



دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)

دانشکده مهندسی صنایع و سیستم‌های مدیریت

تمرین یکم درس اصول شبیه‌سازی

اعضا:

حامد اعراب - ۹۹۲۵۰۰۳

شهریار خلوتی - ۹۹۲۵۰۱۵

استاد: دکتر عباس احمدی

تدریس‌یار: مهدی محمدی

بهار ۱۴۰۳

فهرست

مسائل	۱
مسئله یکم	۱
مسئله دوم	۱
مقدمه	۲
چارچوبی برای شبیه‌سازی رویدادهای گسسته	۲
کنترلر	۳
رویداد	۵
مدل‌سازی و حل مسئله‌ها	۷
حالت کلی: چند خدمت‌دهنده	۷
موجودیت‌های مختص به این سیستم	۷
کنترلر	۷
رویدادها	۹
حالت خاص: دو خدمت‌دهنده	۱۱
نتایج	۱۲
مسئله یکم	۱۲
مسئله دوم	۱۳
نتیجه‌گیری	۱۴
جدول گزارش عملکرد اعضای گروه	۱۴

مسائل

مسئله یکم

آریا و بردیا دو خدمت‌دهنده در یک رستوران خودرویی هستند. فاصله میان ورود هر دو خودرو یکنواخت و در بازه ۳ تا ۶ دقیقه است. زمان خدمت آریا و بردیا به هر خودرو، یکنواخت و در بازه ۹ تا ۱۲ دقیقه است.

- (۱) نمودارهای جریان را برای کنترلر و رویدادهای شبیه‌سازی بکشید.
- (۲) سیستم را برای ۹ ساعت شبیه‌سازی کنید. در حالت اول، هر بار نخستین خدمت‌دهنده در دسترس را برای خدمت‌دهی، و در حالت دوم، یک خدمت‌دهنده در دسترس را تصادفی انتخاب کنید.
- (۳) میزان به‌کارگیری (درصد زمان اشتغال) آریا و بردیا را در هر حالت تخمین بزنید.
- (۴) میانگین زمان انتظار هر مشتری را در هر حالت به‌دست آورید.
- (۵) در هر حالت چند نفر وارد صف انتظار می‌شوند؟

مسئله دوم

در یک سیستم خدماتی با چندین ایستگاه موازی (I خدمت‌دهنده)، زمان میان ورود هر دو مشتری یکنواخت و در بازه ۰ تا ۴ دقیقه، و زمان خدمت هر خدمت‌دهنده به هر مشتری، یکنواخت و در بازه ۶ تا ۱۰ دقیقه است.

- (۱) نمودارهای جریان را برای کنترلر و رویدادهای شبیه‌سازی بکشید.
- (۲) سیستم را برای ۸ ساعت شبیه‌سازی کنید.
- (۳) تعداد بهینه خدمت‌دهنده‌ها را به‌دست آورید، به طوری که میانگین زمان انتظار کمینه و درصد زمان اشتغال هر خدمت‌دهنده بیشینه شود.

مقدمه

مسئله‌هایی که پیش‌تر آمده‌اند، در بنیان خود، شبیه‌سازی رویدادهای گسسته^۱ را بیان می‌کنند. در مسئله نخست، سیستمی داریم با یک صف^۲ و دو خدمت‌دهنده^۳ که در کنار هم به مشتریان^۴ خدمت می‌دهند. رویدادهای^۵ سیستم نیز به ورود^۶ و خروج^۷ مشتریان محدود می‌شوند. در مسئله دوم، حالت کلی مسئله پیشین را داریم، یک صف، چندین خدمت‌دهنده موازی^۸، و همان رویدادها. حال، باید به این نکته نیز بپردازیم که علی‌رغم شباهت سیستم‌های این دو مسئله، خواسته‌های هر مسئله متفاوت است. در مسئله نخست، باید سیستم را با دو روش متفاوت در انتخاب خدمت‌دهنده در دسترس^۹ شبیه‌سازی کنیم. سپس با به‌دست آوردن و مقایسه چند معیار کارایی^{۱۰}، مانند درصد زمان اشتغال^{۱۱} خدمت‌دهنده‌ها، روش برتر را بیابیم. اما در مسئله دوم، صرفاً باید تعداد بهینه‌ی خدمت‌دهنده‌ها را با توجه به مقادیر داده‌شده برای زمان‌های میان هر رویداد بیابیم، به طوری که میانگین زمان انتظار کمینه^{۱۲} و درصد زمان اشتغال هر خدمت‌دهنده بیشینه^{۱۳} شود.

چارچوبی برای شبیه‌سازی رویدادهای گسسته

حال، با توجه به این که (۱) هر دو مسئله، شبیه‌سازی رویدادهای گسسته را در خود گنجانده‌اند و (۲) احتمالاً در تمرین‌های آینده نیز با این نوع مسائل مواجه خواهیم شد، تصمیم گرفتیم که یک چارچوب^{۱۴} و بنیان واحد برای حل آن‌ها طراحی و پیاده کنیم. این چارچوب باید ویژگی‌های زیر را داشته باشد:

¹ Discrete-Event Simulation

² Queue

³ Server

⁴ Customers

⁵ Events

⁶ Arrival

⁷ Departure

⁸ Parallel

⁹ Available

¹⁰ Performance Metrics

¹¹ Utilization Time Percentage (UTP)

¹² Minimized

¹³ Maximized

¹⁴ Framework

چارچوبی برای شبیه‌سازی رویدادهای گسسته

(۱) گسترش‌پذیری^{۱۵}: در مسائل گوناگون با سیستم‌هایی سروکار داریم که متغیرها^{۱۶}، وضعیت^{۱۷}، منطق^{۱۸}، و رفتار^{۱۹} مخصوص خود را دارند. بنابراین، به چارچوبی نیاز داریم که حداقل امکانات شبیه‌سازی را فراهم کرده و اجازه‌ی افزودن سایر اجزای مورد نیاز را بدهد.

(۲) مقیاس‌پذیری^{۲۰}: مسائل شبیه‌سازی در پیچیدگی^{۲۱} و اندازه با هم تفاوت دارند. مثلاً ممکن است تعداد صف‌ها، انواع و مراحل خدمت‌دهی، و تعداد خدمت‌دهنده‌ها در هر مسئله کم و یا زیاد باشند. بنابراین، چارچوب مورد استفاده باید قابلیت به‌کارگیری را در مقیاس‌های مختلف داشته باشد.

(۳) بهره‌وری^{۲۲}: چارچوب ما باید در اجرای فرآیند شبیه‌سازی سریع بوده و مصرف منابع کمی داشته باشد. این منابع شامل انرژی، حافظه، و فضای ذخیره‌سازی‌اند. با داشتن بهره‌وری مناسب می‌توان یک سیستم را به دفعات بیش‌تری شبیه‌سازی کنیم و از صحت خروجی‌ها و درستی مدل شبیه‌سازی‌مان اطمینان حاصل کنیم.

با توجه به این ویژگی‌ها، تصمیم بر آن شد که چارچوب ما دو موجودیت کنترلر^{۲۳} و رویداد را در سطحی پایه‌ای و بنیادین برای استفاده فراهم می‌کند. بدین ترتیب می‌توانیم این دو موجودیت را به اقتضای شرایط هر مسئله گسترش داده و سیستم را مدل‌سازی کنیم. در ادامه، به جزئیات آن‌ها بیش‌تر می‌پردازیم.

کنترلر

کنترلر، وظیفه‌ی اجرا و مدیریت فرآیند شبیه‌سازی را بر عهده دارد. به زبان ساده، با شروع فرآیند شبیه‌سازی، رویدادها به کنترلر ارسال می‌شوند، و کنترلر آن‌ها را به «فهرست رویدادهای آینده^{۲۴}» می‌افزاید و این فهرست را بر اساس «زمان مقرر^{۲۵}» رویدادها مرتب می‌کند. پس از پایان شلیک^{۲۶} رویداد پیشین، رویداد پیش رو^{۲۷} از FEL خارج و پس از جلو بردن ساعت شبیه‌سازی^{۲۸}،

¹⁵ Extensibility

¹⁶ Variables

¹⁷ State

¹⁸ Logic

¹⁹ Behavior

²⁰ Scalability

²¹ Complexity

²² Efficiency

²³ Controller

²⁴ Future Events List (FEL)

²⁵ Due Time

²⁶ Triggering

²⁷ Upcoming Event

²⁸ Simulation Clock

چارچوبی برای شبیه‌سازی رویدادهای گسسته

شلیک می‌گردد. در این میان اگر ساعت شبیه‌سازی به زمان مشخص شده برای پایان برسد یا از آن فراتر رود، شبیه‌سازی متوقف می‌گردد.

کنترلر، دو پارامتر رویداد نخستین^{۲۹} و زمان پایان شبیه‌سازی^{۳۰} را دریافت و صحت‌سنجی^{۳۱} می‌کند. سنجش صحت پارامترها عبارتند از مثبت بودن زمان پایان شبیه‌سازی و صفر بودن زمان مقرر رویداد نخستین. پس از سنجش صحت پارامترها، رویداد نخستین به FEL افزوده شده و ساعت شبیه‌سازی برابر با دقیقه صفر قرار داده می‌شود. لازم به ذکر است که در این چارچوب، تمامی زمان‌ها با واحد دقیقه به کار می‌روند. خلاصه‌ای از پارامترها و متغیرهای مربوط به کنترلر را می‌توانید مشاهده کنید:

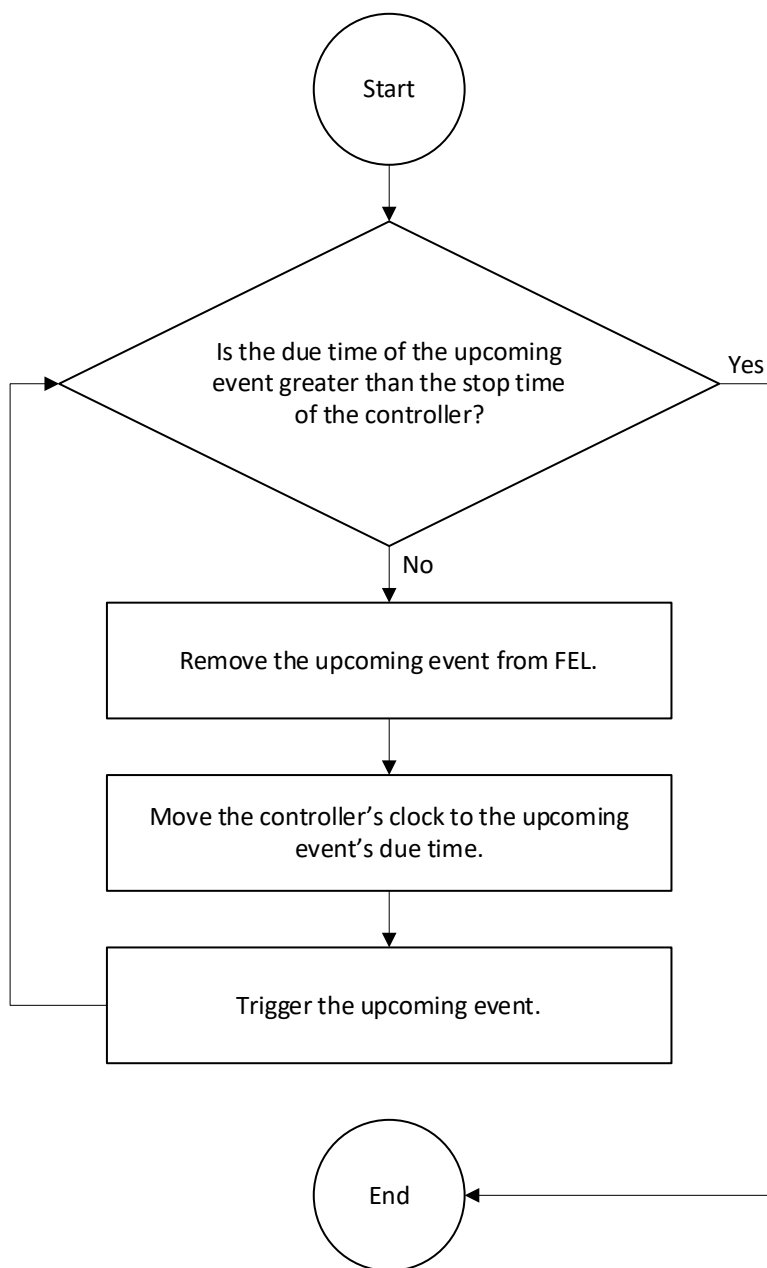
پارامترها	متغیرها
رویداد نخستین	فهرست رویدادهای آینده
زمان پایان شبیه‌سازی	ساعت شبیه‌سازی

حال، کنترلر برای شروع فرآیند شبیه‌سازی آماده است. با شروع فرآیند، رویداد نخستین که به FEL افزوده شده بود را از این فهرست خارج و سپس شلیک می‌کنیم. در پی آن، رویدادهای بعدی دریافت شده و به FEL افزوده می‌گردند و این حلقه تکرار می‌گردد، تا این که به زمان پایان شبیه‌سازی برسیم. در شکل بعدی، نمودار جریان را برای فرآیند شبیه‌سازی کنترلر می‌توانید مشاهده کنید. این نمودار همان توضیحات ساده‌ای که پیش‌تر داده شدند را با جزئیات بیش‌تر ارائه می‌دهد.

²⁹ Initial Event

³⁰ Simulation Stop Time

³¹ Validation



رویداد

این موجودیت، بستر را برای ایجاد رویدادهای مختلف یک سیستم فراهم می‌کند. تنها پارامتری که دریافت می‌کند فاصله^{۳۲} زمان شلیک آن با ساعت شبیه‌سازی است و پس از آن که به کنترلر ارسال شد، زمان مقرر آن برای شلیک مشخص می‌گردد. در نهایت، کنترلر رویداد را به FEL افزوده و فهرست بر اساس زمان مقرر هر یک مرتب می‌کند.

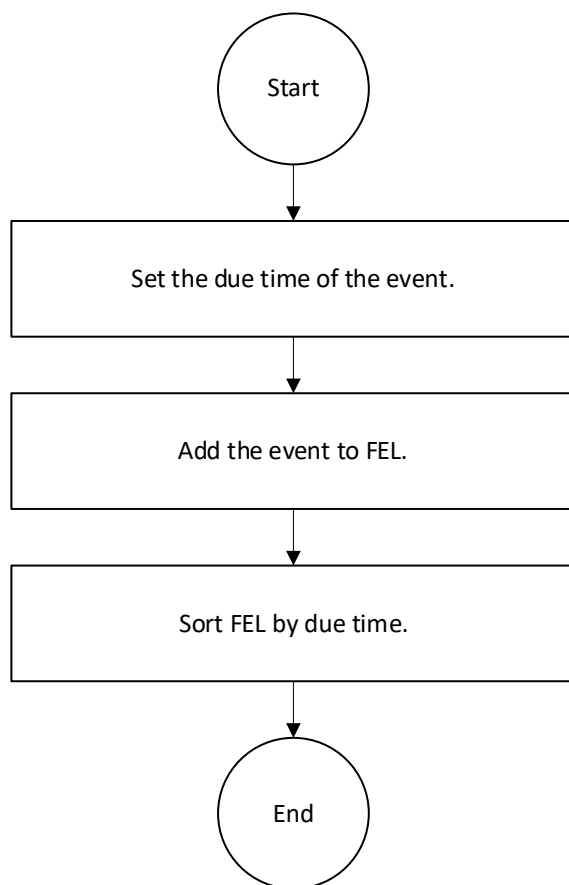
³² Interval

$$Due\ Time_{event} = Simulation\ Clock + Interval_{event}$$

خلاصه‌ای از پارامترها و متغیرهای مربوط به رویداد را می‌توانید مشاهده کنید:

پارامترها	متغیرها
فاصله زمانی	زمان مقرر

در ادامه، نمودار جریان را برای ارسال^{۳۳} یک رویداد به کنترلر شبیه‌سازی مشاهده می‌کنید.



در ادامه، به نحوه مدل‌سازی و رسیدن به جواب هر مسئله با استفاده از این چارچوب می‌پردازیم.

³³ Dispatching

مدل سازی و حل مسئله‌ها

برای حل هر مسئله با استفاده از چارچوب پیشنهادی، کنترلری طراحی می‌شود که رفتارهای بنیادین و پایه‌ای را از کنترلر چارچوب به ارث می‌برد. به نوعی، ما به این مسائل با دیدی شیء‌گرا^{۳۴} نگاه کرده و در کنار کنترلر، رویدادهای شبیه‌سازی را نیز با ارث‌بری^{۳۵} از رویداد بنیادین پیاده می‌کنیم.

از آنجایی که سیستم نخست حالتی خاص از سیستم دوم است، می‌تواند رفتارها و منطق مربوط به شبیه‌سازی این سیستم را با اندکی تغییر به ارث ببرد. بنابراین ابتدا به حالت کلی پرداخته و سپس به سراغ حالت خاص می‌رویم.

حالت کلی: چند خدمت‌دهنده

موجودیت‌های مختص به این سیستم

این سیستم دو موجودیت مخصوص به خود را دارد. خدمت‌دهنده و مشتری. خدمت‌دهنده دارای دو متغیر است: مجموع زمان به‌کارگیری و درحال خدمت‌دهی‌بودن. از متغیر اول در ادامه برای محاسبه معیارهای کارایی سیستم و از دومی برای ارسال مشتریان به خدمت‌دهنده‌هایی که در دسترس‌اند استفاده می‌کنیم.

هر مشتری نیز دارای پارامترهای زمان ورود، مدت دریافت خدمت، و زمان خروج است. با استفاده از این سه، همچنین می‌توان دو متغیر مربوط به زمانی که مشتری در سیستم سپری کرده و زمان انتظار وی را محاسبه کرد.

$$Total\ Time\ the\ i^{th}\ Customer\ Spent\ in\ the\ System = Departure\ Time_i - Arrival\ Time_i$$

$$Waiting\ Time\ of\ the\ i^{th}\ Customer = Total\ Time_i - Service\ Time_i$$

کنترلر

کنترلر این سیستم، فرآیند شبیه‌سازی را از چارچوب به ارث می‌برد. اما نیاز است تا پارامترها، متغیرها، و خروجی‌های شبیه‌سازی مختص به این مسئله را به آن اضافه کنیم. رویکرد به‌کارگیری خدمت‌دهنده می‌تواند «اولین در دسترس^{۳۶}» یا «تصادفی^{۳۷}» باشد. همچنین خروجی‌ها با استفاده از فهرست مشتریان تاریخی و فهرست خدمت‌دهنده‌ها محاسبه می‌شوند:

³⁴ Object-Oriented

³⁵ Inheritance

³⁶ First Available

³⁷ Random

$$AUTP = \frac{1}{n_s} \sum_{i=1}^{n_s} \frac{\text{Total Utilization Time of } i^{th} \text{ Server}}{\text{Simulation Clock (After Stopping the Simulation)}}$$

$$AWTP = \frac{1}{n_c} \sum_{i=1}^{n_c} \frac{\text{Waiting Time of the } i^{th} \text{ Customer}}{\text{Total Time the } i^{th} \text{ Customer Spent in the System}}$$

$$ASTP = 1 - AWTP$$

خلاصه‌ای از این بخش را در جدول زیر می‌توانید مشاهده کنید:

پارامترها	متغیرها	خروجی‌های شبیه‌سازی
رویداد نخستین	فهرست رویدادهای آینده	میانگین درصد زمان به‌کارگیری خدمت‌دهنده‌ها ^{۳۹}
زمان پایان شبیه‌سازی	ساعت شبیه‌سازی	
تعداد خدمت‌دهنده‌ها	فهرست مشتریان تاریخی	میانگین درصد زمان انتظار مشتریان ^{۴۰}
رویکرد به‌کارگیری خدمت‌دهنده ^{۳۸}	صف انتظار مشتریان	میانگین درصد زمان خدمت‌دهی به مشتریان ^{۴۱}
	فهرست خدمت‌دهنده‌ها	

در نهایت، در شکل زیر، نمودار جریان فرآیند شبیه‌سازی کنترلر شبیه‌سازی سیستم چندخدمت‌دهی موازی دارای یک صف را

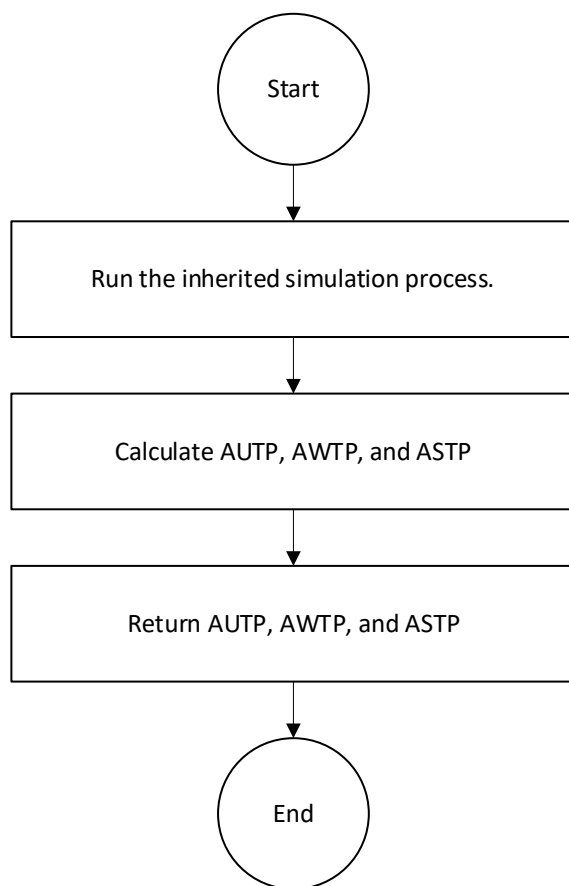
بینید:

³⁸ Server Utilization Strategy

³⁹ Average Utilization Time Percentage (AUTP)

⁴⁰ Average Waiting Time Percentage (AWTP)

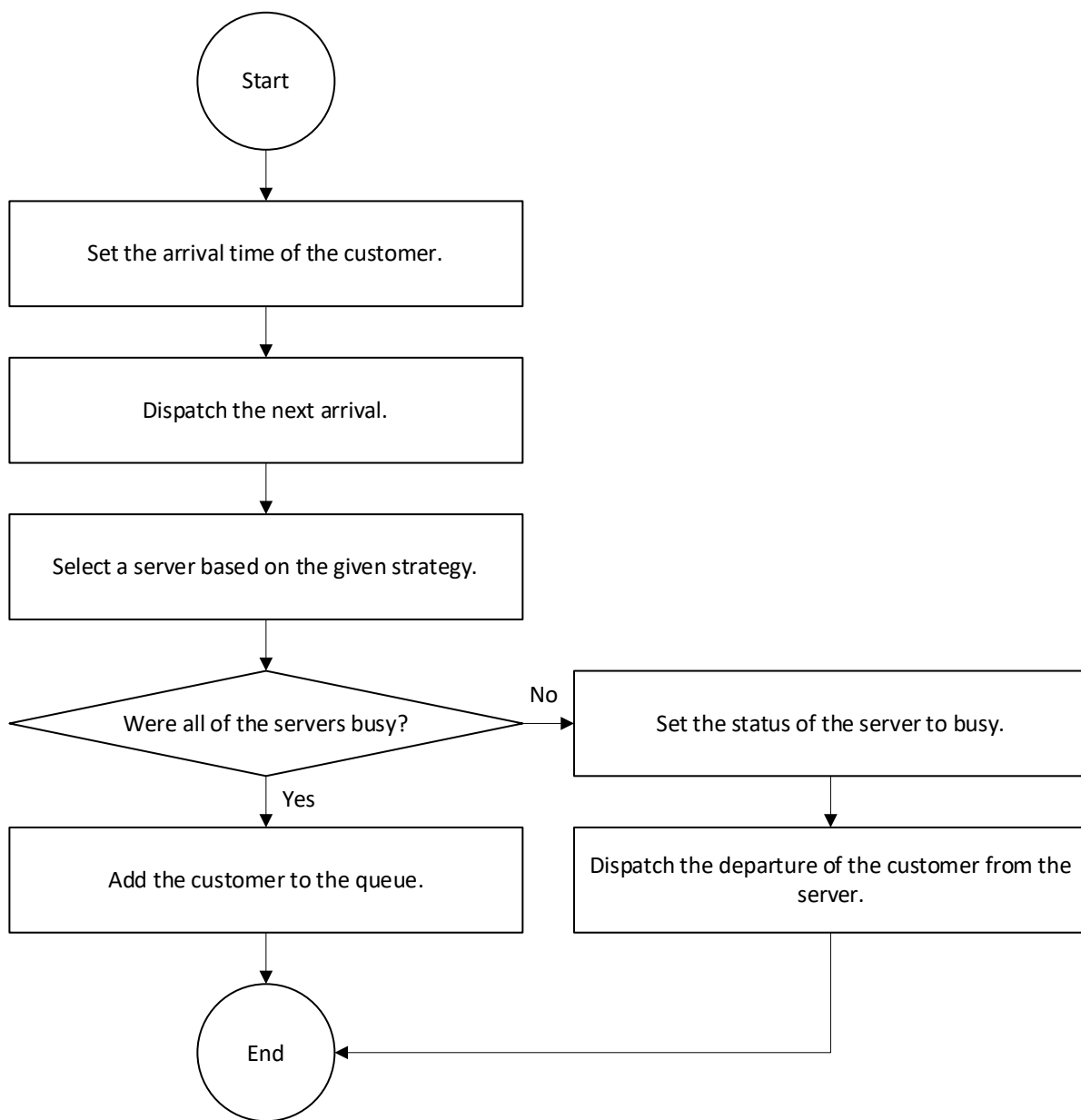
⁴¹ Average Service Time Percentage (ASTP)



رویدادها

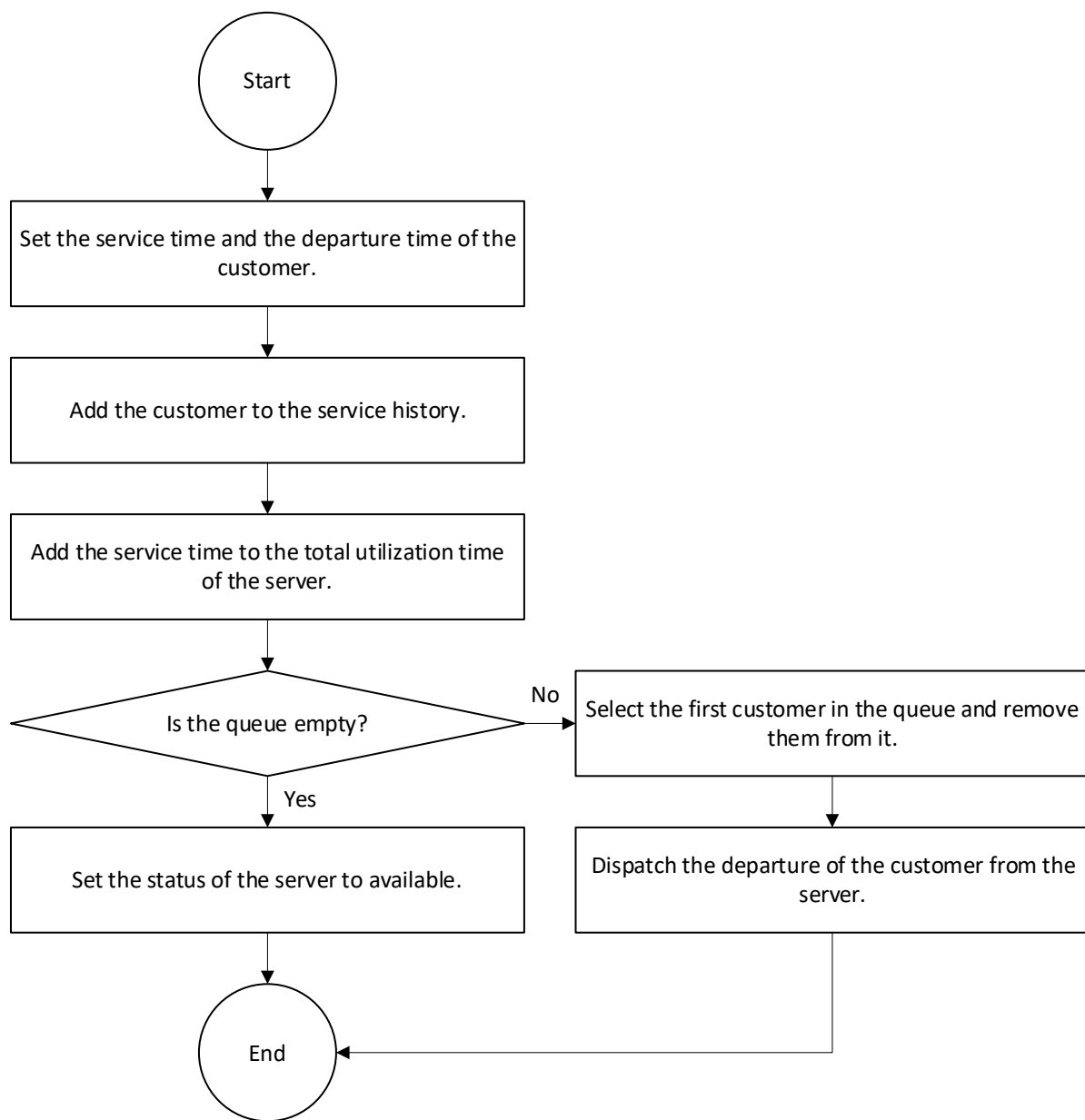
ورود

با شلیک رویداد ورود، زمان ورود مشتری مشخص می‌گردد. و رویداد ورود بعدی نیز به کنترلر ارسال می‌شود. با استفاده از رویکرد مشخص شده برای انتخاب خدمت‌دهنده، به دنبال خدمت‌دهنده‌ای برای خدمت‌دهی برای مشتری می‌گردیم. اگر خدمت‌دهنده‌ای در دسترس نبود، مشتری را به صف اضافه می‌کنیم. در غیر این صورت، وضعیت خدمت‌دهنده را به «مشغول» تغییر داده و رویداد خروج مشتری را به کنترلر ارسال می‌کنیم. نمودار جریان این رویداد را در شکل بعد می‌توانید مشاهده کنید.



خروج

با شلیک رویداد خروج، مدت دریافت خدمت و زمان خروج مشتری را مشخص می‌کنیم. سپس مشتری را به فهرست مشتریان تاریخی می‌افزاییم. همچنین، مدت خدمت‌دهی را به مجموع زمان خدمت‌دهی خدمت‌دهنده اضافه می‌کنیم. حال، اگر صف انتظار خالی از مشتری باشد، وضعیت خدمت‌دهنده را به «دردسترس» تغییر می‌دهیم. در غیر این صورت، اولین مشتری حاضر در صف را از صف خارج کرده و رویداد خروج وی را به کنترلر ارسال می‌کنیم. در نمودار پایین، جزئیات جریان این رویداد را می‌توانید ببینید.



حالت خاص: دو خدمت‌دهنده

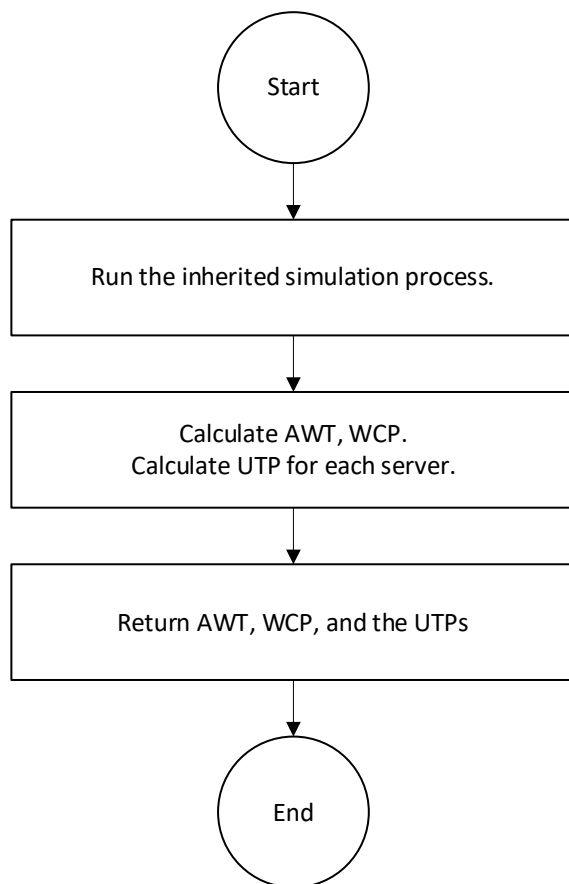
در این حالت، کنترلر و دو رویداد حالت قبلی را استفاده می‌کنیم. تنها تفاوت این است که پارامتر تعداد خدمت‌دهنده‌ها را برابر ۲ قرار داده، و به جای معیارهای کارایی پیشین، از معیارهای میانگین زمان انتظار^{۴۲}، درصد مشتریانی که وارد صف انتظار شدند^{۴۳}، و

^{۴۲} Average Waiting Time (AWT)

^{۴۳} Waiting Customers Percentage (WCP)

نتایج

درصد زمان به کارگیری هر خدمت‌دهنده استفاده می‌کنیم. برای محاسبه معیار اول، از زمان‌های انتظار بزرگ‌تر از صفر میانگین گرفته و برای محاسبه معیار دوم، تعداد مشتریانی که زمان انتظاری بزرگ‌تر از صفر دارند را بر تعداد کل مشتریان تقسیم می‌کنیم.



نتایج

مسئله یکم

همانطور که در جدول زیر می‌توانید مشاهده کنید، با استفاده از دو رویکرد متفاوت در انتخاب خدمت‌دهنده برای خدمت‌دهی، نتایجی با شباهت بسیار بالا و اختلاف ناچیز به دست آوردیم.

زمان به کارگیری بردیا (درصد)	زمان به کارگیری آریا (درصد)	مشتريانی که وارد صف انتظار شدند (درصد)	میانگین زمان انتظار (دقیقه)	رویکرد انتخاب خدمت‌دهنده
99.2777%	98.5024%	97.7572%	38.632304	ارجحیت آریا بر بردیا در خدمت‌دهی
98.9002%	98.8553%	97.7525%	38.490353	انتخاب تصادفی خدمت‌دهنده

مسئله دوم

برای به‌دست آوردن تعداد بهینه‌ی خدمت‌دهنده‌ها، یک معیار امتیاز محاسبه کردیم که برابر با میانگین هارمونیک^{۴۴} دو شاخص AUTP و ASTP می‌باشد.

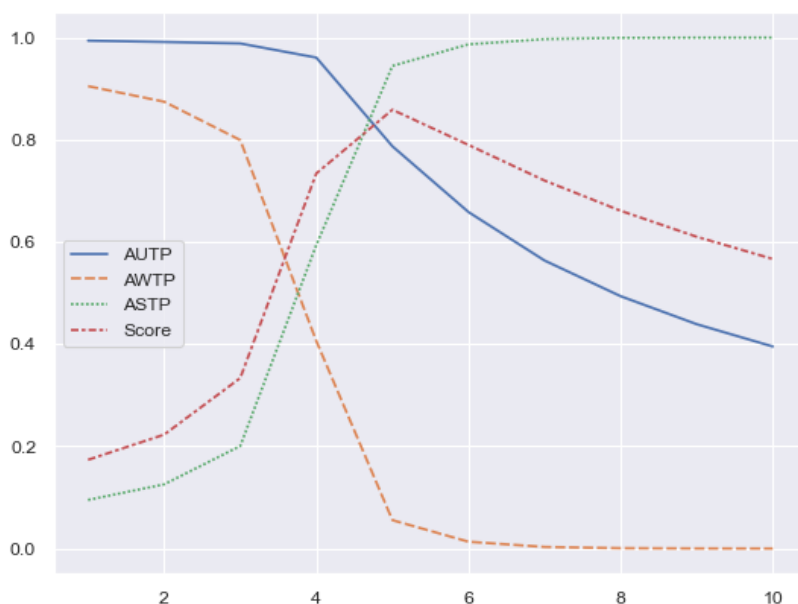
$$Score = \frac{2 \times AUTP \times ASTP}{AUTP + ASTP}$$

هرچه این امتیاز بیش‌تر باشد، به معنای بیش‌تر بودن نسبت زمان به کارگیری خدمت‌دهنده‌ها به زمان کل، و کمتر بودن نسبت مدت زمان انتظار مشتریان به کل زمان سپری شده در سیستم است. بنابراین، خواسته‌ی مسئله را با این روش می‌توانیم برآورده کنیم.

Number of Servers	AUTP	AWTP	ASTP	Score
1	0.993953	0.904890	0.095110	0.173607
2	0.991370	0.874414	0.125586	0.222930
3	0.988410	0.799357	0.200643	0.333572
4	0.960789	0.406068	0.593932	0.734078
5	0.787216	0.055549	0.944451	0.858695
6	0.658503	0.013313	0.986687	0.789862

⁴⁴ Harmonic Mean

7	0.563629	0.003075	0.996925	0.720123
8	0.493824	0.000637	0.999363	0.661015
9	0.439020	0.000141	0.999859	0.610139
10	0.395373	0.000029	0.999971	0.566687



نتیجه گیری

از مسئله اول، نتیجه می گیریم که به دلیل یکسان بودن تابع توزیع زمان خدمت دهی آریا و بردیا، تفاوت رویکردها در انتخاب خدمت دهنده برای خدمت دهی به هر مشتری، تغییری در کارایی سیستم و نتایج حاصل نمی کند. اما از مسئله دوم، با توجه به معیار امتیاز محاسبه شده، نتیجه می گیریم که تعداد بهینه ی خدمت دهنده ها برابر با ۵ نفر می باشد. زیرا در این نقطه به بیشینه امتیاز 78.98% می رسیم.

جدول گزارش عملکرد اعضای گروه