در مثال ۸-۸ مورد بحث قرار گرفت.) میانگین تعداد اعداد تصادفی مورد نیاز، R، به منظور تولید یک مقدار X یکی بیشتر از تعداد ردیها و بنابراین، $\frac{۴}{۳} = ۱٫۳۳$ است. به عبارت دیگر، به منظور تولید ۱۰۰۰ مقدار X، تقریباً به ۱۳۳۳ عدد تصادفی R نیاز است.

در وضعیت فعلی، شیوهٔ دیگری برای تولید مقدار یکنواخت در فاصلهٔ $[\frac{۱}{۴}, ۱]$، وجود دارد، یعنی معادلهٔ (۸-۴) که به شکل سادهٔ $X = (\frac{۳}{۴})R + \frac{۱}{۴}$ در می‌آید. اینکه آیا روش رد و قبول کاراتر است یا شیوهٔ دیگری از قبیل روش تبدیل معکوس [معادلهٔ (۸-۴)]، به ملاحظاتی چند بستگی دارد. کامپیوتر مورد استفاده، مهارت برنامه‌نویس و کارایی نسبی تولید اعداد تصادفی اضافی (رد شده) موردنیاز روش رد و قبول باید با محاسبات لازم در شیوهٔ دیگر مقایسه شود. در عمل، ملاحظات مربوط به کارایی تولید به متخصصانی واگذار می‌شود که عهده‌دار انجام آزمایشهای مفصل در زمینهٔ مقایسهٔ روشهای مختلف هستند (یعنی از زمانی که نیاز مدل شبیه‌سازی به مدت اجرای زیاده از حد ناشی از مولد مورد استفاده آغاز شود).

برای توزیع یکنواخت در فاصلهٔ $[\frac{۱}{۴}, ۱]$، به‌کارگیری روش تبدیل معکوس معادلهٔ (۸-۴) بدون تردید بسیار آسانتر و احتمالاً کاراتر از روش رد و قبول است. قصد اصلی این مثال، تشریح و ارائهٔ مفهوم اساسی روش رد و قبول بود. اما برای برخی از توزیعهای مهم از قبیل گاما، معکوس cdf به شکل بسته وجود ندارد و به این ترتیب، روش تبدیل معکوس قابل استفاده نیست. در مورد سایر توزیعهای مهم از قبیل نمایی و نرمال، روش رد و قبول و دیگر روشهای پیشرفته‌تر به طرحهای بسیار کاراتر تولید می‌انجامد. این روشهای پیشرفته‌تر را فیشمن [۱۹۷۸] تلخیص کرده است.

در زیر بخشهای بعد، روش رد و قبول در مورد تولید مقادیر تصادفی برای توزیعهای پواسون و گاما توضیح داده می‌شود.

۸-۴-۱ توزیع پواسون

هر متغیر تصادفی پواسون، N، با میانگین $\alpha > ۰$، دارای pmf

$$p(n) = P(N = n) = \frac{e^{-\alpha}\alpha^n}{n!}, \quad n = ۰, ۱, ۲, \cdots$$

است، اما مهمتر اینکه می‌توان N را به‌عنوان تعداد موارد ورود از یک فرایند ورود پواسون در واحد زمان تعبیر کرد. از بخش ۴-۵ به یاد دارید که مدتهای بین دو ورود مشتریان متوالی، A_1، A_2، ... توزیع نمایی با آهنگ α دارد (یعنی، α میانگین تعداد ورود در واحد زمان است)؛ به‌علاوه، می‌توان هر مقدار نمایی را طبق معادلهٔ (۸-۲(ب)) تولید کرد. بدین ترتیب، بین توزیع (گسستهٔ) پواسون و توزیع (پیوستهٔ) نمایی رابطه‌ای وجود دارد؛ یعنی، شرط لازم و کافی برای صدق رابطهٔ

$$N = n \qquad (۸\text{-}۳۲(الف))$$

برقراری رابطهٔ زیر است:

$$A_1 + A_2 + \cdots + A_n \le 1 < A_1 + \cdots + A_n + A_{n+1} \qquad (۸\text{-}۳۲(ب))$$

از معادلهٔ (۸-۳۲(الف))، یعنی $N = n$، چنین بر می‌آید که در یک واحد زمان دقیقاً n ورود وجود داشته است؛ اما رابطهٔ (۸-۳۲(ب)) می‌گوید که nامین ورود پیش از زمان ۱ و $(n+1)$امین ورود پس از زمان ۱ رخ داده است. این دو معنی آشکارا معادل است. اینک، اقدام به تولید مدتهای بین دو ورود نمایی کنید تا جایی که یک ورود، مثلاً $n+1$، پس از زمان ۱ رخ دهد؛ پس N را مساوی با n قرار دهید.

با هدف تولید کارا، معمولاً رابطهٔ (۸-۳۲(ب)) ابتدا با استفاده از معادلهٔ (۸-۲(ب))، یعنی $A_i = (\frac{-1}{\alpha})\ln R_i$، ساده می‌شود تا رابطهٔ

$$\sum_{i=1}^{n} -\frac{1}{\alpha}\ln R_i \le 1 < \sum_{i=1}^{n+1} -\frac{1}{\alpha}\ln R_i$$

به‌دست آید. سپس، با ضرب کردن $-\alpha$ که علامت نامساوی را برعکس می‌کند، و استفاده از این واقعیت که جمع لگاریتمها، لگاریتم یک حاصلضرب است، رابطهٔ

$$\ln \prod_{1}^{n} R_i = \sum_{i=1}^{n} \ln R_i \ge -\alpha > \sum_{i=1}^{n+1} \ln R_i = \ln \prod_{1}^{n+1} R_i$$

به‌دست می‌آید. سرانجام، با استفاده از رابطهٔ $e^{\ln x} = x$ به ازای هر عدد x، رابطهٔ

$$\prod_{1}^{n} R_i \ge e^{-\alpha} > \prod_{1}^{n+1} R_i \qquad (۸\text{-}۳۳)$$

به دست می‌آید که معادل رابطهٔ (۸-۳۲(ب)) است. شیوهٔ مربوط به تولید هر مقدار تصادفی پواسون، N، با برداشتن گامهای زیر عرضه می‌شود:

گام ۱. n را مساوی با صفر و P را مساوی با یک قرار دهید.

گام ۲. عدد تصادفی R_{n+1} را تولید و P را با PR_{n+1} جانشین کنید.

گام ۳. اگر $P < e^{-\alpha}$ است، بپذیرید که $N = n$ است. در غیر این صورت، n جاری را رد و به آن یک واحد اضافه کنید و به گام ۲ برگردید.

توجه داشته باشید که با کامل کردن گام ۲، P با عبارت سمت راست در رابطهٔ (۸-۳۳) مساوی می‌شود. مجدداً، ایدهٔ اساسی روش رد و قبول نمایش داده می‌شود: اگر در گام ۳ رابطهٔ $P \ge e^{-\alpha}$ برقرار باشد، n رد می‌شود و فرایند تولید دست‌کم باید یک آزمایش دیگر را انجام دهد.

به منظور تولید هر مقدار پواسون، N، به‌طور متوسط چند عدد تصادفی مورد نیاز خواهدبود؟ اگر $N = n$ باشد، به $n + 1$ عدد تصادفی نیاز داریم، پس تعداد متوسط از رابطهٔ

$$E(N + 1) = \alpha + 1$$

به‌دست می‌آید که اگر میانگین توزیع پواسون، α، بزرگ باشد، کاملاً بزرگ خواهد بود.

■ **مثال ۸-۱۲**

سه مقدار پواسون با میانگین $\alpha =$ ۰٫۲ تولید کنید. ابتدا $e^{-\alpha} = e^{-0.2} =$ ۰٫۸۱۸۷ را محاسبه کنید و سپس دنباله‌ای از اعداد تصادفی R را از جدول پ-۱ به‌دست آورید و از گامهای ۱ تا ۳ فوق پیروی کنید:

گام ۱. n را مساوی با صفر و P را مساوی با یک قرار دهید.

گام ۲. $R_1 =$ ۰٫۴۳۵۷، $P = 1 \cdot R_1 =$ ۰٫۴۳۵۷.

گام ۳. چون $P =$ ۰٫۴۳۵۷ $< e^{-\alpha} =$ ۰٫۸۱۸۷ است، $N =$ ۰ را بپذیرید.

گام ۱-۳. ($R_1 =$ ۰٫۴۱۴۶ به $N =$ ۰ می‌انجامد.)

گام ۱. $n =$ ۰، $P =$ ۱.

گام ۲. $R_1 =$ ۰٫۸۳۵۳، $P = 1 \cdot R_1 =$ ۰٫۸۳۵۳.

گام ۳. چون $P \ge e^{-\alpha}$ است، $n =$ ۰ را رد کنید و با $n =$ ۱ به گام ۲ بازگردید.

گام ۲. $R_2 =$ ۰٫۹۹۵۲، $P = R_1R_2 =$ ۰٫۸۳۱۳.

گام ۳. چون $P \ge e^{-\alpha}$ است، $n =$ ۱ را رد کنید و با $n =$ ۲ به گام ۲ بازگردید.

گام ۲. $R_3 =$ ۰٫۸۰۰۴، $P = R_1R_2R_3 =$ ۰٫۶۶۵۴.

گام ۳. چون $P < e^{-\alpha}$ است، $N =$ ۲ را بپذیرید.

محاسبات لازم برای تولید این سه مقدار تصادفی پواسون به شرح زیر خلاصه شده است:

n	R_{n+1}	P	قبول/رد	نتیجه
۰	۰٫۴۳۵۷	۰٫۴۳۵۷	$P < e^{-\alpha}$ (قبول)	$N = 0$
۰	۰٫۴۱۴۶	۰٫۴۱۴۶	$P < e^{-\alpha}$ (قبول)	$N = 0$
۰	۰٫۸۳۵۳	۰٫۸۳۵۳	$P \geq e^{-\alpha}$ (رد)	
۱	۰٫۹۹۵۲	۰٫۸۳۱۳	$P \geq e^{-\alpha}$ (رد)	
۲	۰٫۸۰۰۴	۰٫۶۶۵۴	$P < e^{-\alpha}$ (قبول)	$N = 2$

در اینجا، به منظور تولید سه مقدار پواسون ($N = 0$، $N = 0$ ، و $N = 2$)، پنج عدد تصادفی، R، نیاز داشتیم، ولی در اجرای بلند تولید، مثلاً ۱۰۰۰ مقدار پواسون با میانگین $\alpha = 0.2$، تقریباً به $1000(\alpha + 1)$ یا ۱۲۰۰ عدد تصادفی نیاز خواهیم داشت. ■

■ مثال ۸-۱۳

اتوبوسهایی طبق فرایند پواسون با میانگین یک اتوبوس در هر ۱۵ دقیقه به یک تقاطع وارد می‌شوند. یک مقدار تصادفی، N، تولید کنید که معرف تعداد اتوبوسهای واردشونده در خلال یک دورهٔ زمانی یک ساعته باشد. در این‌صورت، N توزیع پواسون با میانگین چهار اتوبوس در ساعت دارد. ابتدا $e^{-\alpha} = e^{-4} = 0.0183$ را محاسبه کنید. به‌کارگیری همان اعداد تصادفی مورد استفاده در مثال ۸-۱۲، به نتایج خلاصهٔ شدهٔ زیر می‌رسد:

n	R_{n+1}	P	قبول/ رد	نتیجه
۰	۰٫۴۳۵۷	۰٫۴۳۵۷	$P \geq e^{-\alpha}$ (رد)	
۱	۰٫۴۱۴۶	۰٫۱۸۰۶	$P \geq e^{-\alpha}$ (رد)	
۲	۰٫۸۳۵۳	۰٫۱۵۰۸	$P \geq e^{-\alpha}$ (رد)	
۳	۰٫۹۹۵۲	۰٫۱۵۰۲	$P \geq e^{-\alpha}$ (رد)	
۴	۰٫۸۰۰۴	۰٫۱۲۰۲	$P \geq e^{-\alpha}$ (رد)	
۵	۰٫۷۹۴۵	۰٫۰۹۵۵	$P \geq e^{-\alpha}$ (رد)	
۶	۰٫۱۵۳۰	۰٫۰۱۴۶	$P < e^{-\alpha}$ (قبول)	$N = 6$

بی‌درنگ می‌توان دریافت که مقداری بزرگتر برای α(اینجا $\alpha = 4$) معمولاً به اعداد تصادفی بیشتری نیاز دارد؛ اگر ۱۰۰۰ مقدار تصادفی موردنیاز باشد، تقریباً به $1000(\alpha + 1) = 5000$ عدد تصادفی نیاز خواهیم داشت.

هرگاه α بزرگ باشد، مثلاً $\alpha \geq 15$، روش رد و قبول به شرحی که عرضه شد بسیار پرهزینه می‌شود، اما خوشبختانه یک روش تقریبی مبتنی بر توزیع نرمال، بسیار مناسب است. هرگاه

میانگین، α، بزرگ باشد،

$$Z = \frac{N - \alpha}{\sqrt{\alpha}}$$

تقریباً توزیع نرمال با میانگین صفر و واریانس یک دارد، که این روشی تقریبی را می‌رساند. ابتدا طبق معادلهٔ (۸-۲۶) یا (۸-۲۸) مقدار نرمال استاندارد Z را تولید و سپس مقدار موردنظر پواسون، N، را از رابطهٔ

$$N = \lceil \alpha + \sqrt{\alpha}Z - 0.5 \rceil \qquad (۸\text{-}۳۴)$$

تولید کنید که $\lceil . \rceil$ تابع گرد کردن به سمت بالاست که در زیر بخش ۸-۱-۷ تشریح شد. (اگر $\alpha + \sqrt{\alpha}Z - 0.5 < 0$ باشد، N را مساوی با صفر قرار دهید.) «۰٫۵» به‌کارگرفته شده در فرمول باعث می‌شود که تابع گرد کردن به سمت بالا به تابع «گرد کردن به نزدیکترین عدد صحیح» تبدیل شود. معادلهٔ (۸-۳۴) یکی از روشهای رد و قبول نیست، ولی استفاده از آن به‌عنوان گزینه‌ای در مقابل روش رد و قبول، روشی کاملاً کارا و دقیق برای تولید مقادیر پواسون با میانگینی بزرگ را فراهم می‌آورد. ■

۸-۴-۲ توزیع گاما

به منظور تولید مقادیر تصادفی گاما چند روش رد و قبول به وجود آمده است [فیشمن، ۱۹۷۸]. یکی از کاراترین این روشها مدیون چنگ [۱۹۷۷] است که میانگین تعداد آزمایشهای آن به ازای هر مقدار پارامتر شکل $\beta \geq 1$ بین ۱٫۱۳ و ۱٫۴۷ است.

اگر پارامتر شکل، β، عددی صحیح باشد، مثلاً $\beta = k$، یک امکان، استفاده از روش پیچش از بخش ۸-۳-۱ است زیرا توزیع ارلنگ مورد خاصی از توزیع کلی‌تر گاماست. از سوی دیگر، روش رد و قبول به شرحی که در اینجا می‌آید روشی بسیار کارا برای توزیع ارلنگ است به ویژه اگر $\beta = k$ بزرگ باشد. روال زیر مقادیر تصادفی گاما با پارامتر مقیاس θ و پارامتر شکل β، یعنی مقادیری با میانگین $\frac{1}{\theta}$ و واریانس $\frac{1}{\beta\theta^2}$ تولید می‌کند. گامهای موردنظر به شرح زیر است:

گام ۱. $a = (2\beta - 1)^{1/2}$ و $b = \ell n4 + (\alpha - 1)\ln\alpha - \alpha$ را محاسبه کنید.

گام ۲. R_1 و R_2 را تولید کنید.

گام ۳. $X = \beta[R_1/(1 - R_1)]^{1/a}$ را محاسبه کنید.

گام ۴ (الف). اگر $(\beta - 1)\ell n X - X < b + (1 + \frac{1}{a})\ell n(\frac{1}{R_1} - 1) + \ell n(R_1^2 R_2)$ باشد، X را رد کنید و به گام ۲ بازگردید.

گام ۴ (ب). در غیر این صورت از X به‌عنوان مقدار موردنظر استفاده کنید.

این الگوریتم تا اینجا از توزیع گاما با پارامترهای β و ۱ مقدار تصادفی تولید می‌کند اگر نیاز به تولید مقدار

www.matlabdl.com

تصادفی از توزیع گاما با پارامترهای β و θ (همانند بخش ۴ـ۴) باشد، باید گام زیر را نیز اضافه کرد.
گام ۵. X را با θX جانشین کنید.

ایدهٔ اساسی همهٔ روشهای رد و قبول مجدداً در اینجا نمایش داده می‌شود ولی اثبات این مثال از مجال بررسی این کتاب خارج است. مقدار $X = \beta[R_1/(1 - R_1)]^a$ در گام ۳ توزیع گاما ندارد، اما رد کردن مقادیر X مشخصی در گام ۴ (الف) تضمین می‌کند که مقادیر قبول شده در گام ۴ (ب) توزیع گاما داشته باشد.

■ مثال ۸-۱۴

معلوم شده اتست که مدتهای از کارماندگی یک ماشین شیرینی‌ساز با تولید زیاد، توزیع گاما با میانگین ۲٫۲ دقیقه و واریانس ۲٫۱۰ دقیقه^2 دارد. پس، داریم $2.2 = \frac{1}{\theta}$ و $2.10 = \frac{1}{\beta\theta^2}$ که به معنای $\beta = 2.30$ و $\theta = 0.4545$ است.

گام ۱. $a = 1.90$، $b = 3.74$.

گام ۲. $R_1 = 0.832$ و $R_2 = 0.021$ را تولید کنید.

گام ۳. $X = 2.3(0.832/0.168)^{1/9} = 48.1$ را محاسبه کنید.

گام ۴. $X = 48.1 > 3.74 - \ell n[(0.832)^2 0.021] = 7.97$ است. پس X را رد کنید و به گام ۲ بازگردید.

گام ۲. $R_1 = 0.434$ و $R_2 = 0.716$ را تولید کنید.

گام ۳. $X = 2.3(0.434/0.566)^{1/9} = 1.389$ را محاسبه کنید.

گام ۴. چون $X = 1.389 \leq 3.74 - \ell n[(0.434)^2 0.716] = 5.74$ است، X را قبول کنید.

گام ۵. X را به $\beta\theta = 1.045$ تقسیم کنید تا $X = 1.329$ به‌دست آید.

در این مثال با صرف دو آزمایش (یعنی، یک رد) یک مقدار تصادفی قابل‌قبول برخوردار از توزیع گاما تولید شد، ولی به‌طور متوسط برای تولید مثلاً ۱۰۰۰ مقدار گاما، این روش به ۱۱۳۰ تا ۱۴۷۰ آزمایش، یا به طریق معادل، به ۲۲۶۰ تا ۲۹۴۰ عدد تصادفی نیاز دارد. این روش برای محاسبات دستی تقریباً خسته‌کننده است، اما برنامه‌نویسی آن روی کامپیوتر ساده و در حال حاضر یکی از کاراترین مولدهای شناخته شدهٔ گاماست. ■

۸-۵ خلاصه

اصول اساسی تولید مقدار تصادفی با استفاده از روشهای تبدیل معکوس، پیچش و رد و قبول را معرفی کردیم و با مثالهایی توضیح دادیم. روشهای تولید بیشتر توزیعهای مهم پیوسته و گسسته، به اضافهٔ توزیعهای تجربی ارائه شد. به منظور مرور یک بررسی شامل آخرین پیشرفتها در این زمینه، خواننده را به فیشمن [۱۹۷۸] یا اشمایزر [۱۹۸۱] ارجاع می‌دهیم.

منابع

Box, G. E. P., and M. F. Muller [1958], " A Note on the Generation of Random Normal Deviates," *Annals of Mathematical Statistics*, Vol. 29, pp. 610-11.

Cheng, R. C. H. [1977], "The Generation of Gamma Variables," *Applied Statistician*, Vol. 26, No. 1, pp. 71-75.

Fishman, George S. [1978], *Principles of Discrete Event Simulation*, Wiley, New York.

Gordon, Geoffrey[1975] , *The Application of GPSS V to Discrete System Simulation*, Prentice- Hall, Englewood Cliffs, N. J.

Schmeiser, Bruce W. [1981] , " Random Variate Generation :A State of the Art View", in *Simulation With Discrete Models: A State of the Art View*, T. I. Oren, C. M. Shub, and P. F. Roth, eds.

Schmidt, J. W., and R. E. Taylor [1970], *Simulation and Analysis of Industrial Systems*, Irwin, Homewood, Ill.

تمرینها

۸-۱ یک مولد مقدار تصادفی برای متغیر تصادفی X با pdf

$$f(x) = \begin{cases} e^{2x}, & -\infty < x \leq 0 \\ e^{-2x}, & 0 < x < \infty \end{cases}$$

ایجاد کنید.

۸-۲ طرحی برای تولید مقدار از توزیع مثلثی با pdf

$$f(x) = \begin{cases} \frac{1}{2}(x-2), & 2 \leq x \leq 3 \\ \frac{1}{2}(2 - \frac{x}{3}), & 3 < x \leq 6 \\ 0, & \text{در غیر این صورت} \end{cases}$$

ایجاد کنید. ده مقدار تصادفی تولید، میانگین نمونه را محاسبه و آن را با میانگین حقیقی

توزیع مقایسه کنید.

۸-۳ مولدی برای یک توزیع مثلثی با دامنهٔ $(۱, ۱۰)$ و مد $۴ = x$ ایجاد کنید.

۸-۴ مولدی برای یک توزیع مثلثی با دامنهٔ $(۱, ۱۰)$ و میانگین ۴ ایجاد کنید.

۸-۵ مولدی برای یک متغیر تصادفی پیوسته با دامنهٔ ۳− تا ۴ ایجاد کنید که cdf آن به شرح زیر است:

$$F(x) = \begin{cases} ۰, & x \le -۳ \\ \frac{۱}{۲} + \frac{x}{۶}, & -۳ < x \le ۰ \\ \frac{۱}{۲} + \frac{x^۲}{۳۲}, & ۰ < x \le ۴ \\ ۱, & ۴ < x \end{cases}$$

۸-۶ مولدی برای توزیعی که cdf آن به‌صورت $F(x) = x^۴/۱۶$, $۰ \le x \le ۲$ است ایجاد کنید.

۸-۷ مولدی برای توزیعی که pdf آن به‌صورت $f(x) = x^۲/۹$ ،$۰ \le x \le ۳$ است ایجاد کنید.

۸-۸ مولدی برای یک متغیر تصادفی ایجاد کنید که pdf آن به‌صورت زیر است:

$$f(x) = \begin{cases} \frac{۱}{۳}, & ۰ \le x \le ۲ \\ \frac{۱}{۲۴}, & ۲ < x \le ۱۰ \\ ۰, & \text{در غیر این صورت} \end{cases}$$

۸-۹ cdf متغیر تصادفی گسستهٔ X به صورت

$$F(x) = \frac{x(x+۱)(۲x+۱)}{n(n+۱)(۲n+۱)} \,,\; x = ۱, ۲, \ldots, n$$

است. به ازای $n = ۴$ و با استفاده از $R_۱ = ۰/۸۳$ ،$R_۲ = ۰/۲۴$ و $R_۳ = ۰/۵۷$ سه مقدار برای X تولید کنید.

۸-۱۰ معلوم شده است که مدتهای یک فرایند خودکار تولید تا بازمانی آن، توزیعی تصادفی طبق مدل ویبول با پارامترهای $\beta = ۲$ و $\alpha = ۱۰$ دارد. معادلهٔ (۸-۵) را به‌دست آورید و سپس با استفاده از پنج عدد تصادفی از جدول پ-۱ آن را به منظور تولید پنج مقدار از این توزیع ویبول مورد استفاده قرار دهید.

۸-۱۱ در یک بانک داده‌هایی در مورد مدتهای خدمتدهی باجهٔ اتوبانک گردآوری شده است.

این داده‌ها در قالب فواصل به شرح زیر تلخیص شده است:

فاصله (ثانیه)	فراوانی
۱۵-۳۰	۱۰
۳۰-۴۵	۲۰
۴۵-۶۰	۲۵
۶۰-۹۰	۳۵
۹۰-۱۲۰	۳۰
۱۲۰-۱۸۰	۲۰
۱۸۰-۳۰۰	۱۰

به منظور تولید مدتهای خدمتدهی به روش جدولگرد، جدولی همانند جدول ۸-۳ ایجاد و با استفاده از اعداد تصادفی چهار رقمی، پنج مقدار برای مدت خدمتدهی تولید کنید.

۸-۱۲ فرض کنید مدتهای پاسخ گروه آتش‌نشانان در مثال ۸-۳ در رابطهٔ $۳ \le x \le ۰/۲۵$ صدق می‌کند. جدول ۸-۵ را برای رعایت این فرض اصلاح کنید. با استفاده از اعداد تصادفی چهار رقمی از جدول پ-۱، پنج مقدار برای مدت پاسخ تولید کنید.

۸-۱۳ برای نسخه‌ای مقدماتی از یک مدل شبیه‌سازی، فرض شد که تعداد پالتهایی که باید در سکوی بارگیری در کامیونی بار شود، بین ۸ و ۲۴ توزیع یکنواخت دارد. با این فرض که بارهای کامیونهای متوالی مستقل است، روشی برای تولید X ایجاد کنید. از روش موجود در مثال ۸-۶ برای توزیعهای یکنواخت استفاده کنید. سرانجام، با استفاده از اعداد تصادفی و چهار رقمی، بارهای ده کامیون متوالی را تولید کنید.

۸-۱۴ با گردآوری داده‌های بیشتر، معلوم شد که توزیع مثال ۸-۷ نسبت به توزیع یکنواخت به‌گونه‌ای که در تمرین ۱۳ فرض شد، تقریب بهتری برای تعداد پالتهای بارگیری شده است. با به‌کارگیری همان اعداد تصادفی مورد استفاده در تمرین ۱۳، بارهای ده کامیون متوالی را با استفاده از معادلهٔ (۸-۱۹) تولید کنید.

۸-۱۵ معلوم شده است که تقاضای هفتگی، X، برای کالایی کم تقاضا، طبق توزیع هندسی در دامنهٔ $\{۰, ۱, ۲, \cdots\}$ و با میانگین تقاضای هفتگی ۲/۵ قلم به‌خوبی تقریب زده می‌شود. با استفاده از اعداد تصادفی جدول پ-۱ ده مقدار برای تقاضا در هفته، X، تولید کنید. (*راهنمایی*: میانگین یک توزیع هندسی با پارامتر p و دامنهٔ $\{q, q+۱, \cdots\}$ عبارت از $\frac{۱}{p} + q - ۱$ است.)

۸-۱۶ تصور کنید که در تمرین ۱۵ معلوم شده است که تقاضا توزیع پواسون با میانگین ۲/۵ قلم در هفته دارد. با استفاده از اعداد تصادفی موجود در جدول پ-۱، ده مقدار تقاضا در هفته، X، تولید کنید. تفاوتهای موجود بین توزیعهای هندسی و پواسون را مورد بحث قرار دهید.

۸-۱۷ معلوم شده است که مهلتهای تحویل، توزیع نمایی با میانگین ۳٫۷ روز دارد. برای این توزیع پنج مهلت تحویل تصادفی تولید کنید.

۸-۱۸ معلوم شده است که مدتهای نگهداری یک روال تولید تغییر می‌کند و به‌صورت متغیر تصادفی نرمالی با میانگین ۳۳ دقیقه و واریانس ۴ دقیقه^{۲} مدلسازی شده است. با این توزیع مفروض و به یکی از روشهای این فصل، پنج مدت تصادفی نگهداری و تعمیر تولید کنید.

۸-۱۹ ماشینی پس از بازمانی یا پس از پنج ساعت کار برحسب اینکه کدام زودتر رخ دهد از خط تولید بیرون آورده می‌شود. با بکار انداختن ماشینهای همانند تا بازمانی، معلوم شده است که مدت تا بازمانی، X، توزیع ویبول با $\alpha = ۸$، $\beta = ۰٫۷۵$ و $\nu = ۰$ دارد (به بخش ۴-۴ و زیر بخش ۸-۱-۳ مراجعه کنید). بدین ترتیب، مدت تا بیرون آوردن ماشین از خط تولید را می‌توان به‌صورت $Y = \min(X, ۵)$ معرفی کرد. به منظور تولید Y شیوه‌ای گام به گام ایجاد کنید.

۸-۲۰ مدت تا از خدمت خارج کردن قطعه‌ای بین صفر تا ۸ ساعت توزیع یکنواخت دارد. دو قطعهٔ مستقل از این قبیل را به‌صورت زنجیره‌ای قرار می‌دهیم و هرگاه یکی از دو قطعه از کار بماند همهٔ سیستم از کار خواهد ماند. اگر $X_i (i = ۱, ۲)$ معرف مدت عمل قطعه باشد، $Y = \min(X_۱, X_۲)$ معرف مدت عمر سیستم خواهد بود. برای تولید Y دو راه متمایز ارائه کنید. [*راهنمایی*: یک راه نسبتاً ساده است. برای راه دوم، ابتدا تابع تجمعی Y را محاسبه کنید: به ازای $۰ \le y \le ۸$، $۱ - P(Y > y) = P(Y \le y) = F_Y(y)$، بعد، برابری $\{Y > y\} = \{X_۱ > y \text{ و } X_۲ > y\}$ و استقلال $X_۱$ و $X_۲$ را مورد استفاده قرار دهید. پس از یافتن $F_Y(y)$، با روش تبدیل معکوس کار را ادامه دهید.]

۸-۲۱ مدتهای عمر قطعات در تمرین ۲۰ توزیع نمایی، یکی با میانگین ۲ ساعت و دیگری با میانگین ۶ ساعت دارد. با این فرض تازه دوباره روی تمرین ۲۰ کار کنید. کارایی نسبی دو طرح ارائه شدهٔ تولید را مورد بحث قرار دهید.

۸-۲۲ با استفاده از روش پیچش، روشی برای تولید مقدار تصادفی دو جمله‌ای ارائه کنید. [*راهنمایی*: X را می‌توان به‌عنوان تعداد موفقیتها در n آزمایش مستقل برنویی معرفی کرد که هر موفقیت دارای احتمال p است. بنابراین، $X = \sum_{i=۱}^{n} X_i$ است که $P(X_i = ۱) = p$ و $P(X_i = ۰) = ۱ - p$ است.]

۸-۲۳ به منظور تولید مقدار تصادفی هندسی، X، با پارامتر p و دامنهٔ $\{۰, ۱, ۲, \cdots\}$، یک روش رد و قبول ایجاد کنید. [*راهنمایی*: X را می‌توان به عنوان تعداد آزمایشها پیش از رخداد اولین موفقیت در دنباله‌ای از آزمایشها مستقل برنویی در نظر گرفت.]

۸-۲۴ به منظور تولید مقادیر نرمال استاندارد با روش دقیق مورد بحث در این فصل، برنامه‌ای کامپیوتری به زبان FORTRAN یا BASIC بنویسید و از آن برای تولید ۱۰۰۰ مقدار

استفاده کنید. به ازای $z = -۴, -۳, -۲, -۱, ۰, ۱, ۲, ۳, ۴$، احتمال حقیقی قرار گرفتن مقدار در فاصله $(-\infty, z)$ را محاسبه کنید، یعنی $\Phi(z)$ را با فراوانی نسبی عملاً مشاهده شده مقایسه کنید. مسأله را برای هر یک از دو روش تقریبی تکرار کنید. سه روش را با هم مقایسه کنید.

۸-۲۵ به منظور تولید مقادیر گاما با پارامتر شکل β و پارامتر مقیاس θ، برنامه‌ای کامپیوتری به زبان FORTRAN یا BASIC بنویسید. به ازای $\beta = ۲٫۵$ و $\theta = ۰٫۲$، هزار مقدار تولید و میانگین حقیقی، $\frac{۱}{\theta} = ۵$، را با میانگین نمونه مقایسه کنید.

۸-۲۶ به منظور تولید ۲۰۰ مقدار از یکی از متغیرهای تمرینهای ۱ تا ۲۳، برنامه‌ای کامپیوتری به زبان FORTRAN یا BASIC بنویسید. هیستوگرمی از این ۲۰۰ مقدار بسازید و آن را با تابع چگالی نظری (یا تابع جرم احتمال برای متغیرهای تصادفی گسسته) مقایسه کنید.

www.matlabdl.com

ضمیمهٔ فصل ۸

امروزه روشهایی وجود دارد که با استفاده از آنها می‌توان عملاً برای تمام توزیعهای احتمال تک‌متغیره و توزیعهای تجربی با کامپیوترهای رقمی مقدار تولید کرد. هر گاه برای تولید مقدار از یک توزیع احتمال بیش از یک روش موجود باشد، معمولاً از روشی که به‌کارگیری آن از لحاظ برنامه‌نویسی ساده‌تر است استفاده می‌کنند. برخی از این روشها در زبانهای مختلف شبیه‌سازی مورد استفاده قرار گرفته است. چون انجام شبیه‌سازی هدف اصلی شمرده می‌شود، به‌کارگیری روشهای مهیاشده در قالب زبانهای شبیه‌سازی موجه است. در واقع، با استفاده از یک برنامهٔ آماده، نیازی به صرف وقت در زمینهٔ فراگیری روشهای متفاوت، نوشتن برنامه برای آنها و تعیین نقاط قوت و ضعف روشها نسبت به یکدیگر احساس نمی‌شود.

هر چند که معترفیم هدف اصلی چیزی جز انجام شبیه‌سازی نیست بررسی روشهای گوناگون تولید مقدار تصادفی را به دلیل در برداشتن مزایای عمده توصیه می‌کنیم. روشهای سریع به مقدار کمتری از وقت CPU نیاز دارد و روشهای کوتاه فضای کمتری از حافظه را اشغال می‌کند. نیاز به وقت کمتری از CPU و فضای کوچکتری از حافظه مترادف با صرفه‌جویی در هزینه‌ها یا تخصیص بخش بیشتری از امکانات مالی به سایر فعالیتهاست. به علاوه، برخی از روشها از دقت عددی بیشتری برخوردار است و تقلید بهتری را در زمینهٔ دریافت ورودیها میسر می‌سازد. چنین تقلیدی به هنگام تجزیه و تحلیل خروجیها متضمن خطای کمتری خواهد بود.

مطلوبیت روشهای مختلف تولید مقدار تصادفی، از یکسو به فراوانی استفاده از هر روش و از سوی دیگر به معیارهای عملکردی از قبیل مدت به‌کارگیری CPU، فضای موردنیاز در حافظه، دقت عددی، و دقت آماری بستگی دارد. مثلاً، ۵۰ درصد صرفه‌جویی در مدت به‌کارگیری CPU در موردی که زمان لازم برای تولید مقدار تصادفی کمتر از یک درصد از این مدت برای شبیه‌سازی را دربر گیرد و فضای کوچکی از حافظه را اشغال کند امر مهمی محسوب نمی‌شود. بنابراین،

ملاحظات مربوط به مطلوبیت روشهای مختلف تنها وقتی مصداق می‌یابد که هزینهٔ نظیر تولید مقادیر تصادفی در شبیه‌سازی نسبتاً قابل‌توجه باشد.

نوع زبان و سخت افزار نیز بر عملکرد نسبی روشهای مختلف تأثیر می‌گذارد. در هر روش تولید مقدار تصادفی می‌توان از امکانات زبان خاصی به منظور صرفه‌جویی در مدت به‌کارگیری CPU استفاده کرد حال آنکه سایر روشها ممکن است به سبب طبیعت متفاوت خود از انجام این کار ناتوان باشند. استفادهٔ مناسب از امکانات زبان، معمولاً به هنگام به‌کارگیری زبان ASSEMBLY مصداق پیدا می‌کند. از بین دو روش مختلف که یکی از امکانات زبان ASSEMBLY استفاده می‌کند و دیگری آن را به‌کار نمی‌گیرد، روش اول به مدت کمتری برای به‌کارگیری CPU نیاز خواهد داشت؛ حال آنکه اگر دو روش مزبور به زبان SIMSCRIPT II نوشته شود، روش اول ممکن است به‌مراتب به مدت بیشتری برای به‌کارگیری CPU نیاز داشته باشد. دلیل این امر را چنین می‌توان توضیح داد که SIMSCRIPT II لزوماً اجازهٔ استفاده از امکانات همانند را نمی‌دهد.

در زمینهٔ عوامل سخت‌افزاری نیز مثلاً طول کلمه ممکن است بر دقت عددی روشی که از سرعت تولید مناسبی برخوردار است تأثیری نامطلوب داشته باشد. در این مورد دو نکته را باید در نظر داشت. اولاً، طول کوچک کلمه به معنی دقت عددی کمتر برای تمام روشهای تولید مقدار تصادفی است. ثانیاً، هر روش تکرارپذیر که در قدمهای متوالی مقادیر عددی کوچک را به مقادیر عددی بزرگ می‌افزاید منبع دیگری برای بی‌دقتی عددی است و این بی‌دقتی در مورد کامپیوتری که طول کلمهٔ کوتاهتری دارد نسبتاً بیشتر است.

عملکرد روشهای مختلف تولید مقادیر تصادفی، جنبهٔ دیگری نیز دارد که به پارامترهای توزیع احتمال مورد استفاده مربوط می‌شود. در عمل، ممکن است به ازای مقادیری که پارامترها در محدوده‌های خاصی می‌پذیرد، یک روش معین تولید مقدار، عملکردی مناسب داشته باشد حال آنکه همان روش به ازای مقادیری که پارامترها در محدوده‌های دیگر اختیار می‌کند عملکردی نامناسب از خود نشان دهد. این مسأله، امکان به‌کارگیری بیش از یک روش تولید مقدار در قالب یک برنامهٔ کامپیوتری و انتخاب روش مناسب بر اساس مقدار عددی پارامترها یا توابعی از آنها را قابل تعمق می‌سازد. بدیهی است که تعیین روش تولید مقدار توسط هر برنامهٔ کامپیوتری، خود نیز نیاز به زمان و فضای کامپیوتر دارد.

در موردی که حتی مطلوبترین روش تولید مقدار تصادفی نیاز به مدتی قابل ملاحظه برای به‌کارگیری CPU دارد، مسألهٔ بررسی مقادیر عددی پارامتر به‌صورت یک امر جدی و درخور توجه متجلی می‌شود. در چنین شرایطی، تقریبهای آماری کاربرد پیدا می‌کند. مثلاً فرض کنید متغیر تصادفی N توزیع احتمال پواسون با پارامتر α دارد. در مورد بسیاری از روشهای تولید مقدار تصادفی از توزیع پواسون واقعیت این است که مدت به‌کارگیری CPU با مقدار α نسبت مستقیم دارد. در نتیجه، با بزرگ شدن α این مدت نیز افزایش می‌یابد. نظریهٔ احتمال دلالت بر این دارد که با افزایش مقدار α، توزیع احتمال $(N-\alpha)/\sqrt{\alpha}$ به $N[0,1]$ میل می‌کند. در نتیجه، اگر α

به اندازهٔ کافی بزرگ باشد، می‌توان یک مقدار تصادفی از توزیع نرمال صفر-یک، Z، تولید کرد و سپس با گرد کردن $\alpha + Z\sqrt{\alpha}$ به نزدیکترین عدد صحیح، یک مقدار تصادفی تقریبی برای توزیع پواسون موردنظر تولید کرد. چون تولید مقدار تصادفی از توزیع احتمال نرمال به روشی با پارامترهای ثابت بستگی دارد، استفاده از تقریب نرمال به منظور تولید مقدار از توزیع پواسون، با توجه به مدت به‌کارگیری CPU برای تولید مقدار از توزیع نرمال صورت می‌گیرد. در واقع، مدت موردِ بحث به‌عنوان حد مطرح می‌شود. در صورتی که مدت CPU برای تولید مقدار از توزیع پواسون از حد مزبور تجاوز کند، استفاده از تقریب نرمال ممکن است موجه واقع شود.

به هنگام ایجاد روشهای تولید مقدار تصادفی، تمهید استفاده از حد به‌ندرت مورد توجه قرار می‌گیرد. شکی نیست که مشکل بودن تعیین محدوده‌هایی که برای آنها تقریب نرمال از لحاظ آماری از دقت برخوردار است به این بی‌توجهی کمک می‌کند. مثلاً به منظور قابل‌قبول بودن تقریب نرمال برای توزیع پواسون، مقدار α باید به چه بزرگی باشد؟ مسألهٔ دیگر در این زمینه انتخاب ضابطهٔ قضاوت در مورد دقت آماری است. آیا دقتی تا چهار رقم اعشار در مورد هر تابع تجمعی کافی است؟ گرچه پاسخ چنین سؤالی را باید برحسب مورد داد ولی در اغلب موارد دقتی درحد دو رقم اعشار کافی به نظر می‌رسد.

روشهای نمونه‌گیری عموماً از تولید اعداد تصادفی یکنواخت، اعمال تبدیلها و انجام مقایسه تشکیل می‌شود. هر چند که اظهارنظر واضح در مورد کارآیی نسبی روشی برای تولید مقدار تصادفی تنها پس از نوشتن برنامه، اجرای آن و انجام مقایسهٔ نتایج ممکن است، ولی عنوان کردن برخی نکات حتی پیش از به‌کارگیری روش نیز موجه است. اولاً، هزینهٔ تولید مقدار تصادفی بر اساس تعداد اعداد تصادفی مورد نیاز تغییر می‌کند. ثانیاً، شکل الگوریتم مولد اعداد تصادفی بر مدت به‌کارگیری CPU تأثیر می‌گذارد. در واقع، تبدیلهای خطی کمتر از تبدیلهای غیرخطی وقتگیر است. این واقعیت ما را بر آن می‌دارد تا به‌ندرت از تبدیلهای لگاریتمی و نمایی استفاده کنیم هر چند که اکثر زیر برنامه‌هایی که چنین تبدیلهایی را انجام می‌دهد به‌صورت استاندارد شده‌ای به زبان ASSEMBLY نوشته شده است. انجام هر مقایسهٔ منطقی نسبتاً کم‌هزینه است، ولی اگر در اعمال روشی به منظور استفاده نکردن از یک تبدیل ناچار از تولید چند عدد تصادفی و انجام چند مقایسه شویم، لزوماً قادر به صرفه‌جویی در هزینه نخواهیم بود.

اگر با انجام آزمایشهای کافی در مورد روشهای مختلف تولید مقدار تصادفی، یک رده‌بندی از لحاظ درجهٔ مطلوبیت آنها بر اساس زمان اجرای برنامه‌های کامپیوتری با استفاده از زبانهای مختلف ارائه شود، این رده‌بندی لزوماً همیشه معتبر نخواهد ماند. در زمانی که این سطور نوشته می‌شود، تبدیلهای لگاریتمی سینوسی و کسینوسی همگی از طریق زیر برنامه‌ها انجام می‌گیرد. نحوهٔ تولید اعداد تصادفی نیز به همین صورت است. چون قابل تصور است که در نسل آیندهٔ کامپیوترها خصوصیات سخت افزاری اجازهٔ محاسبهٔ لگاریتمها، سینوسها و کسینوسها را بدون نیاز به زیربرنامه‌ها بدهد، می‌توان انتظار داشت که در زمان اجرای روشهای مختلف تولید مقدار

تصادفی تغییراتی پدید آید و رده‌بندی مورد بحث دگرگون شود. هر چند که ایجاد سخت‌افزار مناسب برای تولید مستقیم اعداد تصادفی لزوماً در نسل بعدی کامپیوترها عملی نخواهد شد، ولی چنین امکانی نیز ممکن است بر مطلوبیت نسبی الگوریتمهای نمونه‌گیری مؤثر واقع شود. با تأکید بر پیشرفتهای آتی برآنیم تا به خوانندگان یادآور شویم که هر چند دوران مطرح بودن برخی از الگوریتمها به‌عنوان الگوریتمهای برتر هنوز فرا نرسیده است ولی تعدادی از الگوریتمهای موجود ممکن است در آینده در نقشی مسلط ظاهر شود.

در این بخش به معرفی روش ترکیب در مورد تولید مقادیر تصادفی می‌پردازیم و ساختار نظری این روش و روشهای تبدیل معکوس و رد و قبول را همراه با توضیحاتی ارائه می‌کنیم.

۸ـ۱ـض روش تبدیل معکوس

متغیر تصادفی X با تابع تجمعی F_X مفروض است. فرض کنید تابع معکوس F_X با F_X^{-1} نمادگذاری و به طریق زیر تعریف شود

$$F_X^{-1}(y) = \inf[x : F_X(x) \geq y] \ , \quad 0 \leq y \leq 1 \qquad \text{(۸ـ۱ـض)}$$

رابطهٔ فوق بدین معناست که $F_X^{-1}(y)$ کوچکترین مقدار x را به شرط صدق رابطهٔ $F_X(x) \geq y$ می‌پذیرد. اینک، متغیر تصادفی Y را به‌صورت زیر تعریف کنید که R توزیع احتمال یکنواخت صفرـیک داشته باشد:

$$Y = F_X^{-1}(R) \qquad \text{(۸ـ۲ـض)}$$

بنابراین، داریم

$$F_X(Y) = R,$$

$$P\{Y \leq y\} = P\{F_X^{-1}(R) \leq y\} = P\{R \leq F_X(y)\}$$

$$= \int_0^{F_X(y)} dr = F_X(y) = F_Y(y) \qquad \text{(۸ـ۳ـض)}$$

پس، دو متغیر تصادفی Y و X هم‌توزیع است و برای تولید یک مقدار تصادفی از تابع F_X کافی است یک مقدار تصادفی از تابع F_Y یافت و عملگر F_X^{-1} را در مورد آن به‌کار برد. باید توجه داشت که این روش در مورد X پیوسته وگسسته قابل اعمال است. به موجب نتیجهٔ بالا، الگوریتم زیر ارائه می‌شود:

گام ۱. r را از توزیع یکنواخت صفرـیک تولید کنید.

گام ۲. $x \leftarrow F_X^{-1}(r)$ (x یک مقدار تصادفی از F_X است.)

۸ـ۲ـض مزایای روش تبدیل معکوس

الف) اگر روش تبدیل معکوس قابل اعمال باشد، به منظور تولید یک مقدار تصادفی از توزیع احتمال موردنظر، تنها به یک عدد تصادفی نیاز است.

ب) مزیت دیگر روش تبدیل معکوس سهولت تولید مقدار تصادفی از توزیعهای احتمال بریده است. در موارد بسیاری چنین پیش می‌آید که متغیر تصادفی X، به‌عنوان زمان تحقق امری، ممکن نیست از مقدار معینی کمتر (یا بیشتر) باشد. مثلاً، در یک مسألهٔ صف، به منظور خدمتدهی به یک متقاضی به صرف زمانی طولانی‌تر از مقداری مانند t نیاز است. در این صورت، $P\{X < t\}$ را باید برابر با صفر تلقی کرد حال آنکه به‌طور نظری چنین احتمالی مساوی صفر نیست. یک راه برای تطبیق دادن نظریه با واقعیت، بریدن تابع چگالی f_X در t است. اگر تابع چگالی f_X را در کلی‌ترین حالت در a و b ببُریم $(a < b)$، تابع بریدهٔ f_X^* به‌صورت زیر تعریف خواهد شد:

$$f_X^*(x) = \frac{f_X(x)}{F_X(b) - F_X(a)}, \quad a \leq x \leq b \qquad \text{(۸ـ۴ـض)}$$

تابع تجمعی نظیر f_X^* به شرح زیر نوشته می‌شود:

$$F_X^*(x) = \frac{F_X(x) - F_X(a)}{F_X(b) - F_X(a)}, \ a \leq x \leq b \qquad \text{(۸ـ۵ـض)}$$

به منظور تولید مقدار تصادفی از توزیع بریدهٔ F_X^* از الگوریتم زیر استفاده می‌کنیم:

گام ۱. یک عدد تصادفی مانند r تولید کنید.

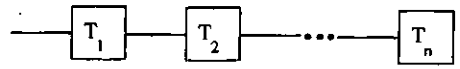
گام ۲. $t \leftarrow F_X(a) + [F_X(b) - F_X(a)]r$

گام ۳. $x \leftarrow F_X^{-1}(t)$

گام ۴. از مقدار x استفاده کنید.

به موجب گام ۲ از تمام اعداد تصادفی استفاده می‌شود و هیچ‌یک از rها بلااستفاده نمی‌ماند. به علاوه، در الگوریتم فوق دو بار از روش تبدیل معکوس استفاده شد؛ بار اول به هنگام تولید t طبق توزیع احتمال یکنواخت در محدودهٔ $[F_X(a), F_X(b)]$ و بار دوم به هنگام تولید مقدار تصادفی x.

ج. مزیت دیگر روش تبدیل معکوس ناظر به استفاده از آن در زمینهٔ تولید مقادیر تصادفی ترتیبی است. در مورد شبیه‌سازی مسائل پایایی، به‌طور رایج از مقادیر تصادفی ترتیبی استفاده می‌شود. اگر یک سیستم از n جزء همانند به‌صورت زنجیره‌ای تشکیل و عمر جزء jام با متغیر تصادفی T_j نمادگذاری شود $(j = 1, \cdots, n)$، عمر سیستم با کوچکترین عمر در میان n جزء معادل خواهد بود. در صورتی که دنبالهٔ متغیرهای تصادفی T_1، T_2، $\cdots$، T_n مرتب شود، دنبالهٔ



شکل ۸ـ۱ـض یک سیستم زنجیره‌ای.

متغیرهای تصادفی ترتیبی $T_{(1)}$، $T_{(2)}$، ...، $T_{(n)}$ حاصل می‌شود. در سیستم زنجیره‌ای فوق، متغیر تصادفی $T_{(1)}$ تغییرات تصادفی عمر سیستم را نشان می‌دهد. اگر n جزء همانند به‌صورت موازی به هم مرتبط شود، $T_{(n)}$ تغییرات تصادفی عمر سیستم را تعریف می‌کند. به هنگام شبیه‌سازی سیستمهای پایایی از نوع فوق، روش متداول تولید عمر برای n جزء و سپس مرتب کردن آنها برحسب مقدار، روشی کُند تلقی می‌شود. اگر اجزاء همانند باشند و مستقل از یکدیگر عمل کنند و روش تبدیل معکوس برای تولید مقدار از تابع چگالی عمر یک جزء قابل اعمال باشد، می‌توان از الگوریتم زیر استفاده کرد. پیش از ارائهٔ الگوریتم، قضیه‌ای را که سرچشمهٔ اعتبار آن است اثبات می‌کنیم:

قضیه. اگر R_1، R_2، ...، R_n یک نمونهٔ تصادفی از توزیع یکنواخت صفر-یک باشد، متغیر تصادفی ترتیبی $R_{(j)}$ طبق تابع چگالی بتا با پارامترهای $j = \alpha$ و $\beta = n - j + 1$ تعریف می‌شود.

اثبات. فرض کنید تعداد R_iهایی که در محدودهٔ $[0, t]$ قرار می‌گیرد با N مشخص شود. در نتیجه، N توزیع احتمال دو جمله‌ای با پارامترهای n و t خواهد داشت. ($t < 1$)، یعنی،

$$P\{R_{(j)} \le t\} = P\{N \ge j\} = \sum_{i=j}^{n} \binom{n}{i} t^i (1-t)^{n-i}$$

به‌علاوه، می‌توان رابطهٔ زیر را نوشت

$$\frac{d}{dt}P\{N \ge j\} = \sum_{i=j}^{n} \binom{n}{i} \left\{ i t^{i-1}(1-t)^{n-i} - (n-i)t^i(1-t)^{n-i-1} \right\}$$

$$= \sum_{i=j}^{n} \left\{ n\binom{n-1}{i-1} t^{i-1}(1-t)^{n-i} - n\binom{n-1}{i} t^i (1-t)^{n-i-1} \right\}$$

$$= n\binom{n-1}{j-1} t^{j-1}(1-t)^{n-j}$$

اگر α و β به‌صورت $j = \alpha$ و $\beta = n - j + 1$ تعریف شود، رابطهٔ اخیر به شکل

$$\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} t^{\alpha-1}(1-t)^{\beta-1}, 0 \le t \le 1$$

در می‌آید که همان تابع چگالی بتاست.

اگر j دو مقدار ۱ و n را بپذیرد، توابع چگالی $R_{(1)}$ و $R_{(n)}$، به‌صورت زیر به‌دست می‌آید:

$$f_{R_{(1)}}(y_1) = n(1-y_1)^{n-1}, \quad 0 \le y_1 \le 1$$ (۸ـ۶ـض)

$$f_{R_{(n)}}(y_n) = n y_n^{n-1}, \quad 0 \le y_n \le 1$$ (۸ـ۷ـض)

به منظور تولید مقدار تصادفی از دو تابع چگالی فوق می‌توان از روش تبدیل معکوس استفاده کرد. اگر j مساوی یک باشد، مقدار تصادفی y_1 از رابطهٔ $y_1 = 1 - \sqrt[n]{1-r}$ و اگر $j = n$ باشد، مقدار تصادفی y_n از رابطهٔ $y_n = \sqrt[n]{r}$ به‌دست می‌آید. الگوریتم زیر نحوهٔ تولید مقادیر عمر سیستم موازی و زنجیره‌ای را نشان می‌دهد.

گام ۱. بر اساس روش تبدیل معکوس، مقدار تصادفی y_j را از تابع چگالی بتا با پارامترهای $\alpha = j$ و $\beta = n - j + 1$ تولید کنید ($j = 1$ یا n).

گام ۲. بر اساس روش تبدیل معکوس، یک مقدار تصادفی برای X از طریق رابطهٔ $x = F_X^{-1}(y_j)$ به‌دست آورید.

منطق الگوریتم فوق بر این اساس استوار است که در یک نمونهٔ تصادفی nتایی از عمر، مانند T_1، T_2، ...، T_i، ...، T_n، iامین عمر کوچک، نظیر iامین عدد تصادفی کوچک در دنبالهٔ R_1، R_2، ...، R_i، ...، R_n خواهد بود. پس، ابتدا از توزیع احتمال متغیر تصادفی $R_{(j)}$ مقدار تصادفی y_j تولید می‌شود و سپس با استفاده از y_j که در فاصلهٔ (۰، ۱) قرار دارد مقدار تصادفی x را تولید می‌کنیم.

۸ـ۳ـض روش ترکیب

این روش را می‌توان در مورد توزیعهای احتمال پیوسته و گسسته به‌کار گرفت. به منظور تشریح مطلب از یک مثال استفاده می‌کنیم.

■ مثال ۸-۱ ـض

احتمالات زیر در مورد تابع احتمال $b(5, 0/2)$ تا چهار رقم اعشار ارائه شده است:

i	۰	۱	۲	۳	۴	۵
p_i	۰/۳۲۷۷	۰/۴۰۹۶	۰/۲۰۴۸	۰/۰۵۱۲	۰/۰۰۶۴	۰/۰۰۰۳

اگر $p_۰$ را در نظر بگیریم، مقدار ۰٫۳۲۷۷ برای آن را می‌توان به‌صورت زیر بسط داد

$$p_۰ = ۰٫۳۲۷۷ = ۰٫۹\left(\frac{۳}{۹}\right) + ۰٫۰۷\left(\frac{۲}{۷}\right) + ۰٫۰۲۷\left(\frac{۷}{۲۷}\right) + ۰٫۰۰۳\left(\frac{۷}{۳۰}\right)$$

همچنین، $p_۱$ و $p_۲$ نیز به‌صورت زیر قابل عرضه است:

$$p_۱ = ۰٫۴۰۹۶ = ۰٫۹\left(\frac{۴}{۹}\right) + ۰٫۰۷\left(\frac{۰}{۷}\right) + ۰٫۰۲۷\left(\frac{۹}{۲۷}\right) + ۰٫۰۰۳\left(\frac{۶}{۳۰}\right)$$
$$p_۲ = ۰٫۲۰۴۸ = ۰٫۹\left(\frac{۲}{۹}\right) + ۰٫۰۷\left(\frac{۰}{۷}\right) + ۰٫۰۲۷\left(\frac{۴}{۲۷}\right) + ۰٫۰۰۳\left(\frac{۸}{۳۰}\right)$$

در حالت کلی، داریم

$$p_i = ۰٫۹t_{i۱} + ۰٫۰۷t_{i۲} + ۰٫۰۲۷t_{i۳} + ۰٫۰۰۳t_{i۴}, \quad i = ۰, ۱, \dots, ۵ \qquad (\text{۸-۸-ض})$$

به ازای مقادیر صفر، یک، ... ، پنج برای i، هر یک از مجموعه‌های $\{t_{i۱}\}$، $\{t_{i۲}\}$، $\{t_{i۳}\}$ و $\{t_{i۴}\}$ یک تابع احتمال را تشکیل می‌دهد. مثلاً $\{t_{i۱}\}$ را در نظر بگیرید:

$t_{۰۱} = ۳/۹$ یا ۰٫۳/۰٫۹ , $i = ۰$
$t_{۱۱} = ۴/۹$ یا ۰٫۴/۰٫۹ , $i = ۱$
$t_{۲۱} = ۲/۹$ یا ۰٫۲/۰٫۹ , $i = ۲$
$t_{۳۱} = t_{۴۱} = t_{۵۱} = ۰$ یا ۰٫۰/۰٫۹

از سوی دیگر، ضرایب t_{ij}ها در رابطهٔ (۸-۸-ض) به شرح زیر تعریف می‌شود

$۰٫۹ = (۱۰^{-۱})$ * (جمع اولین رقمهای اعشاری در p_iها)
$۰٫۰۷ = (۱۰^{-۲})$ * (جمع دومین رقمهای اعشاری در p_iها)
⋮

توضیحات بالا نحوهٔ ایجاد رابطهٔ (۸-۸-ض) را روشن می‌کند.

اینک، به منظور تولید مقدار تصادفی از $b(۵, ۰٫۲)$، به شرح زیر از رابطهٔ (۸-۸-ض) استفاده می‌کنیم.

الف) یک مقدار تصادفی برای متغیر تصادفی گسستهٔ Z با تابع احتمال زیر تولید کنید.

j	۱	۲	۳	۴
$P\{Z=j\}$	۰٫۹	۰٫۰۷	۰٫۰۲۷	۰٫۰۰۳

ب) اگر $Z = j$ شود، یک مقدار تصادفی از توزیع احتمال $\{t_{ij}\}$ ($j = ۱, ۲, ۳, ۴$) تولید کنید و آن را x بنامید. در نتیجه، می‌توان نوشت

$$P\{X = i\} = \sum_{j=۱}^{۴} P\{Z = j\} t_{ij} = p_i$$

به عبارت دیگر، x یک مقدار تصادفی از توزیع احتمال موردنظر، یعنی $b(۵, ۰٫۲)$ است. ■

چون در مسألهٔ بالا، هر یک از p_iها به چهار رقم اعشار گرد شده بود، نتایج به طور کامل دقیق نبود. علیرغم این مطلب، روش فوق را در مورد هر توزیع احتمال گسسته می‌توان به‌کار برد. گرچه در مثال بالا، عمل تولید مقدار تصادفی از تابع احتمال دوجمله‌ای را با عمل تولید مقدار تصادفی از توابع احتمال $\{t_{ij}\}$ و $P\{Z = j\}$، $i = ۰, ۱, \dots, ۵$، $j = ۱, ۲, ۳, ۴$ جانشین کردیم ولی در ۹۷ درصد از موارد به تولید مقدار تصادفی از $\{t_{i۱}\}$ و $\{t_{i۲}\}$ می‌پردازیم که خود شکل بسیار ساده‌ای دارد. از ایرادهای روش ترکیب این است که ناچار از ذخیره کردن توزیعهای احتمال $\{t_{ij}\}$ در حافظهٔ کامپیوتر هستیم. اینک، روش ترکیب را در مورد متغیرهای تصادفی پیوسته معرفی می‌کنیم.

تجزیهٔ یک تابع چگالی مانند $f_X(x)$ به طریق زیر به توابع چگالی دیگر کاری غیر عادی نیست:

$$f_X(x) = \alpha f_۱(x) + (۱ - \alpha) f_۲(x), \quad ۰ < \alpha < ۱ \qquad (\text{۸-۹-ض})$$

مثلاً در روانشناسی می‌توان به این امر برخورد کرد که هیستوگرم زمانهای واکنش انسان، درصدی از موارد (α) تمایل به رفتار عادی و در بقیهٔ موارد به سبب از دست رفتن تمرکز حواس تمایل به رفتار طبق حالتی غیرعادی را نشان دهد. به عبارت دیگر، زمانهای بروز واکنش با احتمال α طبق یک تابع چگالی مانند $f_۱$ و با احتمال $۱ - \alpha$ طبق تابع چگالی دیگری مانند $f_۲$ تعریف می‌شود.

مثال دیگر در مورد هیستوگرم مربوط به قد انسانهاست که ممکن است به خاطر آمیختن هیستوگرمهای قد زنان و مردان، دوکوهانه باشد هر چند که نمونه‌های گرفته شده از چنین آمیزه‌ای لزوماً حاکی از تشکیل تابع چگالی قد از مخلوط دو تابع چگالی قد زنان و مردان نباشد. در واقع، در بسیاری از موارد کاربردی در شبیه‌سازی، رابطهٔ (۸-۹-ض) صرفاً به منظور ساده کردن تحلیل مورد استفاده قرار می‌گیرد، حتی برای یک تابع چگالی یک کوهانه مانند تابع چگالی نرمال. امر ساده‌کردن تحلیل در صورتی تحقق می‌یابد که اولاً α به اندازهٔ کافی بزرگ باشد و ثانیاً، تولید مقدار

تصادفی از f_1 با سهولت زایدالوصفی نسبت به تولید مقدار تصادفی از f_X انجام پذیرد. اگر با احتمال α یک مقدار تصادفی از f_1 و با احتمال $1-\alpha$ یک مقدار تصادفی از f_2 تولید کنیم، در چارچوب رابطهٔ (۸ـ۹ـض) موفق شده‌ایم یک مقدار تصادفی برای f_X تولید کنیم؛ دلیل درست بودن این ادعا، صدق رابطهٔ زیر است

$$\begin{aligned} P\{X \le x\} &= \alpha P\{X \le x \mid \text{مقدار تصادفی از } f_1 \text{ تولید شود}\} \\ &\quad + (1-\alpha) P\{X \le x \mid \text{مقدار تصادفی از } f_2 \text{ تولید شود}\} \\ &= \alpha \int_{-\infty}^{x} f_1(t)dt + (1-\alpha)\int_{-\infty}^{x} f_2(t)dt \\ &= \int_{-\infty}^{x} \alpha f_1(t) + (1-\alpha) f_2(t)dt \\ &= \int_{-\infty}^{x} f_X(t)dt \end{aligned}$$

■ **مثال ۸-۲ ـض**

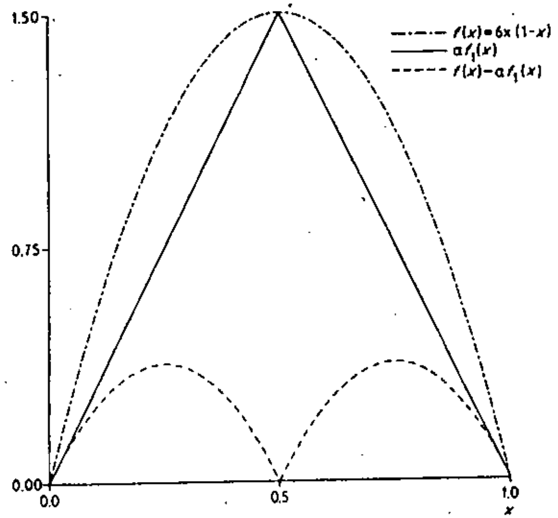
متغیر تصادفی X از توزیع احتمال بتا با تابع چگالی زیر برخوردار است:

$$f_X(x) = 6x(1-x), \quad 0 \le x \le 1$$

با استفاده از روشهای دیگر به سادگی می‌توان برای تابع چگالی فوق مقدار تصادفی تولید کرد، ولی در این مثال مایلیم این عمل را از طریق استفاده از روش ترکیب انجام دهیم. به این منظور، یک تابع چگالی مثلثی قرینه را به‌عنوان f_1 انتخاب می‌کنیم. چون تولید مقدار تصادفی از تابع چگالی مثلثی به راحتی میسر است، سعی داریم ضریب α را تا حد امکان بزرگ تعریف کنیم. چون پایهٔ مثلث تمام دامنهٔ متغیر تصادفی بتا را می‌پوشاند (یعنی ۱ =پایه)، ارتفاع آن در بدو امر مساوی ۲ واحد است. به عبارت دیگر، رأس مثلث در بالای رأس تابع بتا قرار دارد. در واقع، تابع چگالی مثلثی به‌صورت

$$f_1(x) = \begin{cases} 4x, & 0 \le x \le 1/2 \\ 4(1-x) & 1/2 \le x \le 1 \end{cases}$$

تعریف می‌شود.



شکل ۸ـ۲ـض کاربرد روش ترکیب در مورد تابع چگالی بتا.

با ضرب کردن f_1 در α، رأس مثلث را در داخل شکل تابع بتا قرار می‌دهیم. برای انجام این عمل، با رعایت جنبه‌های هندسی مسأله، به سادگی روشن می‌شود که بزرگترین مقدار α مساوی ۳/۴ است. پس از انتخاب تابع چگالی f_1 و ضریب کوچک‌کنندهٔ آن، α، تابع چگالی f_2 از طریق رابطهٔ

$$f_2(x) = \frac{f_X(x) - \alpha f_1(x)}{1-\alpha} \tag{\text{۸ـ۱۰ـض}}$$

به دست می‌آید. تابع چگالی f_2 برای مثال فوق به شرح زیر است:

$$f_2(x) = \begin{cases} 12x(1-2x), & 0 \le x \le 1/2 \\ 12(1-2x)(x-1), & 1/2 < x \le 1 \end{cases}$$

تولید مقدار تصادفی از f_2 تنها در ۲۵ درصد از موارد صورت می‌گیرد و انجام آن به‌عنوان یک تمرین در نظر گرفته شده است.

■

به‌طور کلی، تابع چگالی f_X را در نظر بگیرید و فرض کنید که تابع چگالی f_1 نیز تقریباً

شکلی مانند f_X دارد و عمل تولید مقدار تصادفی از آن با سهولت نسبی فراوان انجام می‌گیرد. به ازای مقداری برای α در محدودهٔ (۱ ,۰)، رابطهٔ زیر را بنویسید

$$f_X(x) = \alpha f_1(x) + (1-\alpha)\left(\frac{f_X(x)-\alpha f_1(x)}{1-\alpha}\right)$$

به منظور تولید مقدار تصادفی از f_X، با احتمال α از f_1 و با احتمال $1-\alpha$ از f_2 مقدار تصادفی تولید می‌شود. چون f_1 طوری انتخاب شده است که تولید مقدار تصادفی از آن ساده باشد، سعی کنید α تا حد امکان بزرگ باشد. محدودیتی که در مورد انتخاب مقدار α وجود دارد این است که باید رابطهٔ $f_X(x) - \alpha f_1(x) \geq 0$ برقرار باشد تا طبق رابطهٔ (۸-۱۰-ض) f_2 هم بتواند یک تابع چگالی محسوب شود. حال، اگر رابطهٔ

$$\alpha = \min_x \left(\frac{f_X(x)}{f_1(x)}\right) \qquad (۸\text{-}۱۱\text{-ض})$$

نوشته شود و برای نقطهٔ می‌نیمم بتوان یک مقدار مثبت یافت، رابطهٔ $f_X(x) - \alpha f_1(x) \geq 0$ برقرار می‌شود و دست‌کم یک مقدار از X به صدق رابطهٔ $\alpha f_1(x) = f_X(x)$ می‌انجامد. نقش α در این روش چیزی جز کوچک کردن f_1 به نحوی که در داخل f_X قرار گیرد نیست. اگر f_1 شباهت زیادی به f_X داشته باشد، ضریب کوچک‌کنندهٔ f_1 از لحاظ مقدار بزرگ خواهد بود.

روش ترکیب را می‌توان به نحوی تعمیم داد که در مورد تابع چگالی f_2 نیز قابل اعمال باشد. حاصل این عمل در n تکرار عبارت است از

$$f_X(x) = \sum_{i=1}^{n+1} \alpha_i f_i(x) \qquad (۸\text{-}۱۲\text{-ض})$$

به‌طوری که روابط

$$\sum_{i=1}^{n+1}\alpha_i = 1,\quad \alpha_i>0,\quad i=1,\cdots,n+1,$$

$$f_i(x)\geq 0,\quad \int_{a_i}^{b_i} f_i(x)dx = 1,\quad a_1=+\infty, b_n=+\infty$$

$$f_{n+1}(x) = \left(f_X(x) - \sum_{i=1}^{n}\alpha_i f_i(x)\right)\Big/\left(1-\sum_{i=1}^{n}\alpha_i\right) \qquad (۸\text{-}۱۳\text{-ض})$$

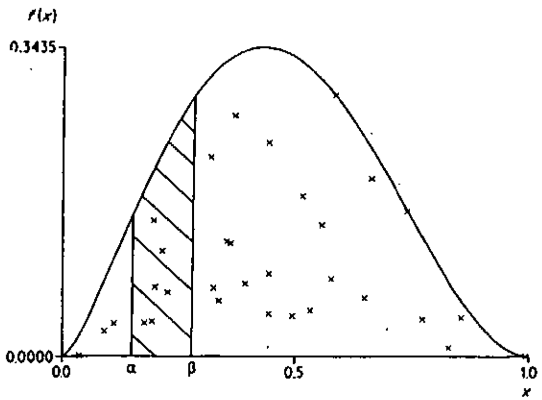
برقرار باشد. طبیعی است که مقادیر کوچک برای n ترجیح داده می‌شود. ولی از سوی دیگر، میزان سهولت نمونه‌گیری از تابع چگالی آخر، f_{n+1}، با احتمال α_{n+1} را نیز باید در نظر داشت.

در مثال ۸-۲-ض تمام توابع چگالی دامنهٔ محدودی داشت. در موردی که f_X دامنهٔ نامحدودی داشته باشد می‌توان f_1 را طوری برگزید که دامنه‌ای محدود داشته باشد. در چنین موردی، در بخشی از دامنهٔ x، روابط $f_1(x) = 0$ و $f_2(x) > 0$ به‌طور توأم برقرار است. به علاوه، در مواردی نیز دو رابطهٔ $f_1(x) > 0$ و $f_2(x) = 0$ ممکن است برقرار باشد.

۸-۴-ض روش رد و قبول

این روش مبتنی بر ادامهٔ نمونه‌گیری تا تحقق شرط خاصی است. با قبول چنین تعریفی، می‌توان روش رد و قبول را در مورد توزیعهای پیوسته و گسسته به‌کار برد. در این زیر بخش، نحوهٔ اعمال روش رد و قبول برای تولید مقدار تصادفی از توزیعهای پیوسته را تشریح می‌کنیم.

چنین فرض کنید که با در اختیار داشتن روشی می‌توانیم نقاطی در زیر هر نوع تابع چگالی به‌طور یکدست و یکنواخت تولید کنیم. شکل ۸-۳-ض را در نظر بگیرید. احتمال این امر که مختصهٔ افقی هر یک از نقاط مورد بحث در فاصلهٔ (α,β) قرار گیرد چقدر است؟



شکل ۸-۳-ض تولید یکنواخت نقاط در ناحیهٔ زیر تابع چگالی B(۲/۵, ۳).

پیشامد $\alpha \leq X < \beta$ معادل پیشامد قرارگرفتن نقطهٔ تولیدشده در ناحیهٔ سایه‌خورده است و چون فرض بر یکنواخت بودن توزیع نقاط است، احتمال نظیر پیشامد اخیر از رابطهٔ زیر به‌دست می‌آید:

$$P\{\alpha \leq X < \beta\} = \frac{\text{مساحت ناحیهٔ سایه خورده}}{\text{مساحت ناحیهٔ زیر منحنی } f_X}$$

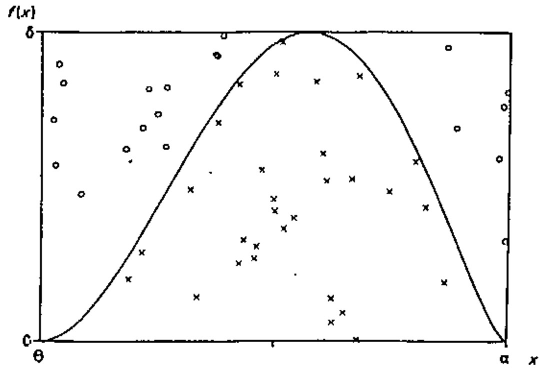
رابطهٔ بالا را می‌توان به‌صورت زیر نوشت:

$$\frac{\int_\alpha^\beta f_X(x)dx}{\int_۰^۱ f_X(x)dx} = \int_\alpha^\beta f_X(x)dx = P\{\alpha \leq X < \beta\}, \quad \alpha < \beta$$

با دراختیار داشتن روشی برای تولید نقاط تصادفی به‌طور یکنواخت در ناحیهٔ زیر هر تابع چگالی قادر خواهیم بود که از توابع چگالی مختلف مقدار تصادفی تولید کنیم. برای تولید مقدار تصادفی در روش رد و قبول می‌توان از مونت‌کارلو استفاده کرد.

در مورد کاربرد روش مونت‌کارلو برای تقریب زدن مساحت یک ناحیه، مساحت مجهول را باید در ناحیه‌ای با مساحت معلوم محاط کرد. اگر برای سهولت ارائهٔ مطلب، تابع چگالی بتا را محور بحث قرار دهیم، به راحتی خواهیم توانست آن را در یک چهار ضلعی مانند شکل ۸ـ۴ـض محاط کنیم.

در شکل ۸ـ۴ـض، دامنهٔ تغییرات X با (α, β) مشخص شده است. به منظور تولید یکنواخت اعداد تصادفی در داخل مستطیل فوق، کافی است که به‌طور یکنواخت به تولید مقادیر تصادفی برای دو مختصهٔ نقطهٔ $P(\alpha + (\beta - \alpha)r_۱, \theta r_۲)$ اقدام کنیم. بدیهی است که $r_۱$ و $r_۲$ معرف دو عدد تصادفی است که باید به‌طور مستقل تولید شود. از میان نقاط تولید شده، آنها را که در



شکل ۸ـ۴ـض تولید یکنواخت نقاط تصادفی محاط در مستطیلی به مساحت $\theta(\beta - \alpha)$.

بالای تابع چگالی قرار می‌گیرد مردود اعلام می‌کنیم و مختصهٔ افقی بقیهٔ نقاط را به‌عنوان مقادیر تصادفی از توزیع احتمال X می‌پذیریم.

چون مساحت مستطیل شکل فوق معادل $\theta(\beta - \alpha)$ و مساحت زیر تابع چگالی f_X مساوی با واحد است، احتمال مورد پذیرش قرارگرفتن یک نقطه (یا احتمال موفقیت) مساوی $۱/[\theta(\beta - \alpha)]$ است. بنابراین، هر چه θ کوچکتر باشد، احتمال موفقیت و در نتیجه کارایی روش بیشتر می‌شود.

روشی که در بالا معرفی شد دو عیب دارد. اولاً، استفاده مستطیل به‌عنوان ناحیهٔ محیطی، علیرغم سادگی ساختار آن از لحاظ تولید یکنواخت نقاط تصادفی، ایجادکنندهٔ این محدودیت است که دامنهٔ تغییرات متغیر تصادفی X باید محدود باشد. به عبارت دیگر، چون تنها می‌توانیم مقادیر تصادفی از تابع یکنواخت با دامنهٔ محدود تولید کنیم،در حالتی که دامنهٔ X نامحدود است، استفاده از ناحیهٔ محیطی مستطیل میسر نیست. ثانیاً، احتمال مردود اعلام کردن هر نقطه ممکن است قابل‌توجه باشد. این مطلب در حالتی مصداق پیدا می‌کند که تابع چگالی یک کوهانهٔ f_X رأسی باریک و تیز داشته باشد.

دو عیب بالا را می‌توان با انتخاب یک تابع چگالی دوم که سطح زیرآن نقش ناحیهٔ محیطی را بازی کند مرتفع کرد. برای روشن شدن مطلب، فرض کنید که تابع چگالی دوم با h_Y نمادگذاری شود و علاوه بر داشتن همان دامنهٔ X به راحتی هم بتوان از آن مقدار تصادفی تولید کرد. در چنین شرایطی می‌توان به راحتی نقاطی مانند $P(x, y)$ را به‌طور یکنواخت در زیر منحنی h_Y تولید کرد. دو مختصهٔ x و y باید به نحوی تولید شود که x طبق توزیع احتمال h_Y تعریف شود و توزیع شرطی مختصهٔ دوم در صورت حصول مقدار x برای مختصهٔ اول، در محدودهٔ $[۰, h_Y(x)]$ یکنواخت باشد. (به شکل ۸ـ۵ـض نگاه کنید.)

در صورت امکان، تابع چگالی h_Y را باید به نحوی انتخاب کرد که شکلی شبیه به f_X داشته باشد. هر چند در اغلب موارد می‌توان h_Y را شبیه به f_X انتخاب کرد ولی محاط کردن f_X در ناحیهٔ زیر h_Y به‌طوری که به ازای همهٔ مقادیر x رابطهٔ $f_X(x) \leq h_Y(x)$ برقرار باشد، ناممکن است. مطلب اخیر را چنین می‌توان توضیح داد که از نظر عملی، دو تابع f_X و h_Y باید متفاوت باشد و چون سطح زیر هر دو معادل واحد است، عدم امکان برقراری نامساوی بالا به ازای تمام مقادیر x تضمین می‌شود. به منظور رفع این نقص، تابع چگالی h_Y را در ضریبی مانند k $(k > ۱)$ ضرب می‌کنند. برای انجام این عمل، مقدار k طوری انتخاب می‌شود که تابع f_X در kh_Y محاط شود. به این ترتیب، با انتخاب h_Y و k به طریق مناسب و نمادگذاری $kh_Y(x)$ با $g(x)$، الگوریتم زیر را می‌توان عرضه کرد (به شکل ۸ـ۵ـض رجوع کنید):

گام ۱. یک مقدار تصادفی مانند x از تابع چگالی h_Y تولید کنید.

گام ۲. بر اساس تابع چگالی یکنواخت در فاصلهٔ $[۰, g(x)]$، یک مقدار تصادفی مانند y برای مختصهٔ عمودی نقطهٔ P تولید کنید.

گام ۳. اگر رابطهٔ $y < f_X(x)$ برقرار است، مقدار x را به‌عنوان یک مقدار تصادفی از تابع

چگالی f_X قبول کنید.

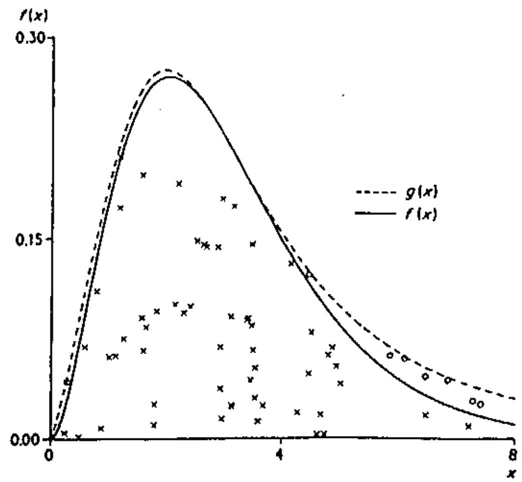
در برخورد اول، ممکن است الگوریتم فوق قدری مبهم و غیرعادی جلوه کند زیرا در گام ۳ مقدار تولیدشده برای مختصهٔ عمودی، Y، مورد آزمایش قرار می‌گیرد ولی در نهایت، مقدار تولیدشده برای مختصهٔ افقی، x، پذیرفته می‌شود. به هر صورت، بر اساس مطالب فوق، گام ۳ باید منعکس‌کنندهٔ این مطلب باشد که توزیع تصادفی و یکنواخت نقاط در زیر تابع $g(x)$ را می‌توان به‌عنوان توزیع تصادفی و یکنواخت نقاط در زیر تابع چگالی f_X نیز تعبیر کرد.

احتمال عدم موفقیت (احتمال مردود شدن یک نقطه) از رابطهٔ زیر به‌دست می‌آید که در آن

$$\int_{-\infty}^{\infty}[g(x) - f_X(x)]dx \Big/ \int_{-\infty}^{\infty} g(x)dx = 1 - \frac{1}{k} \qquad (\text{۸ـ۱۴ـض})$$

اهمیت انتخاب مقادیر کوچک برای k ($k > 1$) مشهود است.

اینک، به ارائهٔ دو مثال در زمینهٔ استفاده از روش رد و قبول می‌پردازیم. در مثال اول، به منظور تولید مقدار تصادفی از تابع چگالی نرمال، یک تابع نمایی منفی را به‌عنوان تابع محیطی انتخاب می‌کنیم. چون تابع چگالی نمایی منفی تنها می‌تواند نیمی از تابع چگالی



شکل ۸ـ۵ـض محاط شدن $f_X(x)$ در $g(x)$.

نرمال استاندارد را دربرگیرد، فقط از نیمهٔ راست نرمال استفاده می‌کنیم. اگر تابع چگالی نرمال استاندارد با $\varphi(x)$ مشخص شود، f_X را به‌صورت $f_X(x) = 2\varphi(x)$ تعریف می‌کنیم و برای تولید مقدار تصادفی از $\varphi(x)$ ابتدا از تابع چگالی $f_X(x)$ ($x \geq 0$) مقدار تصادفی تولید می‌کنیم.

■ مثال ۸-۳ـض یک روش رد و قبول برای تولید مقدار تصادفی از نرمال صفر-یک

در این مثال، f_X و $g(x)$ به شرح زیر تعریف می‌شود:

$$f_X(x) = \sqrt{\frac{2}{\pi}}\, e^{-\frac{x^2}{2}}, \quad x \geq 0$$

$$g(x) = ke^{-x}, \quad x \geq 0$$

به منظور یافتن مقدار k می‌توان $f_X(x)$ را با $g(x)$ مساوی قرار داد و به جستجوی مقادیری برای x پرداخت که تساوی مزبور را ممکن می‌سازد. انجام این کار برای مثال مورد بررسی، به تعریف معادلهٔ

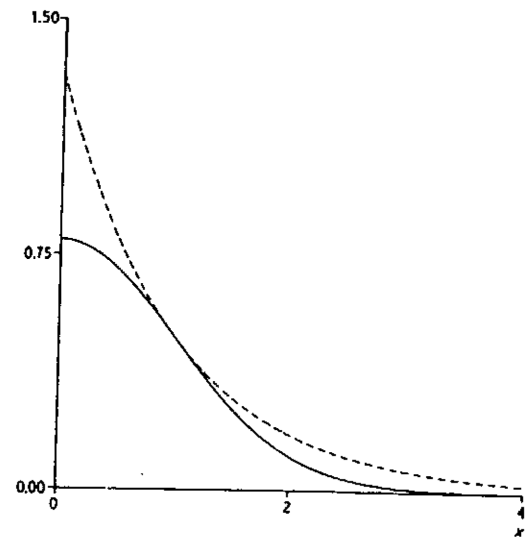
$$ke^{-x} = \sqrt{\frac{2}{\pi}}\, e^{-x^2/2}, x \geq 0$$

می‌انجامد. اگر معادلهٔ اخیر فاقد ریشهٔ حقیقی باشد، بدین معنی خواهد بود که مقدار k بیش از حد بزرگ انتخاب شده است. از سوی دیگر، انتخاب مقداری بیش از حد کوچک برای k، به کسب دو ریشهٔ حقیقی از معادلهٔ فوق می‌انجامد. به‌دست آوردن دو ریشهٔ حقیقی و یکسان از معادلهٔ بالا به معنی تعریف k در مناسبترین (کوچکترین) مقدار خود است. به عبارت دیگر، در چنین حالتی، دو منحنی f_X و $g(x)$ مطابق شکل ۸ـ۶ـض در یک نقطه بر هم مماس می‌شود. اینک، به حل معادلهٔ فوق و تعیین مقدار k می‌پردازیم:

$$ke^{-x} = \sqrt{\frac{2}{\pi}}\, e^{-\frac{x^2}{2}}$$

$$k\sqrt{\frac{\pi}{2}} = e^{x-x^2/2}$$

$$x^2 - 2x + 2\ln\left(k\sqrt{\frac{\pi}{2}}\right) = 0$$



شکل ۸ـ۶ـض رابطهٔ $g(x)$ با f_X به ازای کوچکترین مقدار k.

شرط لازم و کافی برای وجود دو ریشهٔ حقیقی و یکسان از معادلهٔ اخیر، صدق رابطهٔ زیر است:

$$۱ = ۲\ell n\left(k\sqrt{\pi/۲}\right)$$

این رابطه را می‌توان به‌صورت $k^۲(\frac{\pi}{۲}) = e$ نوشت. متعاقباً، مقدار k به شرح زیر تعیین می‌شود:

$$k = +\sqrt{\frac{۲e}{\pi}} \simeq ۱/۳۱۵۴۸۹۲$$

مقدار فوق برای k، نظیر مقدار ۱ برای x است. تذکر داده می‌شود که تابع f_X یک نقطهٔ عطف به ازای $x = ۱$ دارد.

بر اساس مطالب فوق، الگوریتم زیر را می‌توان ارائه کرد:

گام ۱. یک مقدار تصادفی برای تابع چگالی e^{-x} تولید کنید. برای این منظور، یک عدد تصادفی مانند $r_۱$ تولید کنید و x را از رابطهٔ $x = -\ell n r_۱$ به‌دست آورید.

گام ۲. یک مقدار تصادفی از تابع یکنواخت در فاصلهٔ $(۰, ke^{-x})$ تولید کنید. برای این منظور، یک عدد تصادفی مانند $r_۲$ را مستقل از $r_۱$ تولید کنید و y را از رابطهٔ $y = ke^{-x}r_۲$ یا $y = kr_۱r_۲$ به‌دست آورید.

گام ۳. شرط لازم و کافی برای پذیرفتن x، صدق رابطهٔ

$$y < \sqrt{\frac{۲}{\pi}}e^{-\frac{x^۲}{۲}}$$

یا

$$kr_۱r_۲ < \sqrt{\frac{۲}{\pi}}e^{-\frac{x^۲}{۲}}$$

و یا

$$r_۱r_۲ < e^{-\frac{۱}{۲}(۱+x^۲)}$$

است زیرا رابطهٔ $k = \sqrt{۲e/\pi}$ برقرار است.

دیده می‌شود که الگوریتم فوق به‌طور مستقیم به k بستگی ندارد. در چارچوب این مثال، باید به طرح این نکته پرداخت که چگونه می‌توان مقدار تصادفی تولید شده، یعنی x، را به یک مقدار تصادفی برای توزیع نرمال صفرـیک تبدیل کرد. به منظور تعیین علامت مثبت یا منفی برای x، می‌توان یک عدد تصادفی دیگر تولید کرد و برحسب بزرگتر از $\frac{۱}{۲}$ بودن یا نبودن آن، به x علامت، به ترتیب، مثبت یا منفی داد. طریق دیگر انجام این کار، توجه بدین نکته است که مختصهٔ عمودی نقطهٔ P در فاصلهٔ$(۰, g(x))$ توزیع احتمال یکنواخت دارد و اگر مختصهٔ عمودی از $f_X(x)$ تجاوز نکند در فاصلهٔ $(۰, f_X(x))$ نیز توزیع احتمال یکنواخت خواهد داشت. در چنین شرایطی، اگر مقدار به‌دست آمده برای مختصهٔ عمودی از $\varphi(x)$ تجاوز نکند می‌توان به x علامت منفی و در غیر این صورت به آن علامت مثبت داد. ■

دلیل انتخاب e^{-x} به‌عنوان هستهٔ اصلی تابع محیطی به‌جای شکل کلی $\lambda e^{-\lambda x}$ نیز چنین است که با انتخاب مقدار ۱ برای λ، احتمال مردود اعلام کردن نقاط تولیدشده را به کمترین میزان می‌رسانیم. درستی این ادعا را می‌توان با یک تمرین ساده به اثبات رسانید.

دلیل مناسب بودن تابع e^{-x} برای دربرگرفتن تابع $f_X(x) = ۲\varphi(x)$، این واقعیت است که با میل x به سمت بینهایت، آهنگ میل e^{-x} به سمت صفر از آهنگ میل f_X به سمت صفر کندتر است.

یک روش کلی برای تعیین k این است که چون به ازای تمام مقادیر x باید بدون منطبق شدن $kh_Y(x)$ بر $f_X(x)$، رابطهٔ $kh_Y(x) \geq f_X(x)$ برقرار باشد، k به‌صورت

$$k = \max_x \left(\frac{f_X(x)}{h_Y(x)}\right) \qquad (۸\text{ـ}۱۵\text{ـض})$$

تعریف شود. از تعریف k به این صورت چنین برمی‌آید که مقدار k باید متناهی باشد

www.matlabdl.com

و به ازای تمام مقادیر x رابطهٔ $kh_Y(x) \geq f_X(x)$ و به ازای دست‌کم یک مقدار x رابطهٔ $kh_Y(x) = f_X(x)$ برقرار باشد. (نقش α در روش ترکیب را با نقش k در روش رد و قبول همراه با دو رابطهٔ (۸ـ۱۱ـض) و (۸ـ۱۵ـض) مقایسه کنید.) اگر انتخاب تابع محیطی h_Y به طرز مناسبی صورت نگیرد، مقداری نامتناهی برای k به‌دست خواهد آمد.

■ مثال ۸-۴ ـض

مقدار k را با استفاده از رابطهٔ (۸ـ۱۵ـض) برای مثال ۸ـ۳ـض به‌دست آورید.

$$f_X(x)/h_Y(x) = \sqrt{\frac{2}{\pi}}e^{-x^2/2+x},$$

$$y = x - \frac{x^2}{2} + \ell n\sqrt{\frac{2}{\pi}},$$

$$\frac{dy}{dx} = 1 - x, \quad x = 1,$$

$$\frac{d^2y}{dx^2} = -1$$

در نتیجه، با انتخاب مقدار ۱ برای x، تابع $f_X(x)/h_Y(x)$ ماکسیمم می‌شود و مانند مثال قبل، مقدار $\sqrt{2e/\pi}$ برای k به‌دست می‌آید. ■

می‌دانیم که تابع چگالی ارلنگ حالت خاصی از تابع چگالی گاماست. در واقع، در صورتی که پارامتر شکل در تابع چگالی گاما یک عدد صحیح و مثبت باشد، تابع ارلنگ به‌دست می‌آید. اینک، با استفاده از روشی که توسط فیشمن ابداع شده است، تولید مقدار تصادفی از تابع چگالی ارلنگ را مورد بررسی قرار می‌دهیم.

■ مثال ۸-۵ ـض

در این مثال، تابع چگالی f_X به‌صورت

$$f_X(x) = \frac{x^{n-1}e^{-x}}{\Gamma(n)}, \quad n > 1 \ , \quad x \geq 0$$

تعریف می‌شود. هستهٔ اصلی تابع محیطی را به‌صورت زیر انتخاب می‌کنیم:

$$h_Y(x) = \frac{1}{n}e^{-\frac{x}{n}}, \quad x \geq 0$$

چون با میل x به سمت بینهایت، f_X سریعتر از $g(x) = kh_Y(x)$ به صفر میل می‌کند، $g(x)$ به‌عنوان تابع محیطی انتخاب مناسبی به‌شمار می‌آید. مجدداً، y را به‌صورت $y = f_X(x)/h_Y(x)$ تعریف و آن را ماکسیمم می‌کنیم تا مقادیر زیر برای x و متعاقباً k به‌دست آید:

$$\ell n y = (n-1)\ell n x - x + \frac{x}{n} + \ell n\frac{n}{\Gamma(n)},$$

$$\frac{d}{dx}\ell n y = \frac{n-1}{x} - 1 + \frac{1}{n},$$

$$\frac{n-1}{x} - 1 + \frac{1}{n} = 0, \quad x = n,$$

$$\frac{d^2}{dx^2}\ell n y = \frac{1-n}{x^2} < 0, \quad n > 1,$$

$$k = \frac{n^n e^{1-n}}{\Gamma(n)}$$

بر اساس نتایج فوق، الگوریتم زیر ارائه می‌شود:

گام ۱. با استفاده از عدد تصادفی r_1، مقدار تصادفی $x = -n\ell n r_1$ را از h_Y تولید کنید.

گام ۲. r_2 را مستقل از r_1 تولید کنید و از تابع چگالی یکنواخت در فاصلهٔ $(0, g(x))$، مقدار تصادفی $y = r_2 g(x) = kr_2\frac{1}{n}e^{-x/n}$ یا $y = kr_1r_2/n$ را تولید کنید.

گام ۳. شرط لازم وکافی برای قبول x به‌عنوان یک مقدار تصادفی از تابع چگالی f_X، صدق رابطهٔ

$$y < \frac{x^{n-1}e^{-x}}{\Gamma(n)}$$

یا

$$kr_1r_2/n < \frac{x^{n-1}e^{-x}}{\Gamma(n)}$$

یا

$$\frac{n^n e^{1-n}}{\Gamma(n)}\frac{r_1r_2}{n} < \frac{x^{n-1}e^{-x}}{\Gamma(n)}$$

و سرانجام، $r_1r_2(\frac{n}{e})^{n-1} < x^{n-1}e^{-x}$ است. ■

روش رد و قبول را به طریق بسیار کوتاهتری نیز می‌توان توضیح داد. برای این منظور قضیهٔ

زیر عرضه می‌شود:

قضیه. تابع چگالی متغیر تصادفی X را به‌صورت زیر در نظر بگیرید

$$f_X(x) = kl(x)h_Y(x) \quad , \quad a \le x \le b \qquad (۸\text{-}۱۶\text{-ض})$$

که $1 \le k < \infty$، $0 < l(x) \le 1$، $0 \le h_Y(x) < \infty$، و $\int_a^b h_Y(x)dx = 1$ باشد. فرض کنید Y متغیری تصادفی با تابع چگالی $h_Y(x)$ در فاصلهٔ $a \le x \le b$ باشد. به‌علاوه، فرض کنید که پیشامد S نیز نشان‌دهندهٔ صدق رابطهٔ $R \le l(Y)$ باشد ($R \sim U[0,1]$). در نتیجه، اگر S رخ دهد، X و Y هم‌توزیع خواهند بود.

اثبات. به منظور اثبات قضیهٔ فوق، pdf توأم متغیر تصادفی گسستهٔ S و متغیر تصادفی پیوستهٔ Y را با $P\{S, Y = x\}$، تابع احتمال شرطی $S|Y = x$ را با $P\{S|Y = x\}$، تابع چگالی شرطی $Y|S$ را با $h_{Y|S}(x|S)$، تابع احتمال حاشیه‌ای S را با $P\{S\}$ و سرانجام تابع چگالی حاشیه‌ای Y را مانند رابطهٔ (۸-۱۶-ض) با h_Y نمادگذاری می‌کنیم. در این صورت، روابط زیر برقرار است:

$$P\{S|Y = x\}h_Y(x) = h_{Y|S}(x|S)P\{S\} = P\{S, Y = x\} \qquad (۸\text{-}۱۷\text{-ض})$$

$$\begin{aligned} P\{S|Y = x\} &= P\{R \le l(Y)|Y = x\} \\ &= P\{\int_0^{l(Y)} dr|Y = x\} = l(x) \end{aligned} \qquad (۸\text{-}۱۸\text{-ض})$$

$$\begin{aligned} \int_a^b kl(x)h_Y(x)dx &= k\int_a^b P\{S|Y = x\}h_Y(x)dx \\ &= k\int_a^b \frac{P\{S, Y = x\}}{h_Y(x)}h_Y(x)dx \\ &= k\int_a^b P\{S, Y = x\}dx = kP\{S\} = 1 \end{aligned}$$

$$P\{S\} = 1/k \qquad (۸\text{-}۱۹\text{-ض})$$

$$\begin{aligned} f_X(x) &= kP\{S|Y = x\}h_Y(x) = kP\{S, Y = x\} \\ &= kP\{S\}h_{Y|S}(x|S) = k\frac{1}{k}h_{Y|S}(x|S) \end{aligned}$$

$$f_X(x) = h_{Y|S}(x|S) \qquad (۸\text{-}۲۰\text{-ض})$$

به موجب رابطهٔ (۸-۱۹-ض)، احتمال موفقیت در هر آزمایش مساوی $1/k$ است ($k > 1$).

پس، تا می‌توان باید مقداری کوچک برای k انتخاب کرد تا احتمال موفقیت در هر آزمایش بیشتر شود. از دو رابطهٔ (۸-۱۴-ض) و (۸-۱۹-ض) نتیجهٔ واحدی در این زمینه کسب می‌شود. چون آزمایشهای مورد بحث از هم مستقل است، احتمال موفقیت در آزمایش jام از تابع احتمال هندسی

$$(1/k)(1 - 1/k)^{j-1}, \quad j = 1, 2, \cdots \qquad (۸\text{-}۲۱\text{-ض})$$

با میانگین k به‌دست می‌آید. به هنگام ارزیابی هر الگوریتم رد و قبول، بررسی k از امور لازم است. به موجب رابطهٔ (۸-۲۰-ض)، در صورت رخداد پیشامد S، دو متغیر تصادفی Y و X هم توزیع خواهد بود و مقدار تصادفی تولیدشده از تابع چگالی h_Y را می‌توان به‌عنوان یک مقدار تصادفی از تابع چگالی f_X تلقی کرد. بر اساس نتایج فوق، الگوریتم زیر ارائه می‌شود:

گام ۱. مقدار تصادفی y را از تابع چگالی h_Y تولید کنید.

گام ۲. عدد تصادفی r را از $U[0,1]$ تولید کنید.

گام ۳. اگر رابطهٔ $r \le l(y)$ برقرار است، $x \leftarrow y$.

اینک، با استفاده از نتایج فوق، مجدداً مثال ۸-۴-ض را بررسی می‌کنیم.

■ **مثال ۸-۶-ض**

نحوهٔ تولید مقدار تصادفی از تابع چگالی $f_X(x) = \sqrt{\frac{2}{\pi}}e^{-\frac{x^2}{2}}, x \ge 0$ را در قالب نتایج قضیهٔ بالا توضیح دهید.

برای شروع، f_X را به‌صورت زیر می‌نویسیم:

$$f_X(x) = \sqrt{\frac{2e}{\pi}}e^{-\frac{(x-1)^2}{2}}e^{-x}, \quad x \ge 0$$

می‌بینیم که روابط زیر برقرار است:

$$k = \sqrt{\frac{2e}{\pi}}, \quad l(x) = e^{-\frac{1}{2}(x-1)^2}, \quad h_Y(x) = e^{-x}$$

سپس، از تابع چگالی نمایی منفی با میانگین ۱، مقدار تصادفی y را تولید می‌کنیم. در این صورت عدد تصادفی r تولید می‌شود؛ اگر رابطهٔ $r \le e^{-\frac{1}{2}(y-1)^2}$ برقرار باشد، y را به‌عنوان یک مقدار تصادفی از f_X می‌پذیریم.

در این مثال، احتمال موفقیت تقریباً مساوی ۰٫۷۶ است. به عبارت دیگر، امید ریاضی تعداد آزمایشها برای حصول اولین موفقیت، تقریباً، معادل ۱٫۳۲ است.

■

در مورد روش رد و قبول که به وسیلهٔ رابطهٔ (۸ـ۱۶ـض) معرفی شد، باید به این نکته توجه داشت که برای تابع چگالی معینی مانند f_X، ممکن است بتوان رابطهٔ (۸ـ۱۶ـض) را به بیش از یک شکل راه‌اندازی کرد. در این صورت، تحت شرایط مساوی، شکلی که دارای کوچکترین k است ارجح شمرده می‌شود.

قسمت چهارم

تحلیل داده‌های شبیه‌سازی