



دانشگاه صنعتی شریف

دانشکده مهندسی صنایع

پروژه کارشناسی مهندسی صنایع
گرایش برنامه ریزی و تحلیل سیستم‌ها

شبیه سازی با استفاده از نرم افزار **Arena**

و مطالعه موردی یک جایگاه سوخترسانی

استاد راهنما :

دکتر سید تقی اخوان نیاکی

تهیه کننده:

کاظم یاقوتکار

تابستان ۸۱

فهرست

صفحه	عنوان
۳	بخش اول : آشنایی با نرم افزار Arena
۴	۱.مقدمه
۴	۲.قابلیتهای Arena
۵	۳.محیط Arena
۶	۴. نصب نرم افزار
۶	۵.فراخوانی یک مدل موجود
۶	۶.معرفی برخی از ماژولهای Arena
۶	۶.۱.ماژول Arrive
۷	۶.۲.ماژول Server
۷	۶.۳.ماژول Depart
۷	۶.۴.ماژول Simulate
۸	۷.اتصالات ماژولها (Module Connections)
۸	۸.توسعه منابع
۹	۹.جمع آوری داده‌های آماری
۹	۱۰.تغییر دادن اشکال موجودیت‌ها: (Chaning Entity Pictures)
۹	۱۱.متغیرها و عبارتها: (Variables and Expressions)
۱۰	۱۲.مجموعه ها(Sets)
۱۱	۱۳.نمایش نمودارها و متغیرها (Adding Plot & Variables)
۱۱	۱۴.آنالیز ورودی: مشخص کردن پارامترها و توزیع های مدل
۱۱	۱۵.اجرای مدل
۱۲	۱۶.تحلیلگر خروجی: (The Output Analyzer)
۱۳	۱۷.آنالیز حساسیت (Sensitivity Analysis)
۱۳	۱۸.ترتیب و توالی عملیات (Sequences)
۱۳	۱۹.بررسی وصحت مدل (Verification)

۱۵	۲۰. انواع سیستم حمل و نقل در Arena
۱۷	۲۱. اعمال تغییر در مدل در حین اجرای مدل
۱۸	۲۲. آماده سازی مدل و زمان اجرا (Warm-Up and Run Length)
۱۹	۲۳. مدل خدمت و خدمات با موجودیت های برگشتی
۲۵	بخش دوم : شبیه سازی پمپ بنزین
۲۶	۱. مقدمه
۲۶	۲. هدف این مطالعه شبیه سازی
۲۶	۳. مکان آمارگیری
۲۶	۴. زمان آمارگیری
۲۷	۵. نحوه آمارگیری
۲۷	۶. ورود اطلاعات به کامپیوتر
۲۷	۷. پردازش آمار
۲۷	۸. آنالیز ورودی
۲۸	۸. ۱. توزیع زمان بین ورودی ها
۲۹	۸. ۲. توزیع زمان خدمت دهی
۳۰	۹. مدل سازی
۳۲	۱۰. تشریح مدل
۳۳	۱۱. توضیح یک فرض مهم در مدل
۳۳	۱۲. تعیین نوع شبیه سازی و زمان Warm-Up
۳۴	۱۳. شبیه سازی مدل با استفاده از داده های واقعی برای اطمینان از صحت مدل
۳۶	۱۴. تعیین تعداد دوباره سازی ها برای شبیه سازی (سناریوی شماره ۱)
۳۷	۱۵. شبیه سازی سیستم براساس سناریوی شماره ۱
۳۹	۱۶. تعیین تعداد دوباره سازی ها برای شبیه سازی (سناریوی شماره ۲)
۴۰	۱۷. شبیه سازی سیستم براساس سناریوی شماره ۲
۴۱	۱۸. مقایسه دو سناریو
۴۳	۱۹. بحث و نتیجه گیری
۴۴	پیوست ۱. نمونه ای از آمار

بخش اول :آشنایی با نرم افزار Arena

۱. مقدمه

Arena یکی از قوی‌ترین نرم افزارهای شبیه‌سازی سیستم‌های گسسته پیشامد است که در کنار نرم‌افزارهای قدرتمندی چون Witness و Slam بخش قابل ملاحظه‌ای از بازار نرم افزارهای شبیه‌سازی را در اختیار دارد. Arena را در حقیقت می‌توان نرم افزاری دانست که بر ساختارهای SIMAN بنا شده است. موتور داخلی شبیه‌سازی در هر مدل Arena مدل متناظری براساس SIMAN است. محیط کار در Arena کاملاً داری ساختار استاندارد نرم افزارهای شی‌گرای مایکروسافتی است. ابزار تهیه مدل‌های شبیه‌سازی در این نرم افزار بر خلاف SIMAN بجای خط دستور، ماژولهایی است که با قراردادن آنها در صفحه مدل و تعریف پارامترها و ارتباط بین آنها مدل را تعریف می‌کند. تا کنون بیشتر از ۶ بار این نرم‌افزار از طرف شرکت تهیه کننده آن یعنی Rockwell Software Inc. مورد تجدید نظر قرار گرفته و آخرین نسخه آن عنوان ویرایش ۶ را یدک می‌کشد. ویرایش نسخه از این نرم‌افزار که برای این پروژه بکار گرفته شد ۵ است.

۲. قابلیت‌های Arena

Arena با دارا بودن ماژولها و ابزارهای گسترده خود قابلیت‌های زیر را برای شبیه‌سازی فراهم می‌کند:

- تولید مقادیر تصادفی با روشهای مختلف
- یک برنامه مجزا به نام Input Analyzer که امکان تجزیه و تحلیل داده های ورودی را فراهم می کند.
- امکان تهیه انیمیشن بسیار قوی برای ارائه شبیه‌سازی به مدیران
- امکان Import کردن فایل‌های DXF و انواع Icon ها برای تهیه انیمیشن بهتر
- وجود مثالها و مدل‌های آماده شده مختلف که هر کدام می توانند با اعمال تغییرات به مدل‌های شبیه‌سازی برای سیستم‌های مورد نظر کاربر تبدیل شوند.
- برخوردار بودن از الگوهای (Template) مختلف جهت ایجاد مدلها
- امکان فراخوانی داده های ورودی از نرم افزارهای مجموعه Office
- Export کردن داده‌های خروجی به صورت فایل‌های قابل رویت برای نرم‌افزارهای مجموعه Office
- جمع‌آوری داده‌های شبیه‌سازی با روشهای ساده و بسیار منعطف برای کاربر
- آنالیز واریانس و تجزیه تحلیل حساسیت بر روی مدل
- شبیه‌سازی رده‌هایی از سیستم‌های پیوسته

۳. محیط Arena

محیط کلی این نرم افزار شبیه شکل کلی نرم افزارهای میکروسافت است. ماژولهایی که با استفاده از آنها مدل های شبیه سازی ساخته می شوند در Template Panel هایی با عنوانهای زیر گروهبندی شده اند :

۱. Basic Process که ماژولهای زیر را در خود جای داده است:

Creat,Dispose,Process,Decide,Batch,Separate,Assign,Reccord.Entity,Queue,Resource,Variable,Schedule,Set

۲. Advanced Process که ماژولهای زیر را در خود جای داده است:

ReadWrite,Release,Remove,Seize,SearchSignal,Store,Unstore,AdvanceSet,Expression,Failure,File,StateSet,Statistics,Storage,Delay , Dropoff, Hold ,Match, Pickup,

۳. Advanced Transfer که ماژولهای زیر را در خود جای داده است:

Enter,Leave,PickStation,Route,Station,Access,Convey,Exit,Start,Stop,Activate,Allocate,Free,Halt,Move,Request,Transport,Sequence,Conveyor,Segment,Transporter,Distance

۴. Blocks که ماژولهای زیر را در خود جای داده است:

Acess,Activate,Allocate,Alter,Assign,Begin,Block,Branch,Capture,Close,Combine,Convey,Copy,Count,Create,Delay,Detect,Dispose,Dropoff,Duplicate,Else,Elseif,Endif,EndWhile,Event,Exit,FindJ,Free,Group,Halt,If,Include,Insert,Match,Move,PickQ,Pickup,Preempt,Proceed,Qpick,Queue,Read,Release,Relinquish,Remove,Request,Route,Scan,Search,Seize,Select,Signal,Split,,Stack,Start,Station,Stop,Store,Tally,Trace,Transport,Unblock,Unstore,VBA,Wait,While,Write,Zap

۵. ContactData که ماژولهای زیر را در خود جای داده است:

Configuration,Call,Call Pattern,Agent,Schedule,Report,Animate

۶. CSUtil که ماژولهای زیر را در خود جای داده است:

AggregateCounts,TransferConference,QueueForAgent,HiddenBlk

۷. Elements که ماژولهای زیر را در خود جای داده است:

Arrivals,Attribute,Begin,Blockages,Continuous,Conveyors,Counters,Cstats,Discrete,Distances,Distribution,Dstats,Entities,Events,Expressions,Failures,Files,Frequencies,Include,Initialize,Intersections,Levels,Links,Networks,Nicknames,Outputs,Parameters,Pictures,project,Queues

۸. Packaging که ماژولهای زیر را در خود جای داده است:

Machine, Conveyor , Machine Link , Merge ,Split ,Switch , Operator ,Operator Group , Operator schedule , Palletizer , Storage , Valve, Tank ,Product ,Production Plans ,Actions , Label ,Simulate

۹. Script که ماژولهای زیر را در خود جای داده است:

Begin Script, Queue for Agent , Remove from Queue, Wait, Priority , Message , Disconnect , Overflow , Transfer to Script , Transfer to Agent ,Conference , Branch ,Assignment , End Script

۱۰. UtlArena که ماژولهای زیر را در خود جای داده است:

HiddenBlk , SzHidBlck , Cond_Assign , Hidden_All_Type

۴. نصب نرم افزار

برای نصب نرم افزار Arena باید فایل Setup.exe را در محیط ویندوز اجرا کنید. برای اینکه Arena اجرا شود کامپیوتر شما باید به برنامه Visual Basic مجهز باشد. Arena یک نرم افزار تحت سیستم عاملهای Windows 98 , Windows NT است که از همه جنبه ها و عملکردهای معمول آنها بهره می برد.

۵. فراخوانی یک مدل موجود

برای فراخوانی یک مدل از قبل ساخته شده از منوی File استفاده کنید و فایل مورد نظر خود را انتخاب کنید. پنجره ای که ظاهر شده است پنجره مدل نام دارد. در طول سمت چپ پنجره ، نوار ابزار Template قرار دارد که با عنوانهای Common و Attach مشخص شده اند. این نوارها با پانلهای مدلسازی چندگانه مرتبط هستند که در هر زمان یکی از آنها می تواند در قسمت سمت چپ مدل نمایش یابد. بطور پیش فرض نوار Common برای نمایش فعال است. این گزینه محتوی مجموعه ای از ساختارهای مدلسازی است که بطور معمول در ماژولهایی نظیر Arrive ، Depart ، Server . Animate و Simulate به کار می رود. اگر بر روی نوار Attach کلیک کنید ، یک کادر محاوره ای ظاهر می شود که به شما اجازه می دهد به دیگر ساختارهای مدلسازی دسترسی پیدا کنید.

۶. معرفی برخی از ماژولهای Arena

در اینجا چند ماژول مهم و اساسی که در تمام مدلها بکار گرفته می شوند معرفی می شوند. بقیه ماژولها بعلت ارتباطی که با طرح کلی مدل و مورد کاربرد دارند در بین قسمتهای بعد و مطالعات موردی معرفی می شوند.

۶.۱. ماژول Arrive

برای فراخوانی هر ماژول پنجره Model فقط کافی است بر روی کادر سفید رنگ با خطوط حاشیه آبی که حاوی نام ماژول است دو بار کلیک کنید.

ماژول Arrive نقطه تولد موجودیتهایی است که از خارج به مدل وارد می‌شوند. با دوبار کلیک بر روی آن کادر محاوره‌ای Arrive باز می‌شود. این کادر محاوره‌ای از سه قسمت اصلی تشکیل شده است:

- Enter Data ، که طبیعت نقطه ورودی را برای موجودیتهای تشریح می‌کند.
 - Arrival Data ، که طبیعت فرآیند ورود را مشخص می‌کند.
 - Leave Data ، که مشخص می‌کند چه اتفاقی برای داده‌ها وقتی که وارد مدل شده‌اند ، می‌افتد.
- در بخش Enter Data ، بخشی به نام Station وجود دارد. Station یک مکان فیزیکی در مدل برای رجوع انتقال موجودیتهای است.

۶.۲. ماژول Server

ماژول Server ماشین را نشان می‌دهد که شامل منبع ، صف و زمان لازم برای انجام عملیات بر روی قطعات است. این ماژول دارای سه بخش Enter Data ، Server Data ، Leave Data است. در بخش Enter Data نام ایستگاه مورد نیاز مشخص می‌شود. Server Data فرآیند ایستگاه را بر روی موجودیتهای مشخص می‌کند و Leave Data وظیفه کنترل خروج موجودیتهای از ایستگاه را بر عهده دارد.

۶.۳. ماژول Depart

ماژول Depart موجودیتهایی را نمایان می‌سازد که در حال ترک سیستم هستند. سه بخش موجود در ماژول Depart عبارتند از : Enter Data برای چگونگی ترک سیستم ، بخش Count برای شمارش موجودیتهای در حال ترک سیستم و بخش Tally برای محاسبه و ثبت آمار و اطلاعات مورد نیاز.

۶.۴. ماژول Simulate

پارامترهایی مانند طول زمان اجرا و تعداد تکرارها در ماژول Simulate تعیین می‌شود. در بخش Project می‌توان نام پروژه و نام سازنده مدل را مشخص کرد. در بخش Replicate تعداد تکرارها به صورت پیش فرض برابر یک و زمان شروع پیش فرض برابر صفر است.

۷. اتصالات ماژولها (Module Connections)

برای ایجاد اتصالات می‌توان از منوی Module گزینه Connect را انتخاب کنید، سپس نقطه خروج ماژول مبدا را به نقطه ورودی ماژول مقصد متصل کنید. اگر گزینه Auto Connect در منوی Module علامت خورده باشد، شما می‌توانید با انتخاب یک ماژول موجود که دارای یک سر ارتباطی آزاد است و ایجاد یک ماژول جدید که می‌تواند بصورت منطقی با ماژول قبلی ارتباط داشته باشد بصورت اتوماتیک بین این دو ماژول ارتباط برقرار کنید.

در غیر اینصورت اگر بخواهیم یک مدل با زمان حمل غیر صفر داشته باشیم، لازم است به جای گزینه Connect گزینه Route را فعال کنیم که منجر به حذف خطوط راهنما و نقطه خروجی می‌شود. در این حالت، می‌توان از یک ماژول به ماژول دیگر مسیرهای بصری ایجاد کرد تا آیکنهای نمایشگر موجودیتها در طول اجرای شبیه‌سازی در طول آنها جابجا شوند.

ماژول Simulate به هیچ چیزی اتصال ندارد، زیرا این ماژول یک ماژول داده‌ای است که موجودیتها در درون آن هیچ جریانی ندارند و فقط برای ایجاد بعضی مقادیر یا شرطها و یا نشان دادن طول شبیه‌سازی به کار می‌رود. عموماً، برای کنترل جریان موجودیتها به اتصالات دیگری نظیر خطوط اتصال، مسیرها و یا گزینه‌های پیشرفته دیگری نیاز است.

۸. توسعه منابع

به طور کلی Arena چهار وضعیت مختلف برای منابع در نظر می‌گیرد که عبارتند از: Idle بیکاری، Busy مشغول، Inactive غیرفعال و Failed از کار افتاده (خراب) هنگامیکه در حال اجرای شبیه‌سازی به صورت انیمیشن هستید، اگر وضعیت منبع از Idle به Busy تغییر یابد، رنگ نشانه منبع نیز تغییر می‌کند. شما می‌توانید این مسئله را با انتخاب کردن منبع رنگ و دابل کلیک کردن روی آن مشاهده کنید. در این حالت پنجره Resource Picture Placement ظاهر می‌شود که در آن چهار وضعیت منبع، همراه با عکسهای مربوط به هر یک نشان داده شده‌اند، که حالت Inactive دارای یک نشان بدون رنگ و Failure حالت بدون نشان می‌باشد.

نرم‌افزار Arena علاوه بر اینکه عکس هر منبع را با تغییر وضعیت آن تغییر می‌دهد، قادر خواهد بود زمان‌هایی که منبع در هر وضعیت بوده است را نگهداری نماید و به صورت آماری گزارش دهد. یک منبع دارای وضعیت Idle است اگر هیچ موجودیتی (قطعه) درون آن نباشد. به محض اینکه یک قطعه وارد منبع شده و درون آن قرار گیرد وضعیت منبع و شکل آن به حالت Busy تغییر می‌یابد (توجه کنید فقط وقتی وضعیت Idle نشان داده می‌شود که کلیه واحدهای منبع خالی از قطعه باشند). اگر Arena هیچ منبع قابل دسترسی برای تخصیص نداشته باشد، تصویر حالت Inactive را نشان

می دهد. و در نهایت حالت Failed نشان داده می شود، اگر وضعیت Fail باشد که این حالت نیز بر عدم وجود منبع قابل دسترسی برای تخصیص به بخشها دلالت می کند.

توسط ماژولهای Server و Inspect می توانید داده ها و اطلاعات مربوط به خرابی را تعریف کنید. هنگامیکه در یکی از این ماژولها پنجره Resource Information با کلیک کردن روی گزینه Resource فعال شود، قسمت های Stateset، Failures و Downtimes در آن مشاهده می شوند، که به ترتیب نشان دهنده زمان خرابی و غیر قابل دسترس بودن منبع را تعیین می کنند.

قسمت Stateset به شما اجازه می دهد، حالتها و وضعیت های مختلف را برای منبع تعیین کنید.

۹. جمع آوری داده های آماری

یکی از ابتدایی ترین مراحل برنامه ریزی یک پروژه شبیه سازی، مشخص کردن اطلاعاتی است که در مدل مورد نیاز است؛ جمع آوری داده ها و آماده سازی آنها به گونه ای که در مدل به صورت مناسب به کار گرفته می شود، کار بسیار مهم و در عین حال وقت گیر و هزینه بر می باشد. در بسیاری از موارد مدل های شبیه سازی به اطلاعاتی از قبیل زمانهای بین دو ورود متوالی، زمان کل فرآیند، برنامه کاری و اپراتور و نیاز داریم تا از طریق این اطلاعات توزیع مناسبی را برای تولید اعداد تصادفی انتخاب کنیم. با جمع آوری این اطلاعات و ذخیره کردن آن در یک فایل متنی می توان از آنها در Arena استفاده کرد.

۱۰. تغییر دادن اشکال موجودیت ها: (Changing Entity Pictures)

گاهی اوقات برای زیبایی و یا تمیز دادن قطعه A و B تصمیم می گیریم اشکال قطعات را تغییر دهیم. با استفاده از ماژولهای Server, Arena و Inspect قادر به تعیین یک تصویر مناسب برای قطعه مورد نظر خواهید بود.

توجه داشته باشید که می توانید شکل منابع را نیز تغییر دهید که در این حالت اشکال مطابق وضعیت و حالت منبع (Idle، Busy، Failed و Inactive) تغییر پیدا می کنند، که این کار توسط پنجره Resource Picture Placement که با دابل کلیک کردن روی تصویر منبع باز می شود.

۱۱. متغیرها و عبارتها: (Variables and Expressions)

در بسیاری از مدلها، ممکن است بخواهیم از داده های جمع آوری شده در چندین جای مختلف استفاده نماییم. برای مثال در یک سیستم کوچک تولیدی زمان انتقال بین کلیه ایستگاهها، ۲ دقیقه در نظر گرفته شده است، اگر تصمیم بگیریم این مقدار را در طی آزمایشهایمان تغییر دهیم، باید برای هر

مسیر، در کادر محاورهای مربوطه، مقدار مورد نظر را تغییر داد. در بعضی موارد نیاز است اثر و یا رد پای هر یک از موجودیتها را در کل یا بخشی از سیستم تعیین کرده و یا در موارد دیگر از عبارت پیچیده در سرتاسر مدل استفاده کنیم. به عنوان مثال، اگر بخواهیم زمانهای مسیره‌های طی شده از توزیع خاصی پیروی کنند و یا بخواهیم زمان فرآیند را بر اساس نوع قطعه پایه گذاری نماییم، متغیرها و عبارات Arena این نوع احتیاجات را به سادگی بر طرف می نمایند. این کار از طریق ماژول Variables امکان پذیر است.

در ماژول Variables می توان متغیرهای عمومی و مقادیر ابتدایی آنها را تعریف کرد و در موارد لزوم به آنها ارجاع داد. همچنین قادر هستید از آرایه های یک یا دو بعدی استفاده نمایید. ماژول Expressions به شما اجازه می دهد، عبارات و مقادیر وابسته به آنها را در Arena تعریف کنید و در مدلها به نام آنها ارجاع دهید. با اینکه ممکن است متغیرها و عبارات تعریف شده کاملاً شبیه به هم به نظر رسند، ولی با کار کردهایی که به طور واضح تعریف شده اند قابل تشخیص هستند. متغیرهایی که توسط کار بر تعریف می شوند، می توانند بعضی از کمیتها را نگهداری کنند و در طول اجرای شبیه سازی مقادیر تخصیص داده شده به آنها، قابل تغییر خواهد بود به عنوان مثال، می توانیم یک متغیر به نام Number In System با مقدار اولیه صفر تعریف کرده و با ورود یک قطعه جدید به سیستم مقدار آن را ۱ واحد افزایش و با خروج یک قطعه از سیستم مقدار آن را ۱ واحد کاهش دهیم.

۱۲. مجموعه ها (Sets)

همچنانکه مدل شما پیچیده تر می شود، موجودیت باید در هر موقعیت یک یا چند منبع (شی) را از میان چند شی (Object) انتخاب نماید، فرض کنید، سه اپراتور به نامهای Paul، Lynn و Ann در اختیار دارید که هر یک از این اپراتورها قادر است کار مورد نیاز را انجام دهد. ولی شما می خواهید یکی از این سه اپراتور را انتخاب کنید. ماژول Set این امکان را برای شما فراهم می کند. مجموعه های موجود در نرم افزار Arena، گروههایی از اشیای مشابه هستند که توسط یک نام و با یک شاخص (Ident) می توانیم به آنها رجوع کنیم. عناصر و اعضای یک مجموعه خاص باید همه از یک نوع شی مانند: منابع، صف ها، تصاویر و اشکال و باشند. شما تقریباً هر نوع از اشیای موجود در Arena را می توانید در یک مجموعه (بسته به نیازهای مدلتان) جمع کنید. همچنین یک شی می تواند در بیشتر از یک مجموعه قرار گیرد.

فرض کنید از مجموعه اپراتورهای مورد نظر، Lynn شایسته تر است. پس ما از مجموعه ای به نام Lynn, Operations را انتخاب می نماییم. Arena در تعداد مجموعه ها، محدودیتی نداشته و از این نظر به راحتی می توان نیازهای گسترده ای را پوشش داد.

۱۳. اضافه کردن نمودارها و متغیرها (Adding Plots & Variables)

برای اضافه کردن نمودارها و متغیرها می توانید از نوار ابزار استفاده کنید. ولی استفاده کردن از ماژول از صفحه بسیار راحت تر خواهد بود. فرض کنید می خواهیم یک گراف برای تعداد قطعات موجود در صف و یک متغیر برای مقدار منابع قابل دسترسی یک مدل نمونه ایجاد کنیم.

- در قسمت Data Object در ماژول Animation قرار بگیرد.

- گزینه Queue را انتخاب کرده و گزینه صف مربوط به خدمت دهنده مورد نظر را از فروریز Queue Name انتخاب کنید.

- در قسمت Display as ، فقط گزینه Plot را انتخاب کنید، توجه کنید که سایر گزینه ها انتخاب نشده باشند.

- OK را بزنید.

قبل از اجرای مدل، روی گراف ایجاد شده دابل کلیک کنید. با این کار پنجره ای به نام Plot باز می شود، که می توانید توسط گزینه های مختلف آن نمودار را ویرایش کنید. مثلاً اگر روی دکمه Area کلیک کنید، می توانید رنگ زمینه گراف را تغییر دهید و یا برای محور x و y ، یک برچسب (Lable) قرار دهید و.....

۱۴. آنالیز ورودی: مشخص کردن پارامترها و توزیع های مدل

یک اصل بسیار مهم در مدل کردن . مشخص کردن نوع توزیعی است که متغیرهای تصادفی مدل از آنها پیروی می کنند. در بعضی مواقع تعیین دقیق تعداد منابع به طور مثال تعداد اپراتورهای یک ماشین تراش باید دقیقاً مشخص باشد.

استفاده کردن از مدل های قطعی بسیار خطرناک می باشد، زیرا پس از سپری شدن مدت طولانی. ممکن است نتایج شبیه سازی دقیق و صحیح بدست نیاید و این امر موجب می شود تا شخص تحلیل گر نتواند برداشت صحیحی از سیستم مدل شده بدست آورد و در نهایت تصمیم درستی اتخاذ کند. به همین دلیل در اکثر موارد سعی می شود از متغیرهای تصادفی با توزیع های احتمالی مختلف استفاده شود.

۱۵. اجرای مدل

برای اجرای مدل کلید GO را از نوار ابزار Run فشار دهید (مجموعه ای از کلیدهای کنترلی واقع در گوشه پایین سمت چپ) و یا از منوی Run گزینه Go را انتخاب کنید. اولین باری که یک مدل

اجرا می‌شود، (و یا بعد از ایجاد هر تغییر) Arena مدل را از لحاظ دارا بودن خطا چک می‌کند (شما می‌توانید این مرحله را با فشردن کلید Check در نوار ابزار Run و یا انتخاب گزینه Check Model از منوی Run یا فشردن کلید F4 انجام دهید). اگر خطایی وجود داشت، با حوصله به بررسی و رفع آن پردازید. سپس دوباره مدل را اجرا کنید. خواهید دید که موجودیها وارد و خارج می‌شوند؛ ساعت دیجیتالی در قسمت پایین صفحه به پیش می‌رود و شمارنده تعداد محصول افزایش می‌یابد و پلاتها کشیده می‌شوند. اگر مایل باشید که خلاصه مقادیر نهایی را ببینید، در جواب به پرسشی که از شما می‌شود گزینه Yes را انتخاب کنید. در اینجا یک صفحه متنی (با استفاده از Microsoft Notepad) ظاهر می‌شود. کلیه آمارها و اطلاعات اولیه به علاوه نتایجی که Arena به صورت اتوماتیک محاسبه کرده است، در این گزارش وجود دارد. پنجره گزارش Notepad را ببندید. برای دیدن مجدد پنجره گزارش، گزینه Results را از منوی Run انتخاب کنید.

توجه داشته باشید که نام ماژولها در پنجره اجرا مخفی می‌شوند و ماوس نیز در این پنجره قادر به انجام هیچ کاری بر روی اشیاء نمی‌باشد.

معمولاً قبل از اینکه مدل ساخته شده را اجرا کنند، آن را آزمایش می‌کنند تا اعتبار و صحت مدل تضمین شود و مدل عاری از خطا و اشتباه باشد. برای این منظور می‌توانید از روشهای زیر استفاده نمایید:

- ۱- از آیکن Check Button که در نوار ابزار Run Interaction در پایین پنجره Model، قرار دارد.
- ۲- با استفاده از منوی Run، گزینه Check Model

اگر مدل را برای تکمیل شدن و خاتمه یافتن اجرا نمایید، Arena از شما می‌پرسد که آیا تمایل به مشاهده نتایج دارید یا خیر؟ اگر Yes را انتخاب کنید. یک پنجره فعال می‌شود که در آن نتایج و خلاصه شبیه‌سازی نشان داده شده است.

۱۶. تحلیلگر خروجی: (The Output Analyzer)

نرم‌افزار Arena این توانایی را در اختیار کاربر قرار می‌دهد که روی داده‌های فایل‌های خروجی که در حین اجرای مدل ذخیره شده‌اند تجزیه و تحلیل انجام دهد و همچنین کاربر می‌تواند با استفاده از این ویژگی نرم‌افزار، گرافها و نمودارهای متعدد برای نتایج داده‌ها ترسیم نماید (بدون آنکه نیازی به اجرای مجدد مدل باشد). Output Analyzer هم می‌تواند به صورت یک برنامه مجزا همانند سایر برنامه‌های دیگر شروع شود و هم می‌تواند از طریق گزینه Output در منوی Tools در اختیار کاربر قرار گیرد. Output Analyzer با گروهی از داده‌ها کار می‌کند.

۱۷. آنالیز حساسیت (Sensitivity Analysis)

در آنالیز حساسیت این مسئله بررسی می شود، که آیا پارامترها و داده های جمع آوری شده روی مدل اثر می گذارد یا خیر؟ به عبارت دیگر اگر یکی از پارامترهای مدل تغییر کند، این تغییر چه اثری روی مدل خواهد گذاشت؛ در نتیجه می توانید یک فاصله اطمینان و یک دامنه قابل قبول برای داده های مورد نیاز در نظر بگیرید به گونه ای که تغییر آن پارامترها در آن دامنه روی مدل اثر نگذارد.

۱۸. ترتیب و توالی عملیات (Sequences)

بسیاری از سیستمها، توسط موجودیتهایی که از مسیرهای از پیش تعیین شده ولی متفاوت پیروی می کنند، توصیف می شوند. فرآیند مربوط به هر نوع قطعه قبل از ترک سیستم باید کامل شود، همچنین این مسئله در مورد سیستمهای خدماتی نیز صادق است. به عنوان مثال، یک مدل ترافیک مسافر در یک فرودگاه می تواند مسیرهای متفاوتی در فرودگاه نیاز داشته باشد. بسته به اینکه مسافران دارای کیفها و چمدانهای بزرگ، متوسط و یا کوچک هستند، این مسیرها تقسیم بندی می شوند. نرم افزار Arena می تواند موجودیت ها را درون مدل به طور اتوماتیک بر اساس توالی از پیش تعیین شده عبور از ایستگاهها، حرکت دهد. ماژول Sequences واقع در پانل Common اجازه تعریف یک توالی و ترتیب عبور موجودیتها از ایستگاههای مختلف را می دهد.

در ضمن حرکت موجودیت، طبق ترتیب تعریف شده، Arena اطلاعات مربوط به هر قسمت را جمع آوری و نگهداری می نماید. این کار توسط سه ویژگی (NS) Sequences، (M) Station و (IS) Jobshop به طور اتوماتیک انجام می شود. مقدار اولیه تمام این مشخصه ها به طور پیش فرض صفر می باشد، که با ایجاد یک موجودیت جدید، مقدار آن تغییر می کند. مشخصه Station حاوی نام موقعیت یا محل موجودیت و یا نام ایستگاهی است که موجودیت در حال منتقل شدن به سمت آن است، ویژگی Sequence شامل نام توالی و ترتیبی است که موجودیت دنبال خواهد کرد و با لا خره مشخصه Jobshop موقعیت موجودیت را در ترتیب تعیین شده، مشخص می کند.

۱۹. بررسی وصحت مدل (Verification)

آزمایش صحت مدل به معنی آن است که آیا مدل کامپیوتری ساخته شده (در نرم افزار Arena) با مدلی که مد نظر شما بوده است مطابقت دارد یا خیر؟ از طرف دیگر شما رفتار مدل ساخته شده در نرم افزار را بررسی کرده و با رفتار سیستم واقعی مقایسه می نمایید (این فرآیند بسیار ساده و در عین حال بسیار ضروری می باشد). این دو مبحث چیزی جدا از ساخت مدل شبیه سازی نمی باشند و قدم به قدم در حین ساخت مدل باید به آنها توجه شود.

برای بررسی صحت مدل (Verification) فقط کافی است، مدل ساخته شده را اجرا نمایید. اگر پیغام خطایی مشاهده نشد، می توان نتیجه گرفت مدل دارای هیچ اشتباه و خطایی نبوده و درست ساخته شده است در غیر این صورت با Debug کردن مدل یعنی با بر طرف ساختن خطای اعلام شده می توانید صحت مدل را تا مین کنید. البته بررسی صحت مدل های کوچک کار ساده ای می باشد، ولی در مدل های پیچیده و بزرگ این کار بسیار دشوار بوده، در حدی که هیچ گاه نمی توانید ۱۰۰٪ از صحت مدل تان مطمئن باشید.

یک راه ساده تر و ملموس تر برای بررسی صحت مدل، ورود فقط یک موجودیت (قطعه) به سیستم و اجرای قدم به قدم مدل و سپس جمع آوری اطلاعات می باشد. روش دیگر، جایگزین کردن همه (و یا بعضی) داده های ورودی مدل با عدد ثابت می باشد. استفاده کردن از داده قطعی، توانایی پیش بینی رفتار سیستم را به طور دقیق به شما می دهد.

یک راه معمول و مناسب، طولانی کردن زمان اجرای شبیه سازی برای انواع مختلف داده ها و سپس مشاهده نتایج می باشد. البته معمولاً قبل از اجرای مدل و گرفتن نتایج، پاسخها توسط مهندسان و کارشناسان مربوطه تخمین زده می شوند. در این صورت اگر نتیجه مشاهده شده از مدل حدوداً با مقدار تخمینی مطابقت داشت، مدل مورد قبول و در غیر این صورت شخص با مشکلات و مسایل گوناگون در سیستم مواجه است. به طور مثال اگر مهندسی مقدار تخمینی را عدد ۴۰۰ محاسبه کرد و جواب اجرای مدل ۹۰۰ باشد، شما به مشکل بزرگی بر خورد کرده اید. پس با تعریف و تعیین مجموعه شرایط مورد نیاز برای شبیه سازی، نتایج مدل تخمین زده می شوند، در این صورت اگر پس از اجرای مدل و مشاهده نتایج، جوابها با مجموعه شرایط تعریف شده مطابقت داشتند، مشکلی وجود ندارد، در غیر این صورت شما باید به دنبال دلیل وجود مشکل بروید.

۲۰. انواع سیستم حمل و نقل در Arena

برای انتقال مستقیم موجودیت ها بین ماژولها، اساسا از گزینه Connect (که در فصل قبل توضیح داده شده است) استفاده می شود. در این حالت زمان حرکت بین دو ماژول در نظر گرفته نمی شود. همچنین از امکان دیگری به نام Lables برای انتقال موجودیت ها می توان استفاده کرد. این دو گزینه در اکثر زمینه ها مشابه می باشند، تنها اختلاف بین آن دو این است که گزینه Connect به شما اجازه مشاهده جریان موجودیت ها را درون مدل می دهد، در حالیکه گزینه Lables این امکان را به کاربر نمی دهد.

روش دیگر استفاده از Route است. یک Route وظیفه انتقال موجودیت ها را با مد نظر قرار دادن زمان بین ایستگاه ها، بر عهده دارد.

این روشهای انتقال، قدرت پاسخگویی بسیاری از مدلها را دارند، ولی در بعضی از مدلها با محدودیت تعداد انتقال دهنده ها در هر مقطع زمان روبرو می شویم. برای مثال، در ساخت مدل یک شبکه ارتباطات، پهنای باند به یک ظرفیت مشخص محدود می شود. بنابر این ما باید یک روش برای محدود کردن تعداد پیامهای مقارن با یکدیگر که به وسیله هر ارتباطی از شبکه و یا کل شبکه، منتقل می شوند، داشته باشیم. حل چنین مسئله ای ساده می باشد. ارتباطات شبکه را به عنوان منابعی با یک ظرفیت یکسان در نظر بگیرید که بنا به نیاز، نیز قابل تغییر هستند. این نوع انتقالها به منابع محدود شده (Resource constrained) معروفند.

دسته بسیار بزرگ و متنوعی از سیستم های انتقال موجودیت با نام Material Handling)

جابجایی مواد) می باشد که به دو طبقه کلی تقسیم بندی می شوند:

- طبقه اول شامل تعداد انتقالهای هم زمان است که بر پایه تعداد وسایل در دسترس جابجایی مواد استوار می باشد. وسایل جابجایی مواد که در این دسته قرار دارند شامل: گاریها، واگن های دستی، واگن های سه شاخه، AGV ها، نیروی انسانی و هستند. اگر یک حمل مورد نیاز باشد، ابتدا باید وسیله مربوطه به محل مورد نظر منتقل شده و سپس توسط آن موجودیت به مقصد منتقل شود. این طبقه به Transporter (حمل کننده ها) معروف است.
- طبقه دوم، که تعداد انتقالها به فضای در دسترس وابسته است. همچنین در این طبقه باید تعداد کل انتقالهای زمان بین دو محل را کاهش دهیم. از تجهیزات و وسایل این طبقه می توان به نقاله ها، واگنهای برقی و اشاره کرد. در این حالت اگر به یک حمل نیاز داشته باشیم، ابتدا باید مقدار فضای مورد نیاز که توسط هر وسیله اشغال می شود تعیین شود و بعد ماژول مناسب آن انتخاب شود. به این طبقه از انواع انتقال ها در Arena، Conveyors (نقاله ها) اطلاق می شود.

سیستم نقاله در Arena به این صورت است که هر موجودیت برای استفاده از نقاله باید آنقدر منتظر بماند تا فضای کافی روی نقاله در محل مورد نظر آماده شود. همچنین Arena فرض می کند نقاله شامل سلولهایی با طول مساوی است که به طور ثابت در حال حرکت هستند. یک مثال بسیار واضح و ساده از نقاله ها پله های برقی در فروشگاهها می باشند که پله ها همان سلولهای نقاله هستند. برای مثال مسافری با سه چمدان به ۲ یا ۳ پله نیاز دارد، در حالی که شخصی بدون چمدان فقط به یک پله نیاز دارد. وقتی که شما به پله برقی نگاه می کنید اندازه این سلولها تا حدی آشکار است، در حالی که در مورد نقاله اندازه این سلولها چندان آشکار نیست. یعنی Arena طول نقاله را به یک سری متوالی از سلولهای برابر تقسیم می کند. این امر شما را در یک دو راهی قرار می دهد، زیرا شما از یک طرف می خواهید، اندازه سلول ها کوچکتر باشند تا به بیشترین دقت دست یابید و از طرفی مایل هستید، سلول ها بزرگتر باشند تا به بیشترین کارایی محاسباتی برسید با یک مثال این دو راهی را توضیح می دهیم.

شما نقاله ای دارید که ۱۰۰ فوت طول دارد و می خواهید قطعاتی را حمل کنید که طول آنها ۲ فوت است. بخاطر اینکه اندازه ها بر حسب فوت بیان شده است، امکان دارد پاسخ شما ۱ فوت باشد این بدین معنی است که نقاله ۱۰۰ سلول دارد و هر قطعه به دو سلول برای قراردان نیاز دارد شما به راحتی می توانید اندازه و سلول را به ۲ فوت و یا یک اینچ تغییر دهید که در این صورت ۱۲۰۰ سلول خواهید داشت، ولی با کامپیوترهای امروزی نباید نگران مقدار ۱۲۰۰ سلول باشیم. زیرا تفاوت اندکی در سرعت محاسبه مدل ایجاد می کند، اما مدلهایی وجود دارد که ممکن است نقاله هایی با طول بیش از ۵ مایل داشته باشند، در این صورت حتما سرعت مدل تحت تاثیر قرار خواهد گرفت و تفاوت اندازه سلولهای ۱ اینچ و ۱۰۰ فوت می تواند تاثیر زیادی بر سرعت اجرای مدل داشته باشد. تصور کنید که یک موجودیت به یک نقاله خالی رسیده است و می خواهد وارد آن شود. شما اندازه هر سلول را ۱۰۰ فوت مشخص کرده اید و وقتی که قطعه به نقاله می رسد فقط نیم اینچ از آن گذشته است. بنا براین باید قطعه منتظر بماند تا ۹۹ فوت و یازده و نیم اینچ از طول نقاله، از جلوی آن عبور کند تا سوار آن شود. ولی اگر شما اندازه هر سلول نقاله را ۱ اینچ مشخص کرده بودید، قطعه فقط باید به اندازه نیم اینچ منتظر می ماند.

در کل شما باید طول سلولها را با توجه به سیستمی که مدل می کنید مشخص نمایید. توصیه می شود، زمانی که نقاله بیش از حد مورد استفاده قرار نمی گیرد یا تا خیر آن تاثير چندانی در کار کرد سیستم ندارد از اندازه سلول بزرگ استفاده کنید و اگر باعث پدید آمدن بحران در کارکرد سیستم شد از اندازه سلول کوچک استفاده کنید. در این حالت دو محدودیت وجود دارد: اندازه موجودیت باید عدد صحیحی از سلول باشد. بنابراین شما در مدلتان موجودیتی که ۱/۵ سلول باشد نخواهید داشت، همچنین طول بخشهای نقاله (طول قسمتی از نقاله از یک محل به محل دیگر) باید تعداد صحیحی از

سلول ها را شامل شود. بنابراین اگر طول نقاله شما ۱۰۰ فوت بود، شما نمی توانید اندازه هر سلول را ۳ انتخاب کنید.

همانند Transporter Resource ها، نقاله ها هم چندین کلمه کلیدی دارند: Access، Convey و Exit برای محل یک موجودیت بو سیله نقاله های Arena شما ابتدا باید به فضای خالی روی نقاله دست یابید. سپس موجودیت را به مکان مورد نظر حمل (Convey) کنید و در آخر از آن خارج شوید. برای نشان دادن لی اوت فیزیکی نقاله Arena از بخشهایی تشکیل شده است که در ارتباط با یکدیگر، کل نقاله را به وجود می آورند. شما می توانید این قسمتها را به هم وصل کرده و یک نقاله حلقه ای بوجود آورید. برای سیستمهای مختلف، شما می توانید نقاله های متنوع تعریف کنید.

Arena دو نوع نقاله در اختیار کار بر قرار می دهد: انباشته ای Accumulating و غیر انباشته ای Non Accumulating هر دو این نقاله ها فقط در یک جهت حرکت می کنند و شما نمی توانید آنها را بر عکس کنید. نقاله غیر انباشته ای یک سطل و یا نقاله تسمه ای یا پلکان برقی است. فاصله بین موجودیت هایی که روی این نقاله ها قرار دارند تغییر نمی کند، مگر اینکه یک موجودیت خارج شود و فضای آن دو باره آزاد شود. زمانی که یک موجودیت به فضای این نقاله دسترسی پیدا می کند، در حقیقت نقاله از حرکت باز می ایستد. وقتی یک موجودیت به مقصد رسید نقاله می ایستد و موجودیت خارج شده و نقاله مجددا کار را شروع می کند. تفاوت عمده نقاله های انباشته ای با این نقاله ها این است آنها هرگز متوقف نمی شوند، اگر یک موجودیت روی نقاله انباشته ای متوقف شود سایر موجودیت ها به مسیر خود ادامه خواهند داد، ولی به هر حال موجودیت متوقف شده مانع رسیدن سایر موجودیت ها به مقصد می شود و سایر موجودیت ها پشت آن انباشته می شوند. هنگامی که موجودیت متوقف شد، از نقاله خارج شده یا به راه خود ادامه می دهد و سایر موجودیت های انباشته شده آزاد می شوند و به سمت مقاصدشان حرکت می کنند.

اما با توجه به اندازه های مشخص شده در مدل ممکن است همه موجودیت های انباشته شده در یک لحظه حرکت نکنند، مانند اتومبیلهای متوقف شده در یک آزاد راه، وقتی که اتومبیلها متوقف می شوند به یکدیگر نزدیکتر شده و وقتی که راه باز می شود اتومبیلها شروع به حرکت می کنند، به ترتیبی که فضای بیشتری بین آنها باز شود. روشهای متعددی برای تغییر متغیرها حین شبیه سازی وجود دارد.

۲۱. اعمال تغییر در مدل در حین اجرای مدل

برای اعمال تغییرات متناوب در حین شبیه سازی می توانید از ماژول منو در پانل Common استفاده کنید یا اینکه از VBA (ویژال بیسیک برای برنامه های کاربردی) در Arena استفاده نمایید، دومین روش استفاده از کنترل کننده اجرا (Arena Command-Driven Run Controller) است. اگر

چه این ابزار قدرتمندی است ولی شما باید بدانید که چگونه از SIMAN برای این منظور استفاده کنید.

۲۲. آماده سازی مدل و زمان اجرا (Warm-Up and Run Length)

ماژولهای مدل در ابتدا خالی و بی استفاده اند (Empty and Idle) یعنی تعداد موجودیت های اولیه مدل صفر است). در سیستمهای پایدار این موضوع مشکلی ایجاد نمی کند ولی در شبیه سازی های پایان پذیر شرایط اولیه مدل نقش بسیار زیادی دارند.

در ابتدای انجام شبیه سازی باید در مورد وضعیت سیستم در شرایط پایدار یا در مورد زمان اجرای آن، قاطعانه تصمیم نگیرید. بر حسب شرایط (از نقطه نظر حالت پایدار)، مقادیر اولیه را به مدلتان نسبت دهید و طول اجراهای مختلف را امتحان کنید. سپس براساس این اجراهای آزمایشی نسبت به شرایط اولیه مدل تصمیم بگیرید.

راه دیگر تعیین مقدار اولیه آن است که مقدارهای اولیه مختلفی را در مدل آزمایش کنید، به طوری که با داده های بعدی هماهنگی داشته باشد. اصولاً این کار در مدلهایی مؤثر است که تاثیر مقادیر اولیه، در طی یک اجرا به سرعت از بین می رود. در این حالت طول اجرا (زمان شبیه سازی Run Length) هنوز هم باید طولانی باشد. اما شاید نه به اندازه زمانی که شما برای تغییر مقادیر اولیه به وسیله محاسبات محض (Sheet arithmetic) نیاز دارید.

تعیین دوره آماده سازی اولیه یا Warm-Up Period در Arena بسیار ساده است. کافیست ماژول Simulate را باز کرده و در کادر متن رو بروی Warm-Up Period مقداری را قرار دهید. مدل شما (و هر همتا سازی آن Replication) هنوز مانند قبل عمل می کند. با این تفاوت که بعد از اتمام دوره آماده سازی، تمام آمار و ارقام پاک شده و گزارشها و هر داده ذخیره نشده ای در ناحیه Output، ماژول Statistics، فقط نتایج بعد از دوره آماده سازی را نشان خواهد داد. اگرچه فایل های داده ای برای نواحی Time Persistent یا Tallies در ماژول Statistics ذخیره شوند، داده های ناشی از Warm-Up Period حذف نخواهند شد. مشکلی که در اینجا با آن مواجه خواهید شد، آن است که دوره آماده سازی یا Warm-Up Period باید چقدر باشد. احتمالاً عملی ترین راه آن است که از « خروجی های کلیدی Key Output» یا از نمودارها (Plots) استفاده نماید.

۲۳. مدل خدمت و خدمات با موجودیت های برگشتی (Bulking and Reneging)

در این قسمت ابتدا با مفهوم Balk آشنا می شوید. یک موجودیت Balk یک متقاضی است که به صف نمی پیوندد و به جای دیگری می رود. به عبارت دیگر موجودیتی بخواهد وارد صفی شود ولی صف به ظرفیت معینی محدود شده باشد، در حالت Balk، موجودیت، صف را ترک کرده و نیاز خود

را در مکان دیگری بر طرف می نماید . در مقابل این حالت، وضعیت دیگری به نام وضعیت Block وجود دارد که در صورت پر بودن صف ، موجودیت منتظر می ماند تا اجازه ورود به صف را کسب نماید. اکنون مدل پیچیده ای را بررسی می کنیم که متقاضی ردد شده و بر گشتی هر دو وجود دارد. Renging متقاضی است که به صف می پیوندد و بعد آن را ترک می کند.

متقاضیان با توزیع زمانی (5) Expo وارد یک سیستم خدمت با یک خدمت دهنده می شوند. زمانی خدمت دهی به مشتریان (4,25) Expo است (واحد همه زمانها دقیقه می باشد). اگر چه مسیر انتظار ظرفیت نامحدودی دارد، ولی هر مشتری که وارد سیستم می شود ، طول خط انتظار جاری را بررسی و با ترانس و وقت خود مقایسه می کند. اگر تعداد موجودیت ها در صف (Line) بزرگتر از تحمل و ترانس متقاضی باشد، سیستم را ترک می کند. مدت زمان تحمل وانتظار مشتریان برای خالی شدن صف تا ترک سیستم ، از توزیع مثلثی (3,6,15) Tria پیروی می کند. دو مدل برای عملیات رد شدن وجود دارد. فرض می کنیم، ورودیها را ایجاد کرده، میزان ترانس را از توزیع مثلثی تولید نموده و این میزان را به ویژگی موجودیت ، با استفاده از ماژول Create اختصاص داده ایم. باید ورودیها را به ماژول Choose انتقال دهیم و ارزش نمونه خود را با استفاده از NQ با تعداد در خط انتظار (Waiting Line) مقایسه کنیم. اگر ترانس کوچکتر یا مساوی تعداد جاری در صف باشد ، ورودی را از سیستم رد می کنیم، در غیر این صورت مشتری منتظر مانده تا وارد صف شود . یک روش دیگر، تخصیص ترانس مورد نظر به یک متغیر و استفاده از آن برای ظرفیت صف مربوط به خدمت دهنده می باشد. سپس ورودی را مستقیماً به خدمت دهنده ارسال می کنیم.

اگر تعداد جاری در صف بزرگتر یا مساوی ترانس که همان ظرفیت صف است ، باشد ورودی خود به خود رد می شود. با استفاده از روش دوم، ظرفیت صف می تواند مقدار کمتری از تعداد جاری در صف داشته باشد. این روش به این سبب بکار می رود که Arena فقط ظرفیت صف را هنگام ورود موجودیت جدید به صف بررسی می کند. بنا بر این موجودیت های جاری بدون توجه به ظرفیت جدید صف باقی می مانند. برای بررسی فرآیند برگشت ، فرض می کنیم، متقاضی های ورودی قصد رد شدن ندارند ولی می توانند در بازه زمانی محدودی قبل از برگشت از صف در انتظار بمانند.

فرمان ترانس برگشت را از توزیع ارلنگ (2,15) ERLA بدست می آوریم، که معادل 30 می باشد و آنرا به یک صفت (Attribute) موجودیت در ماژول اختصاص می دهیم . مدلسازی فرآیند برگشت می تواند به منزله یک رقابت بین سیستم های رقیب باشد. اگر بتوان ورودی را وارد صف کرد و زمان برگشت فرا برسد، باید بتوانیم موجودیت را یافته و از صف خارج کنیم. در این مرحله می توان از روشهای جایگزینی مختلف استفاده کرد. مثلاً می توان متغیری را که مسیر خدمت دهنده، آماده به کار را حفظ می کند، تعریف نمود. ابتدا اولین زمان تولید موجودیت را ایجاد کرده ، آنرا

به یک صف در ماژول Create اختصاص می دهیم. سپس موجودیت را به ماژول Choose ارسال می کنیم تا برای رد شدن بررسی گردد، اگر موجودیت رد نشد، مجدداً در همان ماژول بررسی کنیم که آیا موجودیت قبل از زمان برگشت، خدمت در یافت می کند یا نه؟ اگر نه، موجودیت را برگشت می دهیم. در غیر این صورت، آنرا به ماژول ارسال می کنیم تا متغیر تعریف شده را (Update) کنیم سپس موجودیت را به صف ارجاع می دهیم.

این مدل به نظر پیچیده می آید ولی می توان آن را به فرم زیر خلاصه کرد.
 Available Time زمان آماده سازی را تعریف کنید = زمانی که خدمت دهنده بعداً آماده می شود.
 ورودی را ایجاد کنید.

Service Time زمان خدمت دهی را تخصیص دهید.
 زمانی که فعالیت برگشت می خورد و برابر زمان ترانس + زمان جاری می باشد.
 $Tolerance\ Time + TNOW$
 Balk Limit محدوده رد شدن را مشخص کنید.

اگر محدوده رد شدن (Balk Limit) < تعداد موجود در صف آنگاه موجودیت را رد کنید.
 اگر زمان برگشت (Reneg Time) > زمان آماده شدن برای خدمت (Available Time)
 آنگاه موجودیت را برگشت دهید

در غیر این صورت
 زمان آماده سازی (Available Time) = Service Time + MX(Available Time , TNOW)
 موجودیت را به صف ارسال کنید .
 MX به معنی علامت ریاضی بیشینه است.

در این روال منطقی اشکالی وجود دارد که به سادگی حل می شود، تعداد در صف صحیح نیست، چون شامل موجودیتهایی که هنوز برگشت نخورده اند، نمی شود. مشکل را با ارسال موجودیت های برگشت خورده به ماژول Delay حل می کنیم که با زمان برگشت توقف می نمایند. همچنین یک Storage با نام Renege-S تعیین می کنیم. سپس عبارت قبلی را تصحیح می نماییم.

اگر زمان و محدوده رد شدن (Balk Time) < تعداد در صف + Renege-S (NSTO)
 آنگاه موجودیت را رد کن

از طرفی تصمیم گیری در مورد برگشت موجودیت، فقط بر اساس زمان برگشت نیست، بلکه به موقعیت متقاضی در صف هم بستگی دارد. مثلاً متقاضیان ممکن بود به حدود ترانس برگشت خود دست می یافتند. ولی اکنون در ابتدای صف انتظارند و شاید فقط در انتظار خدمت باشند. موقعیت

متقاضی در صف، جایی است که برای توقف انتخاب می کند. حتی اگر زمان برگشت آن متغی شده باشد، متقاضی باید در آنجا باقی بماند. اگر محدوده توقف متقاضی ۳۰ باشد و زمان برگشت برای آن حل شود، متقاضی باید در صف به هر طریق باقی بماند تا متقاضی بعد خدمت دهی شود. این موقعیت را با یک توزیع Poisson ایجاد می کنیم (($POIS(0,75)$).

این توزیع را به دو منظور انتخاب می کنیم، اولاً چون تخمین منطقی و معقولی از این جریان کار ارایه می دهد و همچنین خروجی آن یک عدد صحیح است. ثانیاً، تاکنون از آن توزیع استفاده نکرده ایم. برای کسانی که به Poisson Tables دسترسی ندارند، تقریباً معادل توزیع تجربی داده شده است:

$$DISC(0.44, 0.827, 1.0, 0.959, 2.0, 0.993, 3.0, 0.999, 4.1, 5)$$

این فرایند تصمیم گیری جدید به این معناست که منطق مذکور خیلی معتبر نیست. اکنون باید متقاضی ورودی را در صف انتظار جا گذاری کنیم و برگشت را بعد از انقضای زمان برگشت ارزیابی نماییم. اگر واقعاً پیش رویم و متقاضی را در صف قرار دهیم، مکانیسمی برای آشکار ساختن آنکه زمان برگشت سپری شده، وجود ندارد. برای حل این مشکل یک نسخه دیگر از هر موجودیت ایجاد می کنیم و آنرا با زمان برگشت متوقف می نماییم.

نسخه اصل موجودیت، که متقاضی واقعی را نشان می دهد، بایستی به صف خدمت دهی فرستاده شود. بعد زمان برگشت از نسخه بدل، برای کنترل موقعیت صف از موجودیت اصلی استفاده می کنیم. اگر تعداد متقاضی در صف خدمت دهی خیلی زیاد نباشد، موجودیت را به صف واگذار می کنیم. اگر متقاضی هنوز در صف باشد بررسی می کنیم آیا برگشت می خورد یا خیر؟ اگر موقعیت کنونی صف در محدوده انتظار (Stay Zone) متقاضی باشد، موجودیت را واگذار می کنیم. در غیر این صورت، آنرا برای خارج کردن موجودیت Duplicate اصلی از صف خدمت دهی بکار گرفته، هردو را واگذار کنیم.

این مدل بصورت زیر قابل طرح است.

ورودیها (Arrivals) را ایجاد کنید

زمان ورود (Arrival Time) را مشخص کنید: وارد سیستم شوید

زمان برگشت (Reneg Time) را معین نمایید $ERLA(15,2)$

مقدار محدوده توقف (Stay Zone) را تعیین کنید $POIS(0.75)$

تولانس رد شدن (Bulking Tolerance) تعیین کنید $TRIA(3,6,15)$

موجودیت را در نسخه (Duplicate) ایجاد کنید

موجودیت اصلی را به صف خدمت دهی منتقل کنید

قرار گرفتن در صف را بررسی کنید

برای زمان خدمت دهی توقف کنید $EXPO(4.25)$ (Delay Time)

زمان سیستم را محاسبه کنید

براساس زمان برگشت (Reneg Time) توقف کنید

اگر موقعیت صف => مقدار محدوده توقف (Stay Zone)

آنگاه موجودیت را واگذار نمایید

متقاضیان برگشت خورده (Reneg) را محاسبه کنید

حال مدل خود را با ماژول Create که مشتریان ورودی را ایجاد می کند و مقادیر را تخصیص می دهد، شروع می کنیم. از فروریز Time Between این ماژول، گزینه EXPO(5) را انتخاب کرده و عبارت System Enter را در قسمت Make time Attribute وارد می کنیم. روی دکمه Assignment کلیک می کنیم. در پنجره فعال شده گزینه Attribute را انتخاب کرده و عبارت Renege Time را در قسمت Attribute وارد کرده و گزینه ERLA(15,2) را در فروریز Value انتخاب و OK را بزنید.

مجدداً روی Assignment کلیک کرده و گزینه Variable را انتخاب نمایید. در قسمت Variable، عبارت Server-Q Cap و در فروریز Value مقدار TRIA(3,6,15) را وارد کرده و OK را بزنید. با انتخاب مجدد Assignment، گزینه Attribute را انتخاب کنید و عبارت Stay Zone را در قسمت Attribute وارد کنید. از فروریز Value، گزینه POIS(0.75) را انتخاب کنید و OK را بزنید. مجدداً وارد پنجره Assignment شده و روی گزینه Variable کلیک کنید. Total Customers را برای جعبه متنی Variable در نظر بگیرید و عبارت Total Customer+1 را در قسمت Value وارد نمایید و OK را بزنید و نهایتاً در پنجره Assignments، گزینه Attribute را انتخاب کرده و عبارتهای Customer#، Total Customer را به ترتیب در جعبه متنی Attribute و Value وارد نمایید و OK را بزنید.

ورودیهای جدید را مستقیماً به ماژول Duplicate از پانل Support ارسال می کنیم. ماژول Duplicate را باز کرده و روی دکمه Add کلیک کنید، مقدار ۱ را در قسمت Quantity وارد کرده و OK را بزنید. این ماژول امکان ساخت نسخه های موجودیت را فراهم می کند. موجودیت اصلی ماژول را با نقطه خروج که در سمت راست ماژول، تخصیص داده می شود، ترک می کند. نسخ بدل، دقیقاً مانند اصل موجودیت است، از جمله در صفات و مقادیرشان و ...، در مقادیر خواسته شده قابل ایجاد است. اگر بیش از یک موجودیت را نسخه سازی کنید می توانید بصورت یکدسته موجودیتهایی که از یک نقطه خروج یا گروهی از موجودیتهایی که از نقاط خروج مختلف ارسال می شوند، عمل نمایید. توجه کنید، اگر ارزش n را به Quantity جهت نسخه سازی تخصیص دهید، n+1 موجودیت ماژول را ترک می کنند.

موجودیت اصلی (مشتري)، به ماژول ارسال می‌شود و سعی دارد به صف جهت انتظار برای خدمت‌دهنده وارد شود. ظرفیت این صف به عنوان متغیر Server-Q Cap وارد شده که توسط متقاضیان ورودی بعنوان تلرانس رد نشدن تنظیم می‌شود.

اگر تعداد جاری در صف کمتر از ارزش Server-Q Cap باشد، متقاضی وارد صف می‌شود، در غیر این صورت به ماژول Count رد می‌شود. تا تعداد موجودیتهای رد شده افزایش یابد و سپس به ماژول Dispose برای ترک سیستم ارسال می‌گردد. یک مشتری کسی است که وارد صف شده و برای منبع Server در انتظار می‌ماند. متقاضی که خدمت دریافت کرده به ماژول Tally جایی که در آنجا زمان سیستم محاسبه می‌گردد فرستاده می‌شود و بعد به ماژول Dispose می‌رود.

موجودیت بدل به ماژول Delay که با زمان برگشت متوقف می‌شود، رفته و به صف Renege Time در ماژول Create تخصیص می‌یابد. بعد از توقف، موجودیت وارد ماژول Assign می‌شود که ارزش صف (مشخصه) Customer # به متغیر جدیدی به نام Search # اختصاص می‌یابد. مشخصه Customer # شامل شماره منحصر به فرد مشتری است، زمانی که وارد سیستم می‌شود. سپس موجودیت به ماژول Search بعدی از پانل Blocks ارسال می‌شود. ماژول Search امکان جستجوی صف برای یافتن رتبه یا موقعیت صف، از موجودیتی که در موقعیتی تعریف شده است را فراهم می‌کند. رتبه صف با مقدار (۱) به معنی این است که موجودیت جلوی صف قرار گرفته است. در این مدل می‌خواهیم متقاضی اصلی که نسخه بدل را ایجاد کرده، جستجو را انجام دهد.

آن متقاضی ارزش یکسانی با مشخصه Customer # و متغیر Search # دارد که تخصیص دادیم. وارد ماژول SEARCH Block شده و گزینه Server-Q را از فروریز Search Item انتخاب کنید. مقدار را به قسمت Starting Index و عبارت NQ را به Ending Index و عبارت Search # = Customer # را به Search Condition تخصیص داده و OK را بزنید. معمولاً، جستجو برای صف ورودی، باید از ۱ تا NQ باشد. توجه کنید جستجو می‌تواند بطرف عقب از NQ تا ۱ نیز انجام گیرد، ولی باید در هر گستره که مل می‌خواهد جستجو شود. اگر در رده‌ای که تعداد جاری صف فراوان است، باقی بمانید، Arena با خطای زمان اجرا پایان می‌یابد.

Arena به متغیر J رتبه اولین موجودی در طول جستجو اختصاص می‌یابد. اگر موقعیت و شرایط، شامل توابع ریاضی ماکزیمم و مینیمم باشد، تمام محدوده جستجو می‌شود. اگر مشخصه‌ها در شرایط جستجو بکار گرفته شوند، مانند ارزش مشخصه‌ایی از موجودیت در صف در حال جستجو تفسیر می‌شوند. اگر صف خالی باشد، یا هیچ موجودیتی موقعیت را اشغال نکند، ارزش صفر به متغیر اختصاص می‌یابد. معمولاً شما در پی یافتن رتبه موجودیت هستید. پس باید آن موجودیت را از صف منتقل کنید (ماژول REMOVE) یا یک کپی از آن ایجاد کنید (ماژول Copy). ماژول SEARCH نیز

می تواند برای جستجوی موجودیتهایی که در یک گروه موقتی با استفاده از ماژول Batch یا GROUP قرار گرفته اند. مورد استفاده قرار گیرد.

برای این مدل می خواهیم تمام محدوده صف را برای موجودیت اصلی که ارزش Customer # یکسانی با موجودیت نسخه بدلی دارد از ۱ تا NQ جستجو کنیم. اگر موجودیت اصلی یا متقاضی در صف طولانی نباشد میزان (ارزش) متغیر J به صفر می رسد و در غیر این صورت در موقعیت متقاضی در صف قرار می گیرد. آنگاه موجودیت به ماژول Choose ارسال می شود. اولین بررسی در این ماژول، این است که ارزش متغیر J کمتر یا برابر ارزش مشخصه Stay Zone هست یا خیر. اگر این شرایط درست باشد بر این دلالت دارد که صف را ترک می کند ($J=0$)، یا موقعیت متقاضی در صف به اندازه کافی برای باقی ماندن در صف مناسب است. برای بررسی حالت دیگر، از موجودیت نسخه بدلی استفاده می کنیم. اگر این شرایط برقرار نباشد باید موجودیت اصلی را برگشت دهیم. بنابراین نسخه بدل موجودیت را به ماژول بعدی یعنی REMOVE ارسال می کنیم. ماژول REMOVE این امکان را فراهم می کند که موجودیت را از صف انتقال داده و به مکان دیگری از مدل ارسال کنیم. برای رسیدن به این منظور نیاز داریم موجودیت را برای خارج شدن و انتقال براساس ورود به صف و همچنین رتبه موجودیت تعریف کنیم.

اگر برای انتقال موجودیت از یک صف نامشخص یا انتقال موجودیت با رتبه ای که بزرگتر از تعداد موجودیتهای در صف است اصرار کنیم Arena پیغام خطا می دهد. در این مدل می توانیم متقاضی را با رتبه J از صف Server-Q منتقل کنیم. در پنجره REMOVE Block، مقدار J را به قسمت Rank of Entity اختصاص داده و گزینه Server-Q را از فروریز Queue ID انتخاب کرده و OK را بزنید. اگر ماژول REMOVE را ببینید، دو نقطه خروج را در سمت راست ماژول مشاهده می کنید. موجودیتی که وارد ماژول REMOVE شده از نقطه خروج بالایی خارج می شود و به ماژول Depart ارسال می گردد. از ماژول Depart بیشتر از سلسله ماژولهای Count، Tally و Dispose استفاده کردیم، چون قصد محاسبه زمانهای برگشت متقاضیان برگشتی را داشتیم.

همچنین ماژول Simulate را با یک طول Replication، ۲۰۰۰ واحد زمانی قرار می دهیم. اکنون مدل کامل شده است و با انتخاب دکمه Run می توانید شبیه سازی را انجام دهید. بعد از پایان شبیه سازی پانل Report را انتخاب کنید و میزان موجودیتهایی که رد می شوند را مشاهده کنید. همچنین متوسط تعداد موجودیتهای صفها را بررسی کنید و سعی کنید ارتباط بین این دو مورد را پیدا کنید.

بخش دوم : شبیه سازی پمپ بنزین

۱. مقدمه

در این بخش فرآیند و نتایج شبیه‌سازی یک پمپ بنزین که توسط نرم افزار Arena مدل‌سازی و شبیه‌سازی شده است ارائه خواهد شد. در این تحلیل سعی خواهد شد مراحل مختلف شبیه‌سازی و اعتباریابی مدل در یک سیستم تشریح شود. روند انجام پروژه به شرح ذیل می‌باشد:

ابتدا موضوع شبیه‌سازی انتخاب شد. علت انتخاب آن این بود که تقریباً آرایش تمامی پمپ بنزینها یکسان است. مسئله این است که آیا این آرایش براساس اصولی علمی بنا نهاده شده است، و یا اینکه آرایش فعلی بدون دلیل انتخاب شده است. سپس یک پمپ بنزین انتخاب و آمارگیری انجام شد. پس از آمارگیری و حذف داده‌های پرت به آنالیز ورودی پرداخته شد. سپس مدل مربوط در نرم افزار Arena ساخته شده آنالیز خروجی انجام شد.

۲. هدف این مطالعه شبیه‌سازی

هدف از این مطالعه شبیه‌سازی بررسی دو سناریو در مورد تعداد پمپ‌های جایگاه و مقایسه فاکتورهای عملکردی سیستم در هریک از این دو سناریو است.

سناریو شماره ۱: استفاده از ۴ پمپ در هر سکو

سناریو شماره ۲: استفاده از ۲ پمپ در هر سکو

مسئله پیاده‌سازی سناریوی ۱ فاکتورهای عملکردی بهتری را نتیجه خواهد داد. اما با توجه به قیمتی که هر دستگاه پمپ دارد یعنی ۲ میلیون تومان، تصمیم‌گیرندگان علاقه‌مندند که با انجام شبیه‌سازی یک تحلیل سود/منفعت در مورد منافع حاصل از بهبود فاکتورهای عملکردی در مقابل هزینه نصب پمپهای اضافی انجام دهند.

۳. مکان آمارگیری

محل‌ی که برای آمارگیری انتخاب شد پمپ بنزین بزرگراه رسالت حد فاصل پل سید خندان و میدان رسالت است. این پمپ بنزین شامل دو سکو می‌باشد که روی هر سکو چهار پمپ قرار دارد. محل استقرار وسایل نقلیه دو طرف هر سکو می‌باشد. یعنی در آن واحد هشت وسیله توانایی سوخت‌گیری را دارند

۴. زمان آمارگیری

این آمارگیری بر روی تمامی خودروهایی که در روز یکشنبه ۱۸ شهریورماه ۱۳۸۰ وارد پمپ بنزین بزرگراه رسالت شده‌اند از ساعت ۷ تا ۲۲ انجام شده است.

۵. نحوه آمارگیری

هنگامی که یک وسیله وارد صف می شود زمان یادداشت می شود. و یک کاغذ شماره دار زیر برف پاک کن قرار داده می شود. سپس هنگامی که وسیله شروع به سوخت گیری می نماید زمان سوختگیری بر روی همان برگه یادداشت می شود. و در نهایت هنگام خروج خودرو از جایگاه برگه از زیر برف پاک کن برداشته می شود و زمان خروج خودرو در آن ثبت می شود.

۶. ورود اطلاعات به کامپیوتر

در این مرحله آمار جمع آوری شده از روی برگه ها وارد صفحه گسترده Excel شد.

۷. پردازش آمار

در مرحله بعد پردازش مقدماتی روی آمار انجام شد. بدین صورت که داده هایی که به نظر اشتباه و یا پرت می آمد حذف شدند. عمده مواردی که باعث حذف اطلاعات کسب شده از یک خودرو (در حقیقت یک رکورد) می شدند به شرح زیر است.

۱. فراموشکاری آمارگیر در ثبت زمان سوختگیری یا زمان خروج
۲. زیاد بودن زمان سوختگیری که ناشی از ثبت زود هنگام زمان خدمتدهی است. یعنی آمارگیر وقتی هنوز خودرو در صف است شروع خدمتدهی را ثبت کرده است.
۳. منفی شدن زمان خدمت دهی که ناشی از بی دقتی آمارگیر در ثبت زمان شروع و پایان خدمتدهی است.

۴. زیاد بودن زمان خدمتدهی ناشی از بی دقتی در ورود اطلاعات به کامپیوتر
- نمونه ای از آمار اصلاح شده در پیوست ۱ آمده است.

۸. آنالیز ورودی

در این بخش بر روی داده های پردازش شده تحلیل آماری لازم انجام شد تا پارامترهای فرآیندهای تصادفی بکار گرفته شده در مدل تعیین شوند. این تحلیل آماری بر روی زمان بین دو ورود متوالی خودرو به پمپ بنزین و زمان خدمتدهی انجام شد. ابزار Arena برای انجام این تحلیل Input Analyzer است که به صورت یک نرم افزار مستقل هنگام نصب Arena بر روی سیستم نصب می شود. برای انجام این تحلیل ابتدا فایل داده های جمع آوری شده به فرمت متنی ذخیره شد و سپس این فایل متنی از طریق Input Analyzer فراخوانی شد. تنها با انتخاب گزینه Fit All نرم افزار

بهترین توزیع که دارای کمترین مربع خطا است را برای این داده‌ها پیدا کرده و به همراه پارامترهای تخمین زده شده برای این توزیع گزارش می‌دهد.

۱.۸. توزیع زمان بین ورودی‌ها :

تحلیل آماری لازم بر روی فایل زمان بین ورودی‌های متوالی انجام شد. این فایل با نام Entrance در دیسکت ضمیمه موجود می‌باشد. نتیجه این تحلیل آماری در جدولهای زیر ارائه شده است.

. جدول ۱ مربع خطا را برای هر یک از توزیعها نمایش می‌دهد.

جدول شماره ۱: مربع خطا مربوط به هر یک از توزیعها

تابع	مربع خطا
Lognormal	۰/۰۰۵۸۷
Gamma	۰/۰۰۷۷۸
Erlang	۰/۰۰۸۰۴
Exponential	۰/۰۰۸۰۴
Beta	۰/۰۰۹۴۳
Weibull	۰/۰۳۹۷
Normal	۰/۲۱۹
Triangular	۰/۴۰۴
Uniform	۰/۴۲۸

طبق جدول فوق توزیع انتخابی برای زمان بین ورودی‌ها LogNormal با مشخصات مندرج در جدول ۲ می‌باشد. همچنین شکل شماره ۱ نیز نمودار ورودی‌ها را نشان می‌دهد.

جدول شماره ۲: مشخصات توزیع انتخابی برای زمان بین ورودی‌ها

Distribution	Lognormal
Expression	LOGN(36.7, 124)-0.001
Square Error	0.005871



شکل شماره ۱: نمودار توزیع ورودی ها

۲.۸. توزیع زمان خدمت دهی

تحلیل آماری لازم بر روی فایل زمان خدمتدهی انجام شد. این فایل با نام Fueling در دیسکت ضمیمه موجود می باشد. نتیجه این تحلیل آماری در جدولهای زیر ارائه شده است. جدول ۳ مجموع مربعات خطاها را برای هر توزیع نشان می دهد.

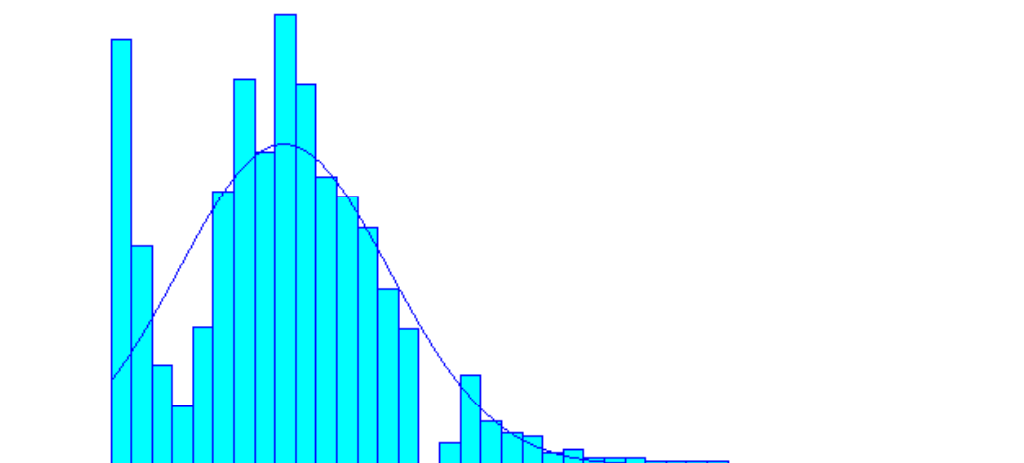
جدول شماره ۳: مربع خطا مربوط به هر یک از توزیعها

خطای استاندارد	تابع
۰/۰۱۱۸	Lognormal
۰/۰۱۹۶	Gamma
۰/۰۲۱۲	Erlang
۰/۰۲۳۶	Exponential
۰/۰۲۵۹	Beta
۰/۰۲۵۹	Weibull
۰/۰۳۳۳	Normal
۰/۰۳۵۵	Triangular
۰/۰۴۵۳	Uniform

همچنین در جدول ۴ مشخصات توزیع انتخابی (Normal) آمده است. شکل شماره ۲ نیز نمودار زمان خدمت‌دهی را نشان می‌دهد

جدول شماره ۴: مشخصات توزیع انتخابی برای زمان خدمت‌دهی

Distribution	Normal
Expression	NORM(56.1, 33.8)
Square Error	0.011766

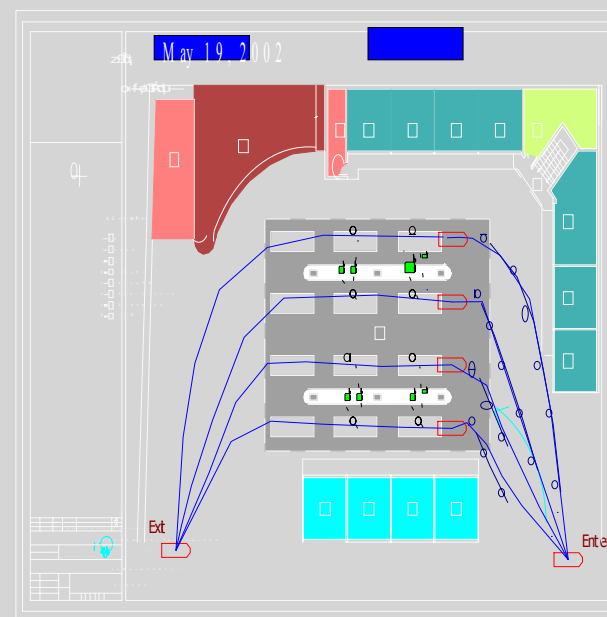
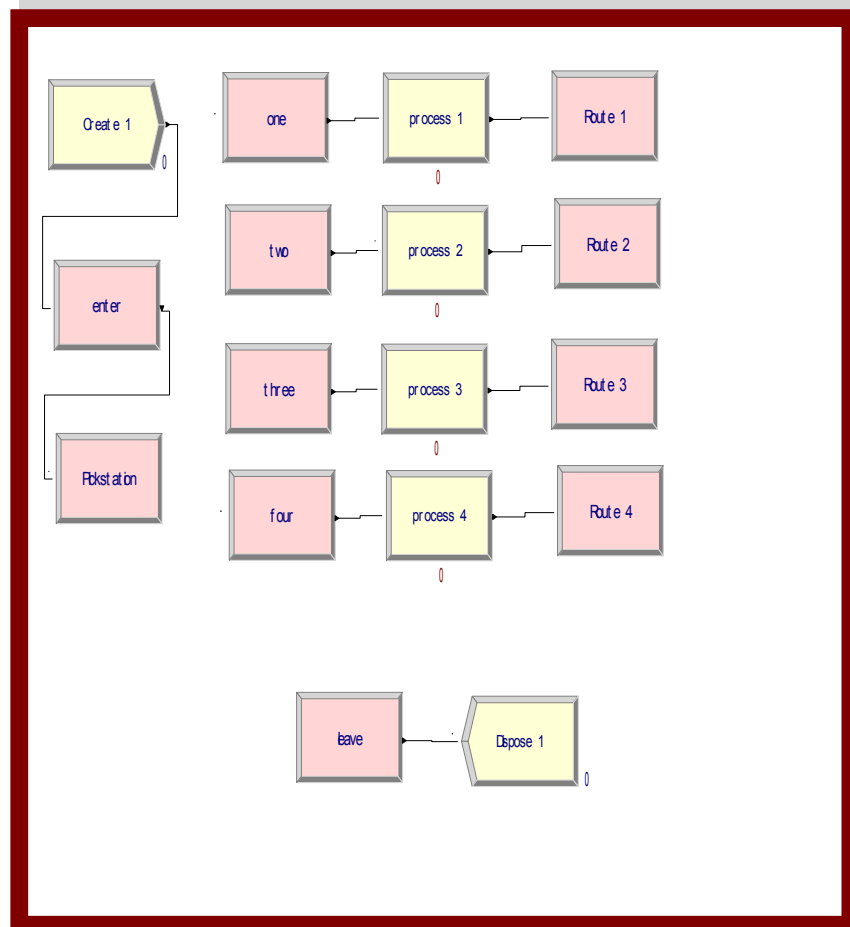


شکل شماره ۲: نمودار توزیع زمان خدمت‌دهی

۹. مدل سازی

این مدل در دیسکت ضمیمه با سه نام Senario1model، RealDatamodel و Senario2model ذخیره شده است. ساختار کلی مدل در هریک از این فایلها تقریباً یکی است. فایل Realdatamodel در بردارنده مدلی است که با استفاده از داده‌های واقعی شبیه‌سازی انجام می‌دهد. در مورد این مدل در بخشهایی بعدی بیشتر صحبت خواهد شد. فایل Senario1model در بردارنده مدلی است که بیان کننده سناریوی شماره ۱ است. فایل Senario2model در بر دارنده مدلی است که بیان کننده سناریو شماره ۲ است. شکل کلی مدلی که برای شبیه سازی پمپ بنزین تهیه شده است در شکل ۳ آورده شده است.

Sharif Service station simulation



Current number in queue:

he 1 queue:

he 2 queue:

he 3 queue:

he 3 queue:

number of cars entered till now

شکل ۳: مدل تهیه شده در نرم افزار

۱۰. تشریح مدل

مدل تهیه شده در کل از دو قسمت انیمیشن و منطق تشکیل شده است. در واقع اصل مدل در قسمت منطق قرار دارد و اجزاء انیمیشن فقط component هایی هستند که هریک متناظر با ماژولهای قسمت منطق اند. تشریح مدل را با قسمت منطق شروع می‌کنیم. اولین ماژول در این قسم ماژول Create است که موجودیت‌ها را براساس توزیع زمان بین دو ورود تولید می‌کند و وارد سیستم می‌کند. این ماژول به ماژول بعدی که یک ماژول Station است و نام آن Enter است متصل شده است. بکارگیری این ماژول برای این بوده که در قسمت انیمیشن یک icon متناظر با آن تعریف شود تا ورود خودرو به سیستم انیمیشن شود. این ماژول به ماژول بعدی که یک ماژول pickstation است متصل شده است. ماژول pickstation تعیین می‌کند که موجودیت بعد از ورود به پمپ بنزین به صف کدام خط پیوندد. منطقی که در این ماژول تعریف شده تعیین می‌کند که موجودیت صفی را که دارای کمترین موجودیت است انتخاب کند. این منطق تقریباً با منطق استفاده‌کنندگان پمپ بنزین در واقعیت مطابقت دارد. بر اساس منطق تعریف شده در pickstation موجودیت به ۴ Process که نشان دهنده خدمتدهی در ۴ خط هستند فرستاده می‌شود. برای هر پروسس در سناریو شماره ۱ دو دستگاه تلمبه به عنوان منبع تعریف شده است که اشتراکی با تلمبه‌های سایر خطها ندارند. در سناریو شماره ۲ برای هر process دو تلمبه به عنوان منبع تعریف شده است که با process مجاور مشترک هستند. هر یک از process ها صف متناظر با خود را دارند که بعنوان مثال برای process 1 به صورت process1.queue در مدل مشخص شده است. بعد از خدمت گرفتن موجودیتها در process ها آنها وارد ماژولهای rout می‌شوند موجودیتها را به سمت خروج از سیستم هدایت می‌کنند علت استفاده از این ماژولها تعریف icon های متناظر آنها در انیمیشن است تا خروج خودرو در انیمیشن نشان داده شود. این موجودیتها به سمت یک ماژول Station به نام leave هدایت می‌شوند که در انیمیشن icon متناظر با آن نقطه ای است که در آن خودرو از سیستم خارج می‌شود و ناپدید می‌شود. بعد از این ماژول، ماژول Dispose قرار دارد که موجودیتهایی را که تا سیستم را طی کرده و به آن رسیده اند را از بین می‌برد.

در قسمت انیمیشن یک فایل Autocad از پلان پمپ بنزین بصورت فایل DXF به Arena، Import شده است و به عنوان زمینه پمپ بنزین بکار گرفته شده است. موجودیتها از یک station icon که متناظر با Enter Station در قسمت منطق است وارد تصویر می‌شوند. این Station از طریق خطوط اتصال Route خودروها را به سمت خطها هدایت می‌کند. در هر خط دو محل برای خدمت‌گیری تعریف شده که یکی مربوط به پمپ جلویی و دیگری مربوط به

پمپ عقبی است. برای نشان دادن تلمبه‌ها متناظر با هر پمپ icon ای در صفحه قرار داده شده است. این آیکونها هنگام خدمتدهی به رنگ قرمز درمی‌آید و هنگام بیکاری به رنگ سبزاند. خودروها بعد از سوخت‌گیری از طریق اتصالهای Rout به Station خروجی هدایت می‌شوند و در آنجا محو می‌شوند.

۱۱. توضیح یک فرض مهم در مدل

در فرآیند سوخت‌گیری یک خودرو در جایگاه سوخت‌رسانی زمان حضور خودرو فقط از زمان انتظار در صف و زمان سوخت‌گیری تشکیل نشده است، بلکه در بار مراجعه خودرو زمان قابل توجهی صرف مبادله نقدی، خروج و برگشت به خودرو توسط راننده، خرید روغن یا فیلتر، باز کردن باک و آماده‌کردن خودرو برای حرکت می‌شود که در شبیه‌سازی این سیستم در نظر گرفته نشده‌اند. علت اصلی چشم‌پوشی از این موارد دو دلیل زیر است: اولاً اطلاعات کاملی از این موارد در آمارگیری جمع‌آوری نشده است. دوماً از تخصیص دادن یک عدد ثابت هم به ازای این موارد بعنوان یک نوع تاخیر خودداری شده است، چون با انجام این کار تعداد موجودیتهایی که در آن واحد در سیستم حضور دارند از مقدار ۱۰۰ عدد تجاوز می‌کند و نسخه دانشجویی Arena که برای این شبیه‌سازی از آن استفاده شده است پیغام خطا می‌دهد و ادامه شبیه‌سازی را ناممکن می‌سازد. به این دلایل به ناچار فقط زمان سوخت‌گیری در این شبیه‌سازی مورد استفاده قرار گرفته شده است که در نتیجه نتایجی که از شبیه‌سازی در مورد زمان انتظار در صف، طول صف و زمان حضور در سیستم بدست می‌آید از مقدار واقعی سیستم کمتر است.

۱۲. تعیین نوع شبیه‌سازی و زمان Warm-Up

چون با نزدیک شدن به ساعت ۱۰ شب نرخ ورود خودرو رو به کاهش می‌گذارد بصورتی تقریباً ساعت ۱۱ دیگر صف انتظاری وجود ندارد شبیه‌سازی که در اینجا انجام خواهد شد از نوع پایان پذیر است و زمانی که ساعت شبیه‌سازی به ۵۴۰۰۰ ثانیه می‌رسد شبیه‌سازی متوقف می‌شود. به منظور تعیین مدت Warm-up برای مدت زمانهای مختلفی مدل اجرا شد. با توجه به نتایج بدست آمده سیستم برای گرم شدن احتیاج به زمان زیادی ندارد و تقریباً ۵۰ ثانیه برای آن کافی است. این مدت زمان گرم شدن در اجرای مدلها لحاظ شده است.

جدول ۵. شبیه‌سازی سیستم با زمانهای مختلف برای تعیین زمان Warm-Up

زمان شبیه‌سازی بر حسب ثانیه	متوسط طول صف
10	0
20	0
50	3.54
100	3.54
200	2.36
300	2.8
400	3

۱۳. شبیه‌سازی مدل با استفاده از داده‌های واقعی برای اطمینان از صحت مدل و

داده‌ها

به منظور اطمینان از صحت مدل تهیه شده و داده‌های جمع‌آوری شده ، در مدل تهیه شده تغییراتی داده شد تا مدل زمان بین دو ورود و زمان خدمت را بجای اینکه خود مدل براساس توزیعهای آماری تولید کند از دو فایل متنی به نامهای Entrance و Fueling فراخوانی کند که محتویات این فایلها دادهای واقعی است که در آمارگیری میدانی جمع‌آوری شده‌اند. نتایج حاصل از این شبیه‌سازی در جدولهای زیر آمده است.

جدول شماره ۶: متوسط زمان انتظار در صف برای داده‌های واقعی

	Waiting Time
<i>Line 1.Queue</i>	48.74
<i>Line 2.Queue</i>	62.08
<i>Line 3.Queue</i>	57.18
<i>Line 4.Queue</i>	56.88
<i>Total</i>	224.89

جدول شماره ۷: متوسط تعداد وسائل در صف برای داده‌های واقعی

	Number Waiting
<i>Line 1.Queue</i>	0.52
<i>Line 2.Queue</i>	0.89
<i>Line 3.Queue</i>	0.73
<i>Line 4.Queue</i>	0.65
<i>Total</i>	2.79

جدول شماره ۸: متوسط زمان خدمت دهی و زمان بودن در سیستم برای داده‌های واقعی

	S. Time	Wait Time	Total
<i>Line 1</i>	49.98	48.74	98.73
<i>Line 2</i>	51.48	62.08	113.56
<i>Line 3</i>	50.22	57.18	107.4
<i>Line 4</i>	52.1	56.88	108.98
<i>Total</i>	203.78	224.89	428.67

جدول شماره ۹: کل زمان بودن در صف و بودن در سیستم برای داده‌های واقعی

	Accum S. Time	Accum Wait Time
<i>Line 1</i>	29641	28904
<i>Line 2</i>	41283	49792
<i>Line 3</i>	36207	41228
<i>Line 4</i>	33398	36458
<i>Total</i>	140529	156382

جدول شماره ۱۰: کل استفاده کنندگان برای داده‌های واقعی

	Number In	Number Out
<i>Line 1</i>	593	593
<i>Line 2</i>	802	802
<i>Line 3</i>	721	721
<i>Line 4</i>	641	641
<i>Total</i>	2757	2757

متوسط زمان سوخت‌گیری برمیانگین گیری از داده‌های جمع‌آوری شده ۶۸ ثانیه است نتایج حاصل از این شبیه‌سازی نیز ۵۰ ثانیه برای هر موجودیت است. براساس مشاهده ای که از سیستم در هنگام آمارگیری انجام شد تقریباً بطور متوسط ۳ الی ۴ خودرو در صف انتظار هر خط حضور داشتند. براساس این شبیه‌سازی تعداد خودرو تقریباً ۱ خودرو در هر خط است. علت کمتر بودن تعداد در صف در هنگام شبیه‌سازی نسبت به مشاهده واقعی همان فرض مهمی است که در قسمتهای بالاتر به آن اشاره شد. با توجه به این فرض اختلاف بین این دو داده توجیه‌پذیر است.

در کل با توجه به نتایج بدست آمده از این اجرای شبیه‌سازی و مشاهداتی که از سیستم واقعی هنگام جمع آوری اطلاعات انجام شده است مورد غیر عادی در خروجی‌ها دیده نمی‌شود و دلیلی بر نادرست بودن مدل وجود ندارد.

۱۴. تعیین تعداد دوباره‌سازی ها برای شبیه‌سازی سناریو شماره ۱

با توجه به اینکه در قسمت مقایسه سناریوها قصد داریم بازه اطمینانی ۹۰٪ ای با دقت ۱ ثانیه برای مدت زمان حضور موجودیت در سیستم ارائه کنیم، باید تعداد دوباره‌سازی ها را براساس این بازه اطمینان و دقت تعیین کنیم. به این منظور ۱۰ بار دوباره سازی آزمایشی برای تخمین واریاس نمونه انجام دادیم که متوسط زمان حضور موجودیت در سیستم برای هریک از این دوباره‌سازی‌ها در جدول زیر آمده است:

جدول شماره ۱۱: ۱۰ بار شبیه سازی آزمایشی

Average Entity Total Time in the System	
Replication	Senario 1
1	67.33
2	68.64
3	68.1
4	68.4
5	65.68
6	66.95
7	69.31
8	69.62
9	68.99
10	68.98

$$S_{10}=1.48826667$$

طبق فرمول دوباره سازی ها کوچکترین مقداری از R است که در رابطه زیر صدق کند:

$$R \geq \left(\frac{t_{\alpha/2,10} S_{10}}{\varepsilon} \right)$$

که ε نشان دهنده خطای قابل پذیرش است

$$R \geq 11.33 \quad \Rightarrow \quad R=12$$

۱۵. شبیه سازی سیستم براساس سناریوی شماره ۱

مدل تهیه شده برای این سناریو به تعداد ۱۲ بار بصورتی که در بالا تعیین شد به مدت ۵۴۰۰۰ ثانیه یعنی از ساعت ۷ صبح تا ۱۰ شب شبیه سازی شد که نتایج این شبیه سازی در زیر آمده است. لازم به ذکر است که این تعداد بار دوباره سازی براساس بازه اطمینان و دقت مورد نیاز برای متوسط زمان حضور موجودیت در سیستم تعیین شده است. ارائه سایر معیارهای عملکرد سیستم فقط برای شناسایی بهتر سیستم است و به تحلیل نهایی ارتباطی ندارند.

جدول شماره ۱۲: متوسط زمان حضور موجودیت در سیستم در سناریو شماره ۱

	VA time	Wait time	Total time
Entity	56.68	9.56	68.2

جدول شماره ۱۳: متوسط زمان بودن در صف در سناریو شماره ۱

	Waiting Time
<i>Line 1.Queue</i>	2.68
<i>Line 2.Queue</i>	11.20
<i>Line 3.Queue</i>	7.56
<i>Line 4.Queue</i>	5.19
<i>Total</i>	26.63

جدول شماره ۱۴: متوسط تعداد وسائل در صف در سناریو شماره ۱

	Number Waiting
<i>Line 1.Queue</i>	0.00103
<i>Line 2.Queue</i>	0.1841
<i>Line 3.Queue</i>	0.06287
<i>Line 4.Queue</i>	0.01319
<i>Total</i>	0.26119

جدول شماره ۱۵: متوسط تعداد خدمتدهی‌ها توسط هریک از خطها در سناریو شماره ۱

	Average Number Out
<i>Line 1</i>	24
<i>Line 2</i>	918
<i>Line 3</i>	462
<i>Line 4</i>	140
<i>Total</i>	1544

جدول شماره ۱۶: متوسط درصد استفاده از پمپها در سناریو شماره ۱

<i>Resource</i>	Perc Utilization	<i>Resource</i>	Perc Utilization
<i>Pump1</i>	0.0056	<i>Pump5</i>	0.1981
<i>Pump2</i>	0.4312	<i>Pump6</i>	0.04959
<i>Pump3</i>	0.0195	<i>Pump7</i>	0.2709
<i>Pump4</i>	0.4971	<i>Pump8</i>	0.0900

۱۶. تعیین تعداد دوباره‌سازی ها برای شبیه‌سازی سناریو شماره ۲

با توجه به اینکه در قسمت مقایسه سناریوها قصد داریم بازه اطمینانی ۹۰٪ ای با دقت ۱ ثانیه برای مدت زمان حضور موجودیت در سیستم ارائه کنیم. باید تعداد دوباره‌سازی ها را براساس این بازه اطمینان و دقت تعیین کنیم. به این منظور ۱۰ بار دوباره سازی آزمایشی برای تخمین واریانی نمونه انجام دادیم که متوسط زمان حضور موجودیت در سیستم برای هریک از این دوباره‌سازی‌ها در جدول زیر آمده است:

جدول شماره ۱۷: ۱۰ بار شبیه‌سازی آزمایشی

Average Entity Total Time in the System	
Replication	Senario 1
1	75.79
2	76.13
3	81.57
4	81.54
5	74.33
6	74.83
7	80.22
8	82.39
9	81.25
10	81.2494

$$S_{10}=10.4276685$$

طبق فرمول دوباره‌سازی ها کوچکترین مقداری از R است که در رابطه زیر صدق کند:

$$R \geq \left(\frac{t_{\alpha/2,10} S_{10}}{\varepsilon} \right) \quad \text{که } \varepsilon \text{ نشان دهنده خطای قابل پذیرش است.}$$

$$R \geq 556.44 \quad \Rightarrow \quad R=557$$

۱۷. شبیه سازی سیستم براساس سناریوی شماره ۲

مدل تهیه شده برای این سناریو به تعداد ۵۵۶ بار بصورتی که در بالا تعیین شد به مدت ۵۴۰۰۰ ثانیه یعنی از ساعت ۷ صبح تا ۱۰ شب شبیه سازی شد که نتایج این شبیه سازی در زیر آمده است. لازم به ذکر است که این تعداد بار دوباره سازی براساس بازه اطمینان و دقت مورد نیاز برای متوسط زمان حضور موجودیت در سیستم تعیین شده است. ارائه سایر معیارهای عملکرد سیستم فقط برای شناسایی بهتر سیستم است و به تحلیل نهایی ارتباطی ندارند.

جدول شماره ۱۸: متوسط زمان حضور موجودیت در سیستم در سناریو شماره ۲

	VA time	Wait time	Total time
<i>Entity</i>	58.58	20.34	78.92

جدول شماره ۱۹: متوسط زمان بودن در صف در سناریو شماره ۲

	Waiting Time
<i>Line 1.Queue</i>	58.02
<i>Line 2.Queue</i>	15.89
<i>Line 3.Queue</i>	15.60
<i>Line 4.Queue</i>	52.56
<i>Total</i>	142.07

جدول شماره ۲۰: متوسط تعداد وسائل در صف در سناریو شماره ۲

	Number Waiting
<i>Line 1.Queue</i>	0.0699
<i>Line 2.Queue</i>	0.2566
<i>Line 3.Queue</i>	0.1286
<i>Line 4.Queue</i>	0.1082
<i>Total</i>	0.5633

جدول شماره ۲۱: متوسط تعداد خدمتدهی‌ها توسط هریک از خط‌ها در سناریو شماره ۲

	Average Number Out
<i>Line 1</i>	24
<i>Line 2</i>	918
<i>Line 3</i>	462
<i>Line 4</i>	140
<i>Total</i>	1544

جدول شماره ۲۲: متوسط درصد استفاده از پمپ‌ها در سناریو شماره ۲

<i>Resource</i>	Perc Utilization
<i>Pump1</i>	0.4610
<i>Pump3</i>	0.5171
<i>Pump5</i>	0.2575
<i>Pump7</i>	0.3185

۱۸. مقایسه دو سناریو

به منظور بررسی اینکه آیا اضافه کردن پمپ به سیستم تغییر قابل ملاحظه‌ایی در فاکتورهای عملکردی سیستم اعمال خواهد کرد بازه‌های اطمینان برای چند فاکتور عملکردی سیستم تهیه شده است.

براساس ۱۰ بار شبیه‌سازی مدل سناریو شماره ۱ و ۱۰ بار مدل سناریو شماره ۲ که اطلاعات آن در زیر آمده است ، بازه اطمینانی با ۹۰٪ اطمینان محاسبه شد که در زیر نشان داده شده است.

$$(\bar{Y}_2 - \bar{Y}_1) \pm t_{\alpha/2; v} s.e. (\bar{Y}_2 - \bar{Y}_1)$$

جدول شماره ۲۳: دوباره سازی های سناریو ۱ و سناریو ۲

Average Entity Total Time in the System		
Replication	Senario 1	Senario 2
1	67.33	75.79
2	68.64	76.13
3	68.1	81.57
4	68.4	81.54
5	65.68	74.33
6	66.95	74.83
7	69.31	80.22
8	69.62	82.39
9	68.99	81.25
10	68.98	81.2494
11	68.65	76.44
12	67.9	75.91
	—	76.87
556	—	79.09

جدول ۲۴. محاسبه بازه اطمینان برای اختلاف میانگینهای زمان حضور در سیستم برای سناریوها

انحراف معیار نمونه ۱	1.48826667
انحراف معیار نمونه ۲	10.4276685
خطای معیار	3.33093392
درجه آزادی	9.44795006
معکوس توزیع تی	7.03E-01
میانگین نمونه ۱	68.2
میانگین نمونه ۲	78.92994
اختلاف میانگین نمونه ۱ و ۲	10.72994 2.34
بازه اطمینان ۹۰٪ برای اختلاف میانگین نمونه ۱ و ۲	(13.12994 , 8.38994)

۱۹. بحث و نتیجه گیری

با توجه به بازه اطمینانی که برای اختلاف زمان حضور خودرو در سیستم در سناریوها بدست آمده معلوم می شود که در سناریوی شماره ۲ خودرو تقریباً ۱۵٪ زمان بیشتری را نسبت به سناریو شماره ۱ در سیستم سپری می کند که زمان قابل توجهی است. از طرفی با توجه به فرض بسیار مهمی که در این شبیه سازی کرده ایم یعنی در نظر نگرفتن زمانهایی مثل مبادله نقدی، مدت زمانی که خدمت گیرنده در سیستم سپری می کند در سناریو اول نسبت به سناریو دوم مسلماً بیشتر از مقدار ذکر شده در اینجاست. پیشبینی می شود چنین تفاوتی در زمان حضور در سیستم به مقدار قابل ملاحظه ای رضایت مشتریان را در استفاده از این پمپ بنزین در حالت سناریو شماره ۲ کاهش دهد و به تبع آن مشتری های این پمپ بنزین کاهش پیدا می کنند. با توجه به تعداد قابل توجهی از خودروها، تقریباً بیشتر از ۵۰۰۰ خودرو، که در طی روز به این پمپ بنزین مراجعه می کنند و قیمت اضافه کردن هر پمپ یعنی ۲ میلیون تومان با اطمینان زیادی می توان گفت که پیاده سازی سناریو شماره ۱ منافع بیشتری برای اداره کننده پمپ بنزین در برخواهد داشت.

پیوست ۱

نمونه ای از آمار

زمان سوختگیری	زمان بین دو ورود	خروج	میزان سوخت گیری	پایان سوخت گیری	شروع سوخت گیری	ورود به جایگاه
0:00:37	0:02:57	6:21:18	30	6:20:27	6:19:50	6:13:40
0:00:52	0:02:23	6:22:30	15.51	6:21:22	6:20:30	6:16:37
0:03:12	0:01:15	6:25:13	25.02	6:24:31	6:21:19	6:19:00
0:01:13	0:00:03	6:26:24	34.3	6:25:36	6:24:23	6:20:15
0:01:10	0:01:20	6:26:28	30.1	6:25:34	6:24:24	6:20:18
0:00:46	0:00:46	6:23:36	30.3	6:22:38	6:21:52	6:21:38
0:00:52	0:00:25	6:26:20	23.303	6:25:32	6:24:40	6:22:24
0:01:15	0:00:28	6:28:29	20.04	6:27:00	6:25:45	6:22:49
0:01:00	0:00:37	6:28:15	20.02	6:28:10	6:27:10	6:23:17
0:00:59	0:00:21	6:28:12	31.11	6:28:10	6:27:11	6:23:54
0:01:00	0:04:05	6:31:10	13.03	6:30:08	6:29:08	6:24:15
0:01:38	0:00:09	6:34:16	33	6:33:50	6:32:12	6:28:20
0:02:14	0:00:08	6:34:20	42.57	6:33:16	6:31:02	6:28:29
0:02:08	0:00:47	6:34:10	27.19	6:33:38	6:31:30	6:28:37
0:01:06	0:01:35	6:35:40	22.2	6:34:55	6:33:49	6:29:24
0:01:11	0:02:36	6:43:45	22.4	6:43:27	6:42:16	6:30:59
0:00:37	0:00:01	6:36:47	15.1	6:35:53	6:35:16	6:33:35
0:00:53	0:02:23	6:46:24	25.55	6:45:22	6:44:29	6:33:36
0:01:45	0:00:22	6:47:15	29.15	6:47:10	6:45:25	6:35:59
0:01:18	0:00:02	6:47:13	27.99	6:46:38	6:45:20	6:36:21
0:00:40	0:00:07	6:43:51	31.02	6:43:25	6:42:45	6:36:23
0:01:00	0:01:25	6:40:22	24.02	6:37:40	6:36:40	6:36:30
0:01:10	0:00:08	6:49:23	31.99	6:49:23	6:48:13	6:37:55
0:01:17	0:00:46	6:41:52	30	6:39:37	6:38:20	6:38:03
0:01:00	0:00:04	6:50:14	25.51	6:48:39	6:47:39	6:38:49
0:01:00	0:00:37	6:44:00	18	6:42:05	6:41:05	6:38:53
0:01:29	0:01:03	6:45:57	35.41	6:43:49	6:42:20	6:39:30
0:01:09	0:00:30	6:47:41	31.01	6:45:42	6:44:33	6:40:33
0:01:03	0:00:12	6:52:11	30	6:51:40	6:50:37	6:41:03
0:01:00	0:01:02	6:52:13	39.54	6:51:35	6:50:35	6:41:15

6:42:17	6:46:20	6:47:40	30	6:48:09	0:00:43	0:01:20
6:43:00	6:45:00	6:46:38	30	6:47:43	0:01:16	0:01:38
6:44:16	6:48:22	6:49:59	26	6:50:44	0:00:15	0:01:37
6:44:31	6:52:33	6:53:02	17.74	6:54:09	0:00:50	0:00:29
6:45:21	6:52:43	6:53:37	30.06	6:54:05	0:02:03	0:00:54
6:47:24	6:51:20	6:53:05	28.9	6:53:46	0:01:41	0:01:45
6:49:05	6:52:38	6:54:40	49.98	6:55:40	0:01:39	0:02:02
6:50:44	6:56:29	6:57:12	20.01	6:59:41	0:00:13	0:00:43
6:50:57	6:56:17	6:57:15	30	6:58:04	0:00:05	0:00:58
6:51:02	6:55:40	6:56:22	20.3	6:57:53	0:00:20	0:00:42
6:51:22	6:56:22	6:56:58	30.02	6:58:06	0:00:28	0:00:36
6:51:50	6:54:36	6:55:42	18.1	6:57:13	0:00:43	0:01:06
6:52:33	7:00:25	7:01:57	30	7:02:45	0:00:20	0:01:32
6:52:53	6:57:42	6:58:35	20.01	7:00:15	0:00:27	0:00:53
6:53:20	6:53:49	6:54:34	25	6:55:20	0:00:21	0:00:45
6:53:41	6:58:34	6:59:32	40.01	7:00:30	0:00:58	0:00:58
6:54:39	6:58:45	6:59:58	27.95	7:01:23	0:00:18	0:01:13
6:54:57	7:01:35	7:02:38	19.97	7:03:40	0:02:31	0:01:03
6:57:28	7:02:47	7:03:47	21.1	7:04:23	0:00:21	0:01:00
6:57:49	7:01:30	7:02:20	30.01	7:04:24	0:00:10	0:00:50
6:57:59	7:02:01	7:04:05	32.83	7:04:33	0:00:03	0:02:04
6:58:02	7:03:24	7:04:19	19.99	7:05:23	0:00:30	0:00:55
6:58:32	7:04:23	7:05:21	31	7:05:42	0:01:07	0:00:58
6:59:39	7:03:23	7:05:08	39.99	7:06:40	0:00:15	0:01:45
6:59:54	7:06:23	7:08:39	34.2	7:09:36	0:00:14	0:02:16
7:00:08	7:05:05	7:06:19	20	7:08:39	0:00:30	0:01:14
7:00:38	7:05:18	7:07:18	48.91	7:07:41	0:00:10	0:02:00
7:00:48	7:05:31	7:06:25	33	7:06:54	0:00:01	0:00:54
7:00:49	7:04:55	7:05:34	30.02	7:06:19	0:00:16	0:00:39
7:01:05	7:05:07	7:06:22	38.83	7:07:01	0:00:18	0:01:15
7:01:23	7:09:40	7:10:29	15.05	7:11:35	0:00:38	0:00:49

منابع:

1. Kelton ,David.Randall P.Sadowski , Deborah A.Sadowsky.Simulation with Arena,2002,McGrawHill
 2. www.arenasimulation.com
 3. www.acm.org/digitallibrary Winter Simulation Conference Proceedings
۴. بنکس، جری. جان کارسن. شبیه سازی سیستم‌های گسسته پیشامد، ۱۳۷۶، ترجمه هاشم محلوچی، انتشارات دانشگاه صنعتی شریف

تماس:

Kazem Yaghootkar

Email: yaghootkar@yahoo.com

yaghootkar@ppars.com

Homepage: <http://www.geocities.com/yaghootkar>

Tel : 09132366795

