



Lucas Begnini - 1015080103  
Kevin Gustavo Guispe - 0825080229

Simulador de Lava a Jato

UNIVERSIDADE DO ESTADO DO AMAZONAS  
UEA - AM

Lucas Begnini - 1015080103  
Kevin Gustavo Guispe - 0825080229

## Simulador de Lava Jato

Esse Trabalho, proposto pelo Profº Rodrigo Choji, da disciplina de Simulação de Sistema, tem como intuito aprimorar o conhecimento a respeito do uso de simulador através da implementação de um simulador de lava a jato.

Manaus - AM  
October 4, 2013

## **1 Introdução e justificativa**

Um grande crescimento da necessidade de simular ambientes com um baixo custo gerou o objetivo desse trabalho, e nele iremos tentar abordar a simulação de um lava jato. Claro, em um ambiente computacional, jamais será 100 por cento igual ao ambiente real, mas o que a matéria pretende é chegar ao mais perto possível, para tentar prever e resolver problemas que muitas vezes só enxergamos se colocarmos em prática, e criando um simulador é uma boa abordagem pra tentar colocar em prática com um baixo custo.

## **2 Objetivos**

Nesse trabalho iremos gerar um simulador de um posto de lavagem onde simularemos as entradas e saídas de carros, e até mesmo as perdas de clientes que chegaram e não puderam ficar devido ao tamanho da fila. A criação do simulador é para poder ter uma idéia da circulação de carros dentro de um posto de lavagem. E ao final propor algumas modificações dentro das variáveis controladas do sistema(fila, quantidade de máquinas de lavagem) para que o mesmo atinja o ponto ótimo, apresentando ao usuários possíveis cenários de melhorias.

## **3 Etapas do desenvolvimento**

Todo o projeto foi desenvolvido em etapas. Ou seja, por partes, para melhor desenvolvimento do projeto. Ao todos foram feitas 4 etapas: Levantamento de requisitos, análise de ferramentas utilizadas para o desenvolvimento, desenvolvimento e análise dos dados obtidos. Ao longo de cada etapa haverá maior detalhamento.

### 3.1 Etapa 1: Levantamento de Requisitos

Conforme os dados apresentados pelo professor, dividimos os cenários ao total em 2:

- Cenário 1
- Cenário 2

#### 3.1.1 Cenário 1

Um cenário em que o tempo entre chegadas é de 45 min no máximo, e que o tempo de serviço é de 15 minutos no máximo. Considerando que o expediente é de 10 horas. Para essas simulações foram gerados gráficos e histogramas que serão apresentados logo mais.

#### 3.1.2 Cenário 2

Para esse cenário foi implementada o método de *Monte Carlo*. Mas para isso é necessário *à priori* a simulação de um sistema, onde para tal foram utilizados os mesmos critérios do cenário 1, ou seja, tempo de chegada de no máximo 45 min e tempo de serviço também no máximo 15 min. Fazendo isso 10000 vezes e guardando os resultados em uma tabela para depois utilizar o GNA(Gerador de Numeros Aleatórios) para capturar dentro dessa tabelas os valores médios a serem usados durante a simulação. Todas essas tabelas e valores vão ser mostrado logo mais.

### 3.2 Etapa 2: Ferramentas utilizadas para o Desenvolvimento

Em uma conversa *à priori* chegamos a conclusão que seria necessário a utilização das seguintes ferramentas:

- Astah
- Eclipse

#### 3.2.1 Astah

O Astah é uma poderosa ferramenta de modelagem, onde nele se foi possível enxergar todas as classes necessárias para o desenvolvimento do

simulador.

### 3.2.2 Eclipse

Ferramenta de desenvolvimento de programas, o eclipse é hoje uma das IDEs mais utilizadas, principalmente quando o assunto é *JAVA*, que no nosso caso foi a linguagem utilizada para desenvolvimento do simulador.

## 3.3 Etapa 3: Analise de resultados

### 3.3.1 Cenário 1

Utilizando-se dos simulador criado em trabalhos anteriores dessa mesma matéria e os critérios para tempo de chegada e lavagem já falados anteriormente nesse trabalho, simulamos o sistema e obtivemos a seguinte tabela de tempo de Chegadas:

Classe ID	Frequencia	Frequencia Relativa	Frequencia Acumulada	Ponto.Medio
0--9	136	0,1878453	0,1878453	5
10--19	135	0,18646409	0,37430939	15
20--29	146	0,20165746	0,57596685	25
30--39	159	0,21961325	0,7955801	35
40--49	137	0,18922652	0,98480662	45

Figure 1: Tabela de Frequencia TEC

Essa tabela nos mostra a frequência de chegada em cada uma das classes definidas por nós. Como por exemplo, entre 0 e 9 minutos chegaram 136 carros. Ou seja, nessas simulações feitas 136 carros chegaram entre 0 e 9 minutos. Analogamente podemos observar a seguinte tabela de tempo de serviço:

0--3	132	0,18232045	0,18232045	2
4--7	223	0,30801105	0,4903315	6
8--11	177	0,24447514	0,73480664	10
12--15	192	0,26519337	1,00000001	14

Figure 2: Tabela de Frequencia TS

Com isso Conseguimos fazer o histograma tanto do tempo de Chegada, quanto do tempo de serviço que você pode observar a seguir:

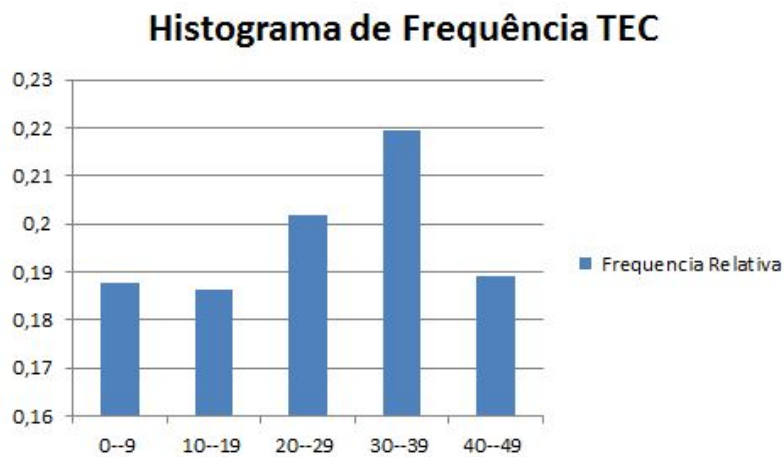


Figure 3: Histograma TEC

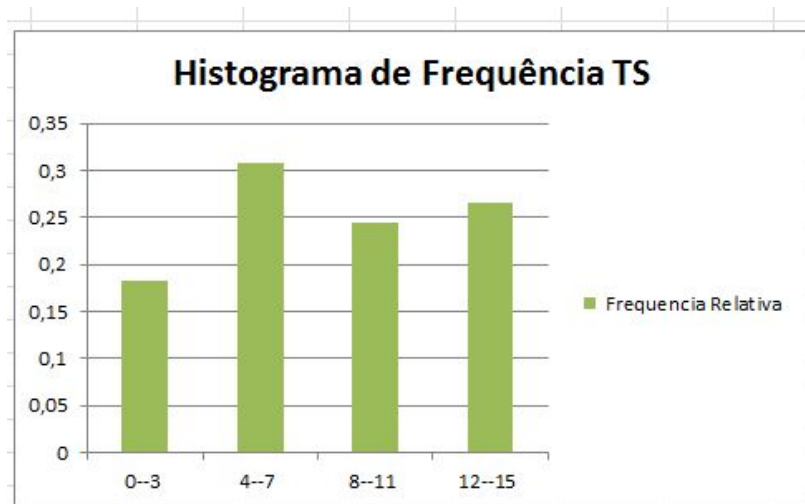


Figure 4: Histograma TS

Nessa mesma simulação obtivemos os dados médios que foram:

- Tempo médio de chegadas (minutos): 25,77.
- Tempo médio de atendimento (minutos): 7,5.
- Tempo médio de espera (minutos): 0,89.
- Quantidade média de carros atendidos: 22,86.

E também para cada simulação feita foram armazenados os seguintes dados de cada cliente:

- Tempo desde a última chegada;
- Tempo de chegada no relógio;
- Tempo de serviço;
- Tempo de início do serviço no relógio;
- Tempo do cliente na fila;
- Tempo final do serviço no relógio;
- Tempo do cliente no sistema.

Todos esses valores ficaram armazenados em um arquivo txt que será encaminhado em anexo.

### 3.3.2 Cenário 2

Para esse cenário, como falado anteriormente, foi primeiro necessário a simulação de chegadas e serviços *a priori* que, como no cenário 1 foram organizados em tabela e histograma. Segue a Tabela e o Histograma do tempo de chegada achados:

Classe ID	Frequencia	Frequencia Relativa	Frequencia Acumulada	Intervalo de Frequencia	Ponto.Medio
0--9	1828	0,1828	0,1828	[00:0,1828]	5
10--19	1910	0,191	0,3738	[0,1828:0,3738]	15
20--29	2051	0,2051	0,5789	[0,3738:0,5789]	25
30--39	2041	0,2041	0,783	[0,5789:0,783]	35
40--49	1961	0,1961	1	[0,783:1,00]	45

Figure 5: Tabela de Frequencia TEC de Entrada

Do mesmo modo foram retirados a tabela e o Histograma do tempo de Serviço

Em ambas tabelas o campo "Intervalo" serve para que quando o GNA gere um numero, o mesmo indicará em que posição está esse numero(de 0 a 1) e com que classe ele equivale na tabela, pegando assim o ponto médio dessa classe e utilizando o tempo de serviço e o tempo de chegada. Com essa tabela então definida obtivemos os seguintes dados da simulação.

- Tempo médio de chegadas (minutos): 26,08.
- Tempo médio de atendimento (minutos):8,1.

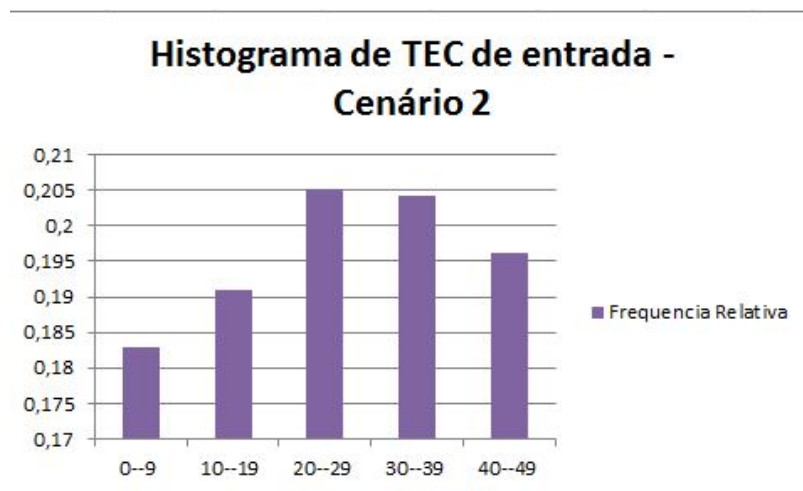


Figure 6: Histograma TEC de Entrada

Classe ID	Frequencia	Frequencia Relativa	Frequencia Acumulada	Intervalo de Valores	Ponto.Medio
0--3	1983	0,1983	0,1983	[00:0,1983]	2
4--7	2637	0,2637	0,462	[0,1983:0,462]	6
8--11	2684	0,2684	0,7304	[0,462:0,7304]	10
12--15	2696	0,2696	1	[0,7304:1,00]	14

Figure 7: Tabela TS de entrada

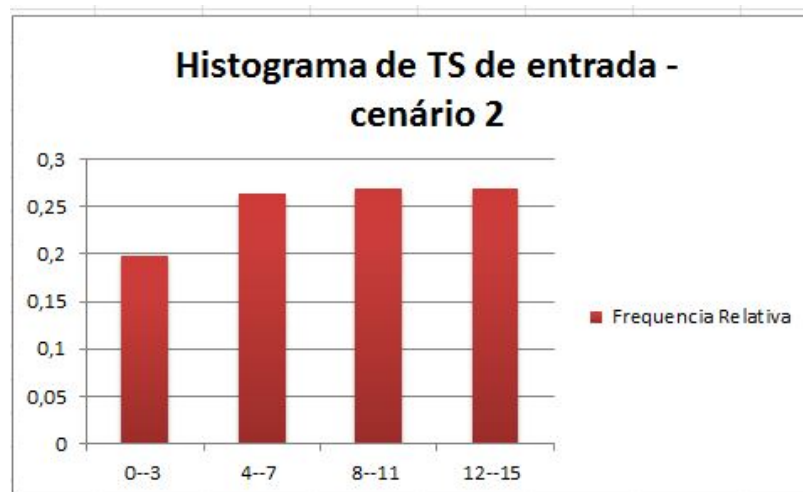


Figure 8: Histograma TS de Entrada

- Tempo médio de espera (minutos):0,94.
  - Quantidade média de carros atendidos:22,1.
- e também obtivemos as seguintes tabelas de tempo de chegada e



serviço:

Classe ID	Frequencia	Frequencia Relativa	Frequencia Acumulada	Ponto.Medio
0--3	136	0,19345662	0,19345662	2
4--7	192	0,27311522	0,46657184	6
8--11	165	0,2347084	0,70128024	10
12--15	210	0,29871976	1	14

Figure 9: Tabela TS de saída

Classe ID	Frequencia	Frequencia Relativa	Frequencia Acumulada	Ponto.Medio
0--9	126	0,17923187	0,17923187	5
10--19	119	0,16927454	0,34850641	15
20--29	150	0,21337126	0,56187767	25
30--39	157	0,22332859	0,78520626	35
40--49	153	0,21763869	1	45

Figure 10: Tabela de Frequencia TEC de Saida

E também seus respectivos Histogramas:

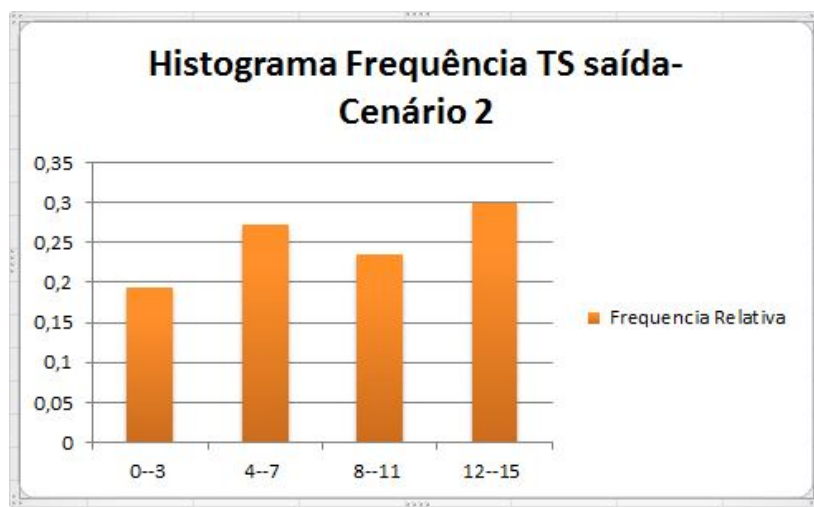


Figure 11: Histograma TS de Saida

### 3.4 Conclusão

Analisando os resultados dos Tempos de saída do cenário 2, onde foi implementado o Monte Carlo, e os resultados dos tempos do cenário 1, onde os tempos foram randômicos gerados pela própria programação(JAVA),

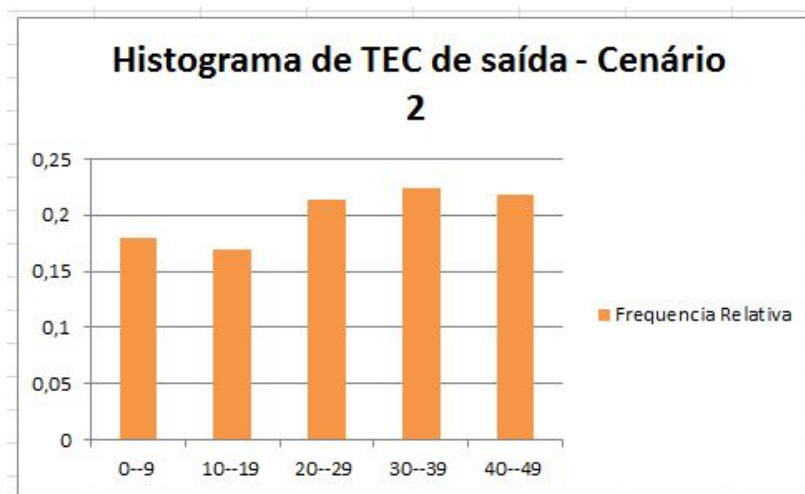


Figure 12: Histograma TEC de Saida

conseguimos notar uma grande semelhança entre os dois, isso devido a dois fatos: Primeiro, o tempo gerado radômicamente pelo java é são tempos lineares, ou seja, procura ser o mais "distribuido" possível. Oque é oque o Monte Carlo propõe também. Segundo, segundo pesquisas feitas pela gente, o randômico da Classe Random do java utiliza extamente o Monte Carlo para fazer a sorteio, ou seja, usa o GNA implementado no segundo cenário.