Univeridade de coimbra

Licenciatura em engº informática

Simulação e computação científica

Estudo do funcionamento de uma estação de serviço

**Introdução**

Foi pedido, como âmbito da disciplina de Simulação e Computação Científica, a criação de um programa que simulasse o comportamento de uma Estação de Serviço por forma a analisar esse comportamento num abiente controlado e também detectar possíveis melhorias no sistema. A estação de serviço é inicialmente composta por 3 bombas (duas de gasolina e uma de gasóleo) e uma loja onde se efectua o pagamento após o abastecimento. A estação funciona 24 horas por dia e tem 12 trabalhadores (escalonados em turnos de 8 horas) sendo que esses turnos são constituídos sempre por um elemento na loja e três nas bombas.

Desde logo, estudando no papel a taxa de utilização de cada serviço, percebemos que o problema nesta estação de serviço era a elevada taxa de utilização das bombas de gasolina. Foram-nos então propostas duas soluções:

- Acrescentar um novo posto de abastecimento de gasolina, que necessitava de mais um funcionário por turno e elevava o número total de funcionários da estação para 15.

- Substituir os postos de abastecimento por um novo sistema constituído por quatro postos self-service multifuncionais. Este sistema não só permite dispender da loja para o pagamento como também diminui o número de funcionários da estação. Para este cenário apenas são necessários 2 funcionários por turno para supervisionar a estação de serviço, o que nos permite ter apenas 6 funcionários a trabalhar na estação. É por outro lado necessário para esta solução, um investimento maior e os tempos de abastecimento tornam-se mais lentos.

Criámos um programa com um ambiente gráfico simples de forma a tornar possível simular casos de teste diferentes, que nos permitissem chegar a uma conclusão sobre a melhor abordagem a seguir. Tivemos também que validar o nosso simulador, tanto internamente como recorrendo a outro simulador de confiança, o GPSS World.



**Arquitectura do simulador (classes)**

* Simulador – Classe onde está guardada a lista de eventos, assim como a métida de chegada e os serviços existentes, neste caso gasolina, gasóleo, loja e ainda self-service. É nesta classe que é inicializado o ambiente gráfico e a restante simulação.
* Transicao – Classe que , utilizando o método “executa”, faz a transição entre o posto de Gasolina para Loja.
* Evento – Classe abstracta que contém os métodos comuns a todo o tipo de eventos (por exemplo o método executa já falado em cima).
* Serviço (deriva da classe Evento)– Esta classe tem os métodos que permitem inserir/remover um cliente num determinado serviço (através de uma fila de espera), e ainda métodos que calculam os valores estatísticos referentes aos clientes do serviço
* Chegada/Saida – Classes que representam respectivamente a chegada e saída de um cliente a um serviço.
* Aleatorio – Classe onde são gerados números aleatórios através de diferentes funções matemáticas, como uma distribuição exponencial negativa ou destribuição normal.

**Validação**

**a) Obter valores estáveis**

Para podermos validar o nosso simulador, precisamos de chegar a um valor de tempo que nos permita obter resultados consistentes, para que os valores que obtenhamos não dependam do tempo. Para achar esse valor, começamos por testar com um valor bastante grande, para o qual assumimos que os resultados obtidos estão estabilizados. Testámos portanto para um valor de 12 000 minutos, equivalente a um tempo de simulação de 200 horas, o que corresponde, aproximadamente, a 8.3 dias. Estes foram os resultados obtidos:

(print do simulador para 12 000 minutos)

A partir daí, fomos correndo o simulador com tempos de hora a hora, até obter valores próximos deste. É importante notar que são usadas as mesmas seeds para as diferentes simulações, de modo a garantir que os resultados podem ser relacionados entre si:

Simulação para 60 minutos:

(print do simulador para 60 min)

(VAI ANOTANDO OS RESULTADOS (TEMPO, TAXA DE UTILIZACAO DE CADA COISA) NO EXCELL PARA NO FIM FAZER UM GRAFICO!)

Simulação para 120 minutos:

(print do simulador para 120 min)

Simulação para 180 minutos:

(print do simulador para 180 min)

Simulação para 240 minutos:

(print do simulador para 240 min)

Simulação para 300 minutos:

(print do simulador para 300 min)

Simulação para 360 minutos:

(print do simulador para 360 min)

Simulação para 420 minutos:

(print do simulador para 420 min)

Simulação para 480 minutos:

(print do simulador para 480 min)

Simulação para 540 minutos:

(print do simulador para 540 min)

Simulação para 600 minutos:

(print do simulador para 600 min)

Podemos observar, no gráfico que se segue, que com 600 minutos obtemos resultados muito próximos aos resultados obtidos na simulação com 10 000 minutos. Consideramos então que os resultados estabilizam com um tempo de simulação de 600 minutos.

(GRÁFICO COM OS RESULTADOS DA SIMULAÇÃO)

**b) Validação Interna**

Para fazermos a validação interna, optámos por usar o método das réplicas. Este método consiste em correr simulações com os mesmos valores mas seeds diferentes. Caso os resultados obtidos sejam semelhantes, podemos considerar que o nosso simulador está validado internamente.

(FAZER ISTO NO SÁBADO)

**c) Comparação com valores teóricos**

Uma maneira de validarmos o nosso simulador é calcular matemáticamente os valores que seriam esperados tendo em conta as condições simuladas.

**d) Validação Externa**

A validação externa consiste em compararmos o nosso simulador com um simulador já existente e de fiabilidade atestada, como é o caso do GPSS World. Para isso, criámos uma simulação com as mesmas condições, cujo código segue em anexo.

Verificamos que os valores obtidos são semelhantes. Como partimos do princípio que o GPSS World obtém resultados fiáveis, chegamos também à conclusão de que o nosso simulador em Java apresenta resultados correctos e fidedignos.

(POR PRINT AQUI DO RESULTADO DO GPSS WORLD)

**e) Conclusão**

O nosso simulador passou os testes feitos nas alíneas anteriores. Dizemos, portanto, que o nosso simulador está validado e obtém resultados fidedignos e que podemos usar para simular as diferentes condições na estação de serviço.

**Simulação**

**a) Cenário Inicial**

Agora que o nosso simulador está devidamente validado, podemos correr o cenário inicial, em que temos 2 funcionários na bomba de gasolina, um funcionário na bomba de gasóleo e 1 funcionário na loja. Sabe-se que os clientes chegam em média com

um intervalo de tempo entre chegadas que segue uma distribuição exponencial negativa de média 1.2 minutos. 20% dos clientes pretende abastecer gasóleo e os restantes gasolina.

Essa operação demora em ambos os casos em média 4 minutos com desvio padrão de 2.5 minutos segundo uma distribuição normal.

Simulamos então este cenário com um tempo de simulação de 600 minutos, obtendo os seguintes resultados:

(SIMULAÇÃO COM 600 minutos)

--> Escrever conclusão sobre dados obtidos

**b) Soluções**

No enunciado deste projecto, são nos dadas duas soluções para este problema, cada uma delas com os seus prós e contras:

**i) Acrescentar um novo posto de gasolina**

**ii) Substituir todos os postos por um novo sistema constituído por quatro postos self-service multifuncionais**

**Conclusão**

**c) ALINEA D**