

• U



C •

FCTUC

FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Fábrica de Móveis

Projecto SCC - 2017

João Afonso Póvoa Marques 2015227041
Leonardo Machado Alves Vieira 2015236155
Tiago Miguel Vitorino Simões Gomes 2015238615

Introdução

O principal objetivo deste trabalho é, através de um estudo de simulação, obter dados sobre o funcionamento de um setor fabril. Desta forma, será possível avaliar o desempenho do mesmo e melhorar o sistema de modo a aumentar a sua capacidade produtiva até atingir o seu máximo potencial.

Para realizar este estudo desenvolvemos um simulador interativo, em *python*, para representar a situação descrita anteriormente. Deste modo, conseguimos avaliar com relativa facilidade diferentes cenários e analisar o comportamento do sistema.

Arquitetura

- Simulador

A implementação deste simulador assenta sobre os serviços pelos quais o móvel, de dois tipos possíveis, irá passar até estar completo. Estes serviços são a Perfuração, o Polimento e o Envernizamento. As peças são distinguidas em dois tipos, peças grandes (tipo A) e peças pequenas (tipo B). Como Eventos são tidos uma Chegada e três Saídas, com os respetivos destinos, ou nenhum no caso da saída final onde se dá por concluída a peça.

- GUI

A GUI (figura 1) foi construída em python com a livreria Tkinter. É aqui que é feita toda a interação com o utilizador, assim como todo o tipo de proteções de input. A GUI irá apresentar algumas mensagens de erro, caso os critérios de input não sejam satisfeitos. Isto é, caso o utilizador, por exemplo, não preencha todos os espaços, ou preencha com valores negativos (ou iguais a 0 em alguns casos). Aqui será dada a opção do limite de simulação ser o numero de clientes, ou parar a a mesma num certo instante. É também disponibilizado ao utilizador um input das seeds a usar nas várias distribuições aleatórias. Após pedir a simulação, uma nova janela aparece, mostrando os resultados em forma de tabela, como mostrado na figura 2.

SCC - Fabrica

Tempo de Simulação Clientes

Média de chegada de A e Seed (0->Default)

Média de chegada de B e Seed (0->Default)

Número de Máquinas de Perfuração A

Média, Desvio Padrão e Seed (0->Default)

Número de Máquinas de Perfuração B

Média, Desvio Padrão e Seed (0->Default)

Número de Máquinas de Polimento A

Média, Desvio Padrão e Seed (0->Default)

Número de Máquinas de Polimento B

Média, Desvio Padrão e Seed (0->Default)

Número de Máquinas de Envernizamento

Média, Desvio Padrão e Seed (0->Default)

ERRO - Por favor complete todos os campos...

Simular Exit

Figura 1 - Interface Gráfica

Resultados						
Tempo da Simulação: 50000.166001162535						
	Tempo Médio de Espera	Comp Médio de Espera	Utilização de Serviço	Nº de Clientes Atendidos	Nº de Clientes na Fila	Nº de Clientes em Serviço
Perfuração A	0.7680374143564346	0.15357625149798693	0.399911373180692	9998	0	0
Perfuração B	0.5653352066181839	0.42409045060654044	0.5653658259505864	37507	1	1
Polimento A	8.28457919152945	1.6564132642238483	0.798343942289615	9997	0	1
Polimento B	2786.293031622685	2089.9914641999208	0.9999715061359934	33265	4240	2
Envernizamento	0.09184670423793417	0.07946917853530971	0.6061623133008744	43262	0	0

Figura 2 - Resultados

Interpretação de Código

- simulador.py

O ficheiro `simulador.py` contém a classe *Simulador*, que tem a função de inicializar a simulação com os valores que recebe da *GUI* (tempo de simulação/número de clientes, médias, desvios e seeds), processar a simulação até atingir os limites de tempo ou clientes atendidos definidos e por fim devolver os resultados finais à *GUI*. As principais alterações em relação ao código original foram a adaptação a uma interface gráfica (através da generalização das variáveis), o uso de 5 serviços em vez de apenas 1 (suportando número de máquinas, média de serviço e desvio-padrão diferentes para cada um), a capacidade de definir seeds de aleatoriedade e ainda os diferentes casos de fim de simulação mencionados acima.

- lista.py

Este ficheiro teve uma única alteração em relação à funcionalidade original: em vez de serem juntados novos eventos ao fim da lista e apenas depois se proceder ao sorting da lista, os novos eventos são inseridos imediatamente na posição correcta.

- eventos.py

O ficheiro `eventos.py` contém 3 classes: *Evento*, *Chegada* e *Saida*.

Evento é a superclasse das outras duas classes. Um evento *Chegada* representa uma chegada de uma peça de um certo tipo ao **sistema**. Um evento *Saida* representa uma saída de uma peça de um **serviço**. Assim, uma peça completamente processada terá associada a si mesma uma *Chegada* e três *Saidas*.

Todas as classes foram alteradas para estarem sempre associadas a um cliente e uma fila, de modo a facilitar a lógica de simulação.

As *Chegadas* têm 2 funções: colocar a peça no primeiro serviço e de seguida programar a próxima *Chegada*, sendo ambos os processos diferentes dependendo do tipo de peça (tipos diferentes entram em serviços diferentes e têm médias de chegada diferentes).

As *Saidas* apenas removem o cliente do serviço a que estava associado e colocam-no no serviço seguinte (se houver).

- fila.py

No ficheiro *fila.py* existe apenas a classe *Fila*. Cada objecto desta classe representa um serviço no sistema. As principais alterações a esta classe foram no sentido de suportar mais do que uma máquina (ou seja, mais do que um cliente a ser servido em simultâneo) e de ter uma função que, em conjunto com funções de outros ficheiros, permite obter o próximo número aleatório com distribuição normal produzido por uma stream associada à *Fila*. O resto das funcionalidades mantiveram-se inalteradas.

- cliente.py

A classe *Cliente* é a única pertencente a este ficheiro. Foi acrescentada a esta classe uma variável que permite distinguir os dois tipos de peças suportadas (A e B).

- aleatorio.py

Este ficheiro contém funções que permitem criar sequências de números que seguem uma certa distribuição, necessitando apenas de números pseudo-aleatórios. Foi alterado no sentido de suportar distribuição normal (para além da exponencial, já existente) e de utilizar números pseudo-aleatórios gerados pelas funções no ficheiro *rand_generator.py* (em vez de gerados pelas funções da livraria *Random* do Python).

- rand_generator.py

Funções completamente inalteradas.

Experiências de Validação e Conclusões

- Validação Interna

Foram realizados testes nas mesmas condições, aumentando progressivamente o número de minutos da simulação adicionando 9600 minutos (1 mês). Realizámos os testes tendo em consideração uma precisão de duas casas decimais. Após a realização concluímos que foi aproximadamente no 8º mês que se deu a estabilização.

Os valores estabilizados foram os seguintes:

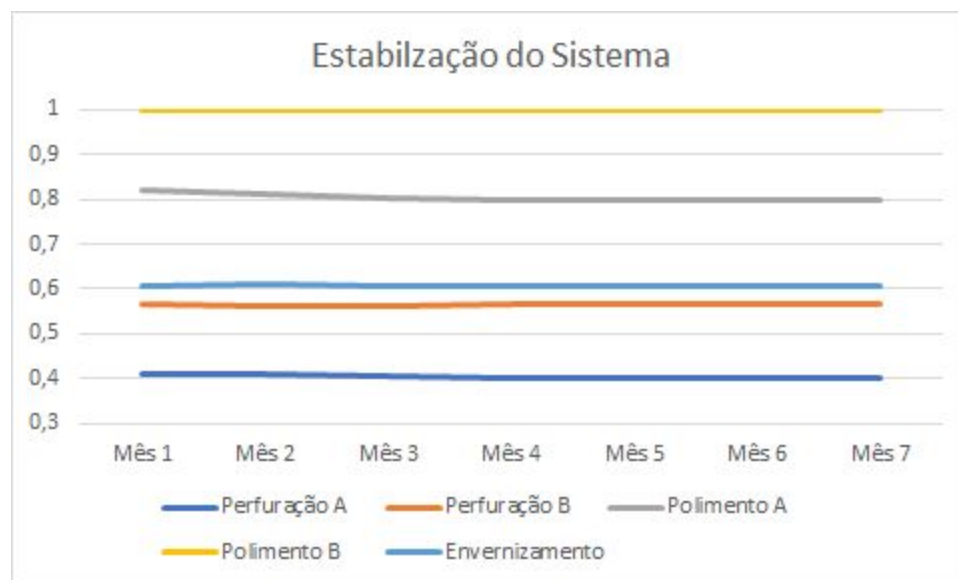
Perfuração A – 0.39%

Perfuração B – 0.56%

Polimento A – 0.79%

Polimento B – 0.99%

Envernizamento – 0.60%



Foi também efectuado um pequeno teste nos valores iniciais do sistema, em que colocámos a 0 os valores de chegada de peças A e B separadamente, para mostrar que o simulador, como esperado, não cria nenhuma peça do tipo em questão. Abaixo encontram-se os resultados da GUI a estes testes.

Resultados						
Tempo da Simulação: 50000.166001162535						
	Tempo Médio de Espera	Comp Médio de Espera	Utilização de Serviço	Nº de Clientes Atendidos	Nº de Clientes na Fila	Nº de Clientes em Serviço
Perfuração A	0	0	0.0	0	0	0
Perfuração B	0.5653352066181839	0.42409045060654044	0.5653658259505864	37507	1	1
Polimento A	0	0	0.0	0	0	0
Polimento B	2786.2930316227016	2089.9914641999335	0.9999715061359934	33265	4240	2
Envernizamento	0.011166842389247625	0.007429275576198797	0.4663405405511946	33265	0	0

Média de chegada de peças A = 0

Resultados						
Tempo da Simulação: 50000.980489865586						
	Tempo Médio de Espera	Comp Médio de Espera	Utilização de Serviço	Nº de Clientes Atendidos	Nº de Clientes na Fila	Nº de Clientes em Serviço
Perfuração A	0.7680374143564346	0.1535737498246062	0.3999048588425233	9998	0	1
Perfuração B	0	0	0.0	0	0	0
Polimento A	8.284579191529529	1.6563862821551512	0.7983472271568176	9997	0	1
Polimento B	0	0	0.0	0	0	0
Envernizamento	0.0	0.0	0.13996041552665112	9997	0	0

Média de chegada de peças A = 0

● Comparação com resultados do GPSS

De maneira a validar o nosso simulador, também simulámos o sistema no simulador GPSS, e realizámos vários testes e comparações entre o nosso simulador e o já validado GPSS. Abaixo apresenta-se o código utilizado na simulação no GPSS assim como todos os gráficos relativos à comparação entre os dois simuladores.

Código GPSS

GENERATE [tempo de simulação];
TERMINATE 1;

furaA STORAGE 1;
polA STORAGE 1;
env STORAGE 2;
furaB STORAGE 1;
polB STORAGE 2;

GENERATE (Exponential(1,0,1.33));
TRANSFER ,PerfuradoraB;
GENERATE (Exponential(1,0,5));

PerfuradoraA QUEUE filafuraA;
ENTER furaA;
DEPART filafuraA;
ADVANCE (ABS(Normal(1000,2,0.7))));
LEAVE furaA;

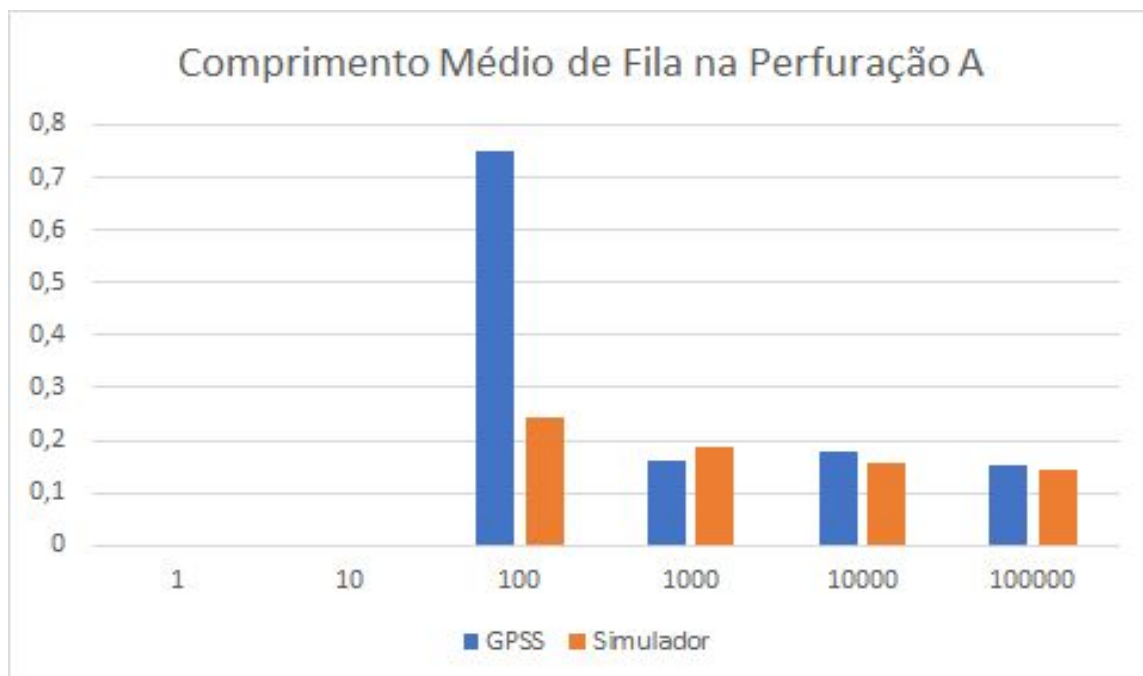
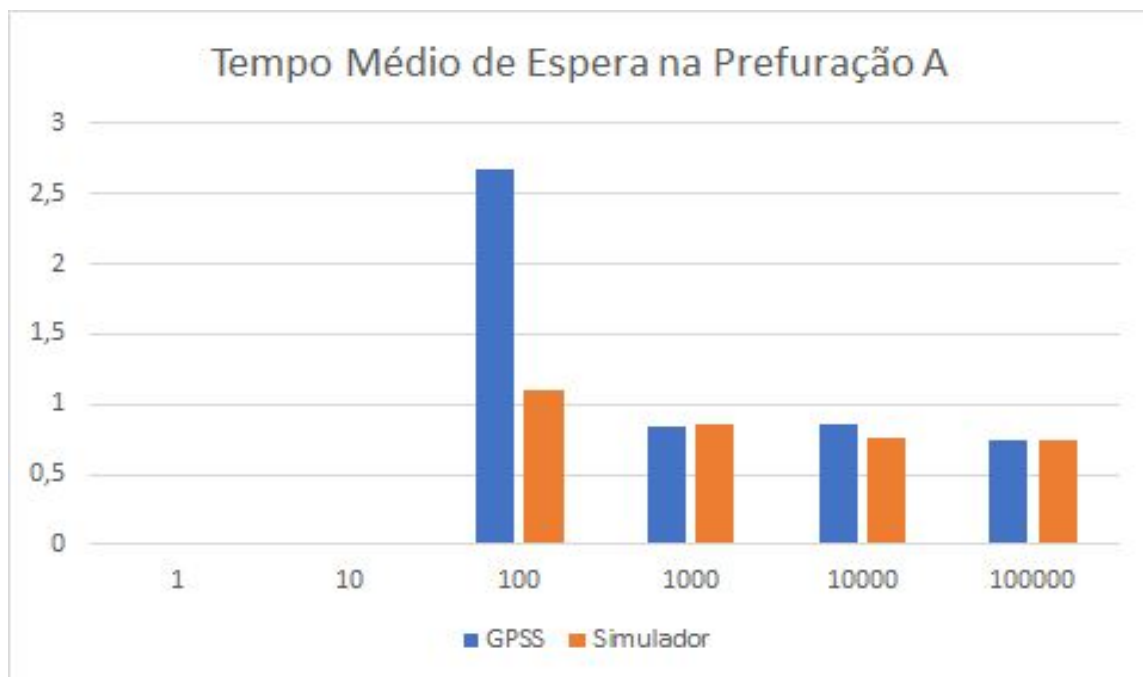
PolimentoA QUEUE filapolA;
ENTER polA;
DEPART filapolA;
ADVANCE (ABS(Normal(2000,4,1.2))));
LEAVE polA;

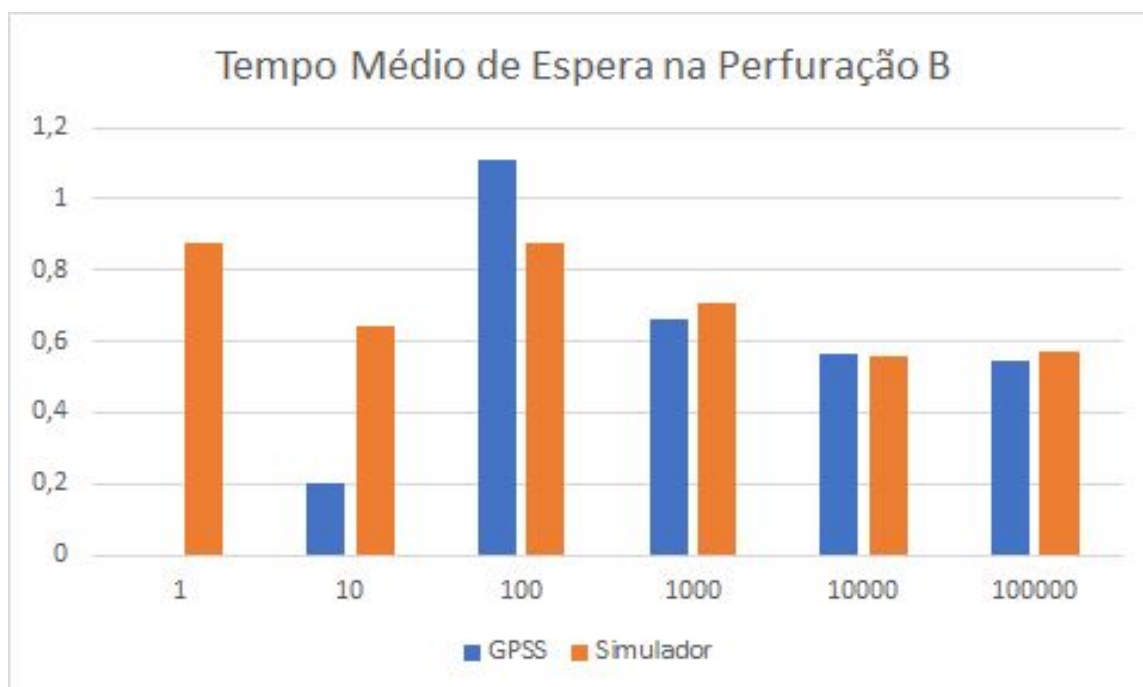
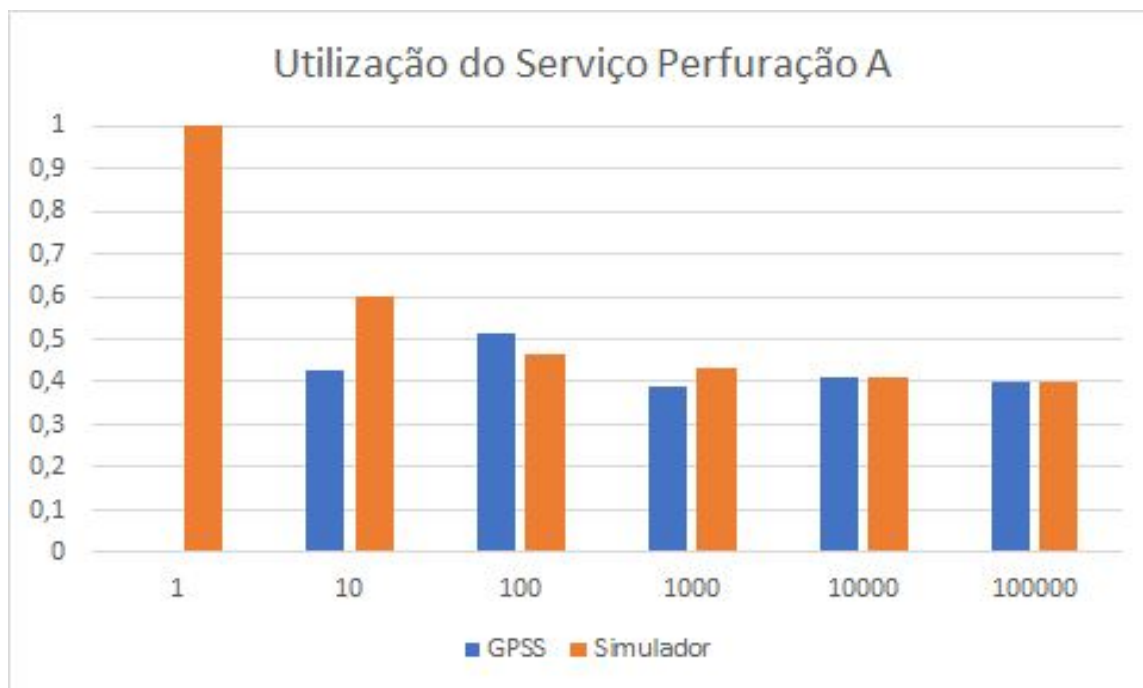
Envernizamento QUEUE filaenv;
ENTER env;
DEPART filaenv;
ADVANCE (ABS(Normal(3000,1.4,0.3))));
LEAVE env;
TERMINATE;

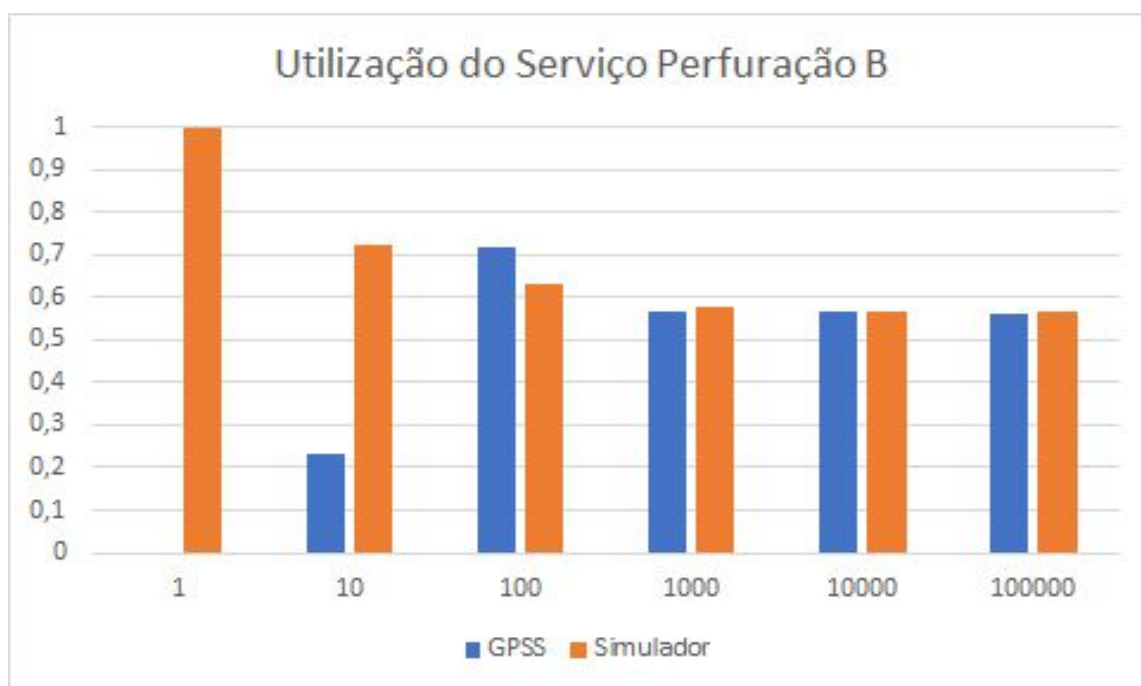
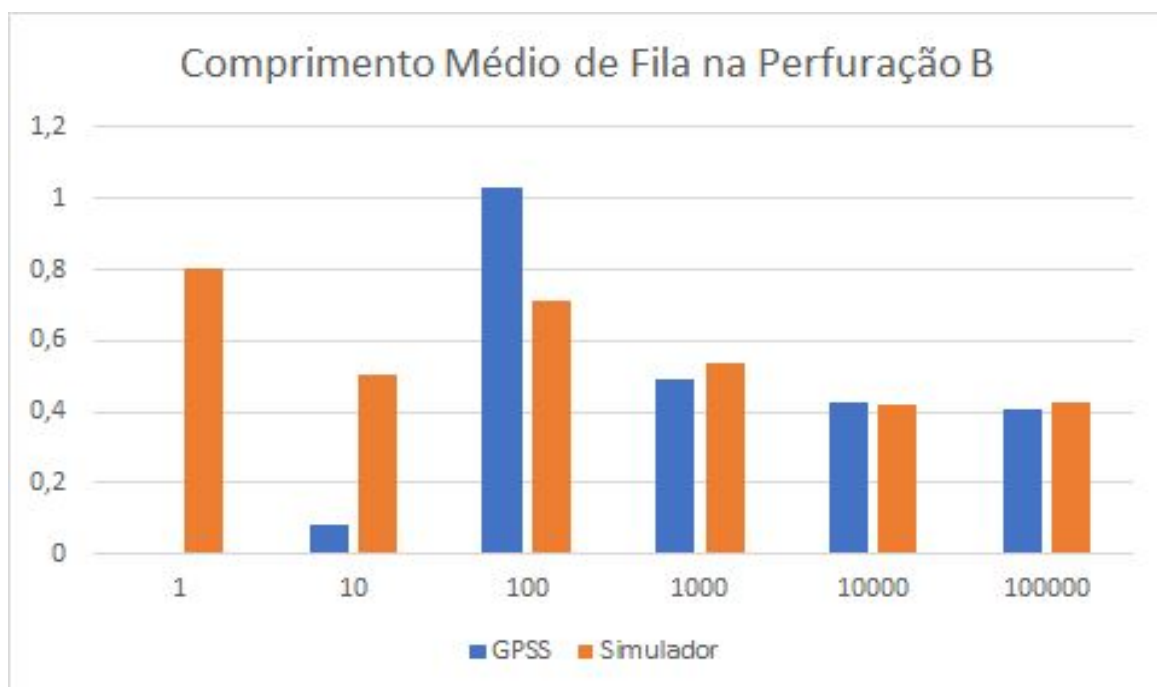
PerfuradoraB QUEUE filafuraB;
ENTER furaB;
DEPART filafuraB;
ADVANCE (ABS(Normal(4000,0.75,0.3))));
LEAVE furaB;

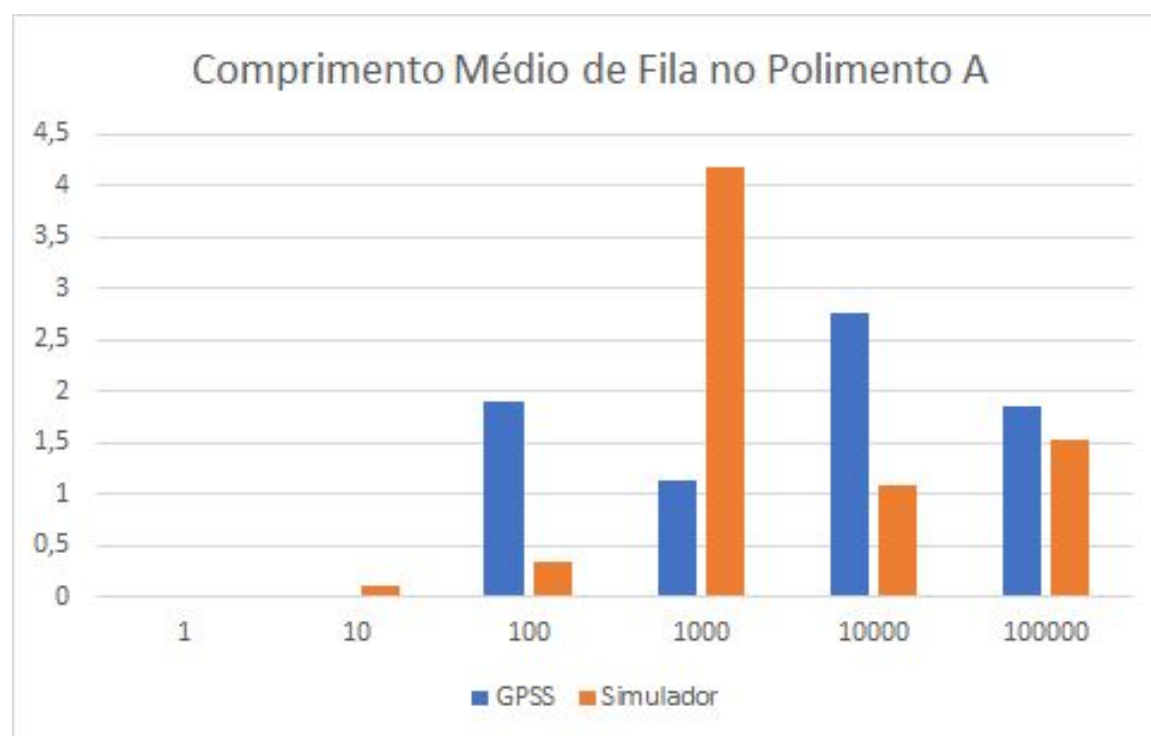
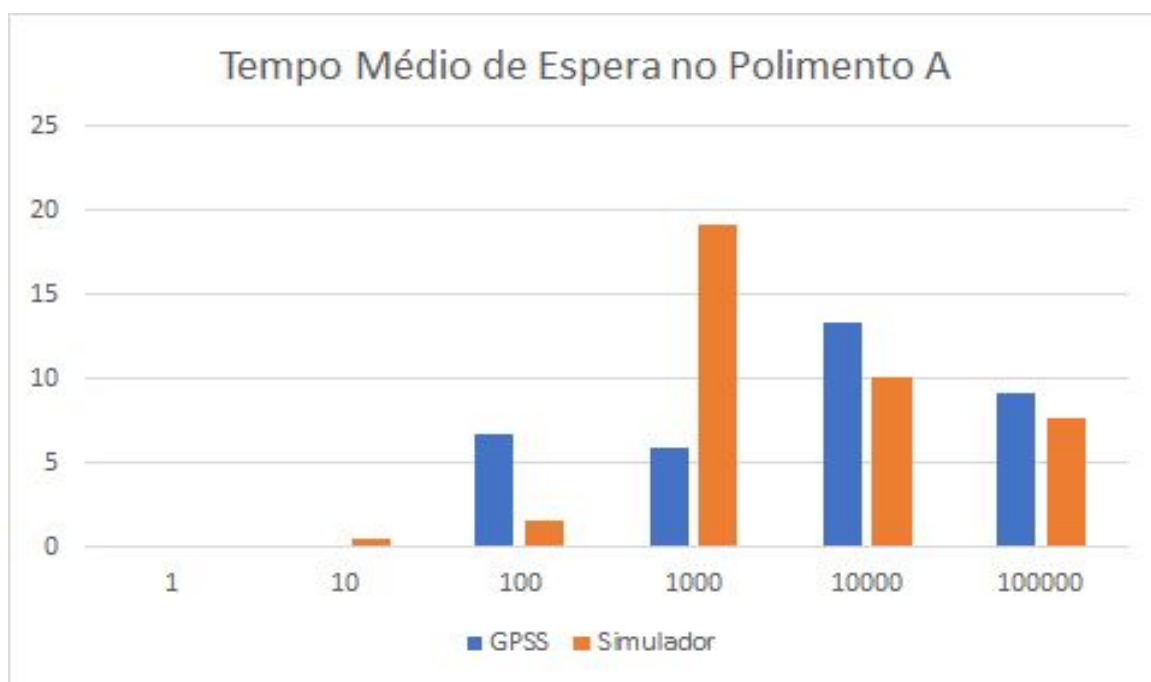
PolimentoB QUEUE filapolB;
ENTER polB;
DEPART filapolB;
ADVANCE (ABS(Normal(5000,3,1))));
LEAVE polB;

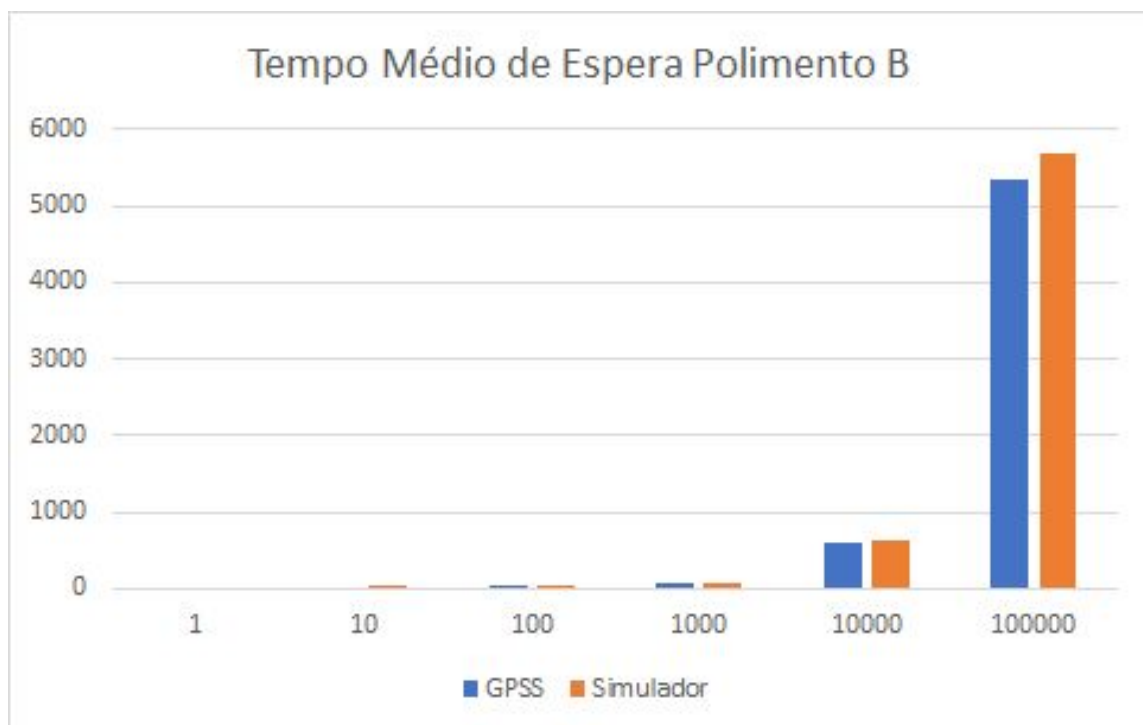
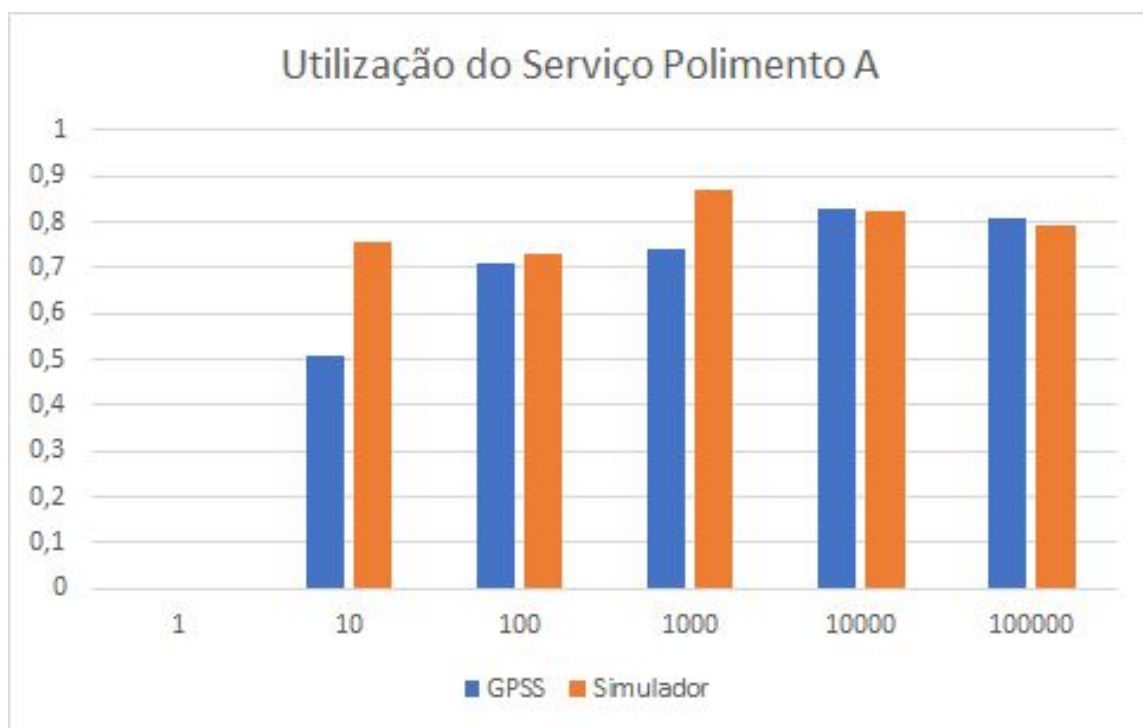
TRANSFER ,Envernizamento;

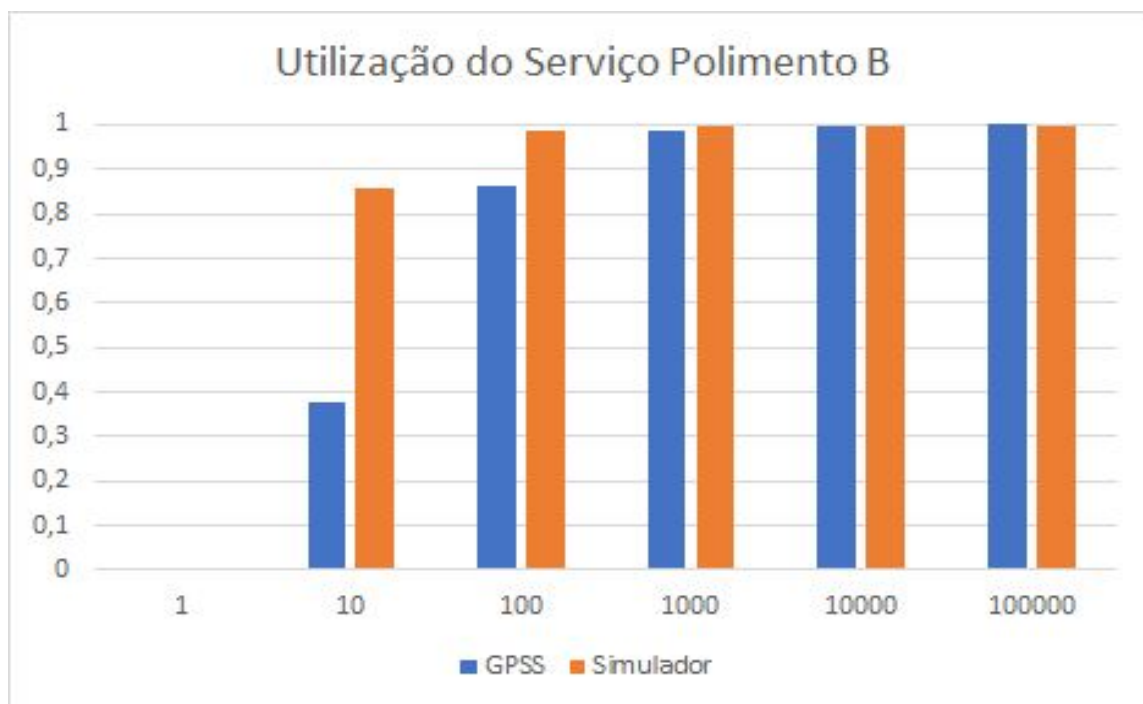
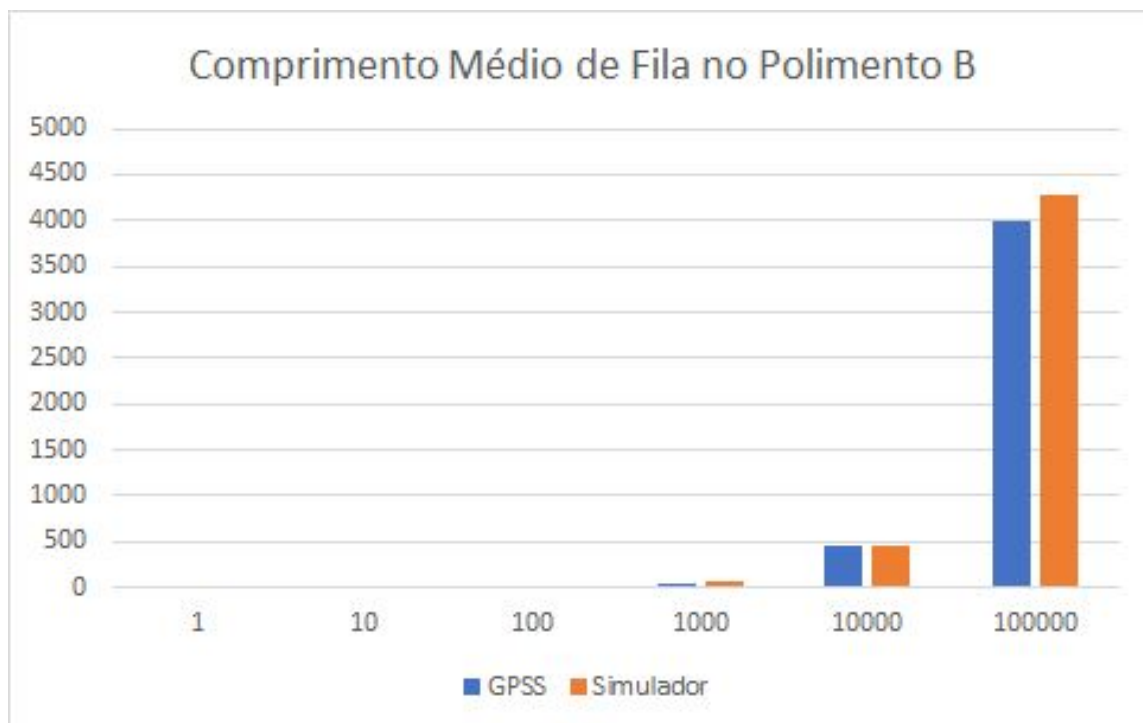


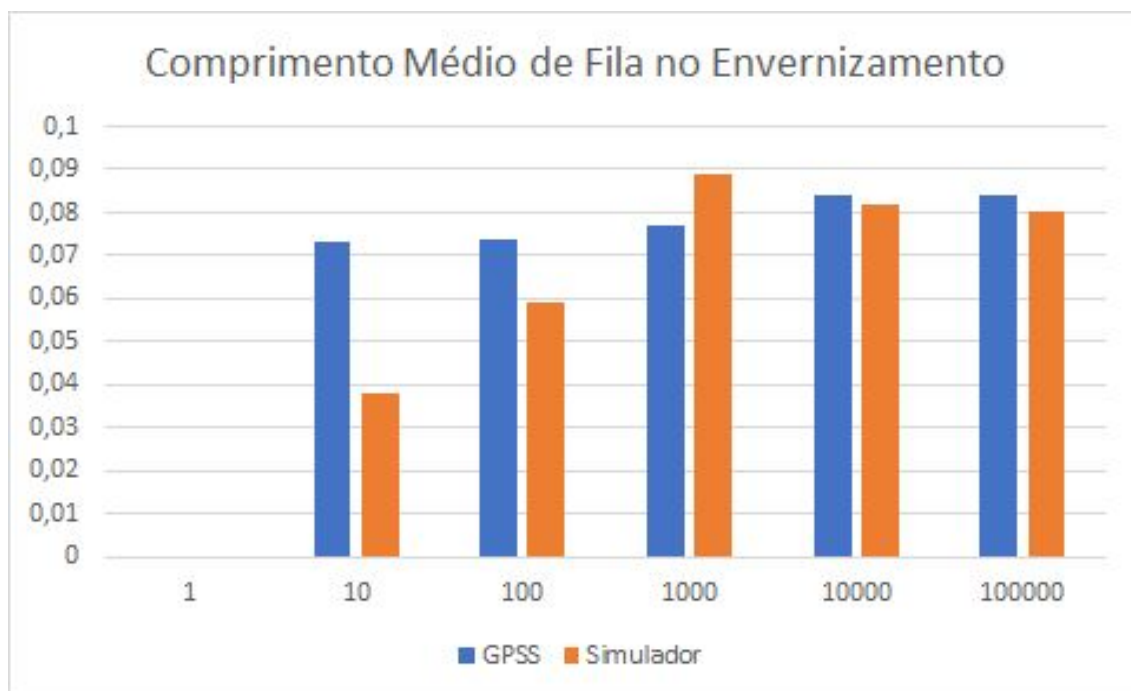
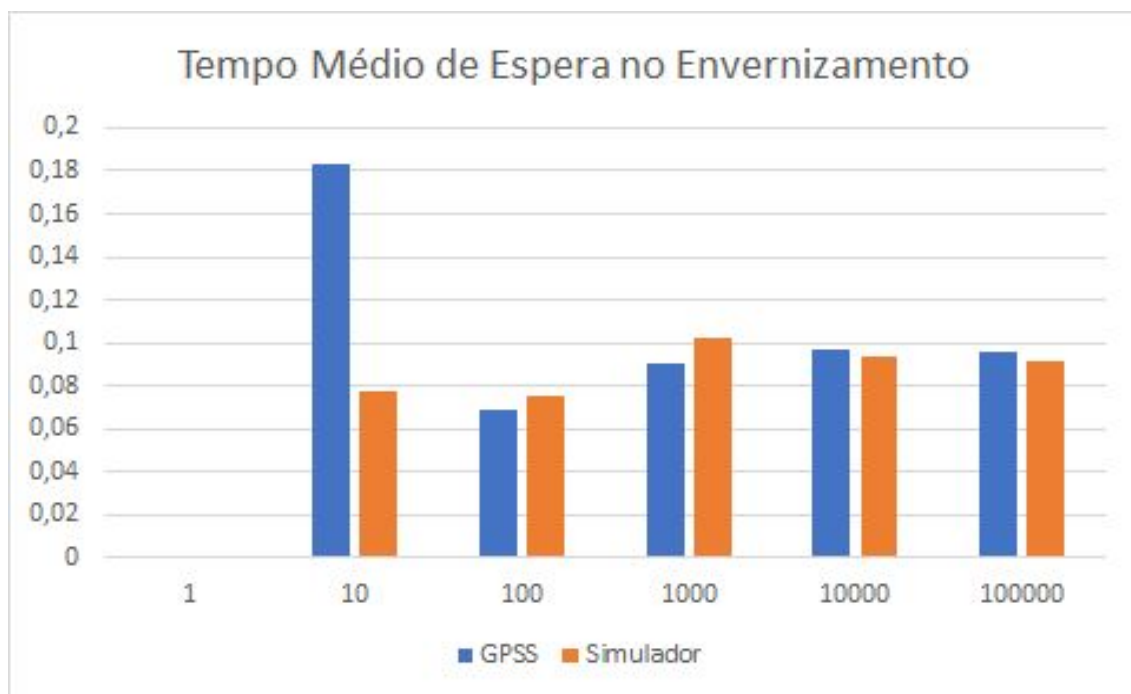


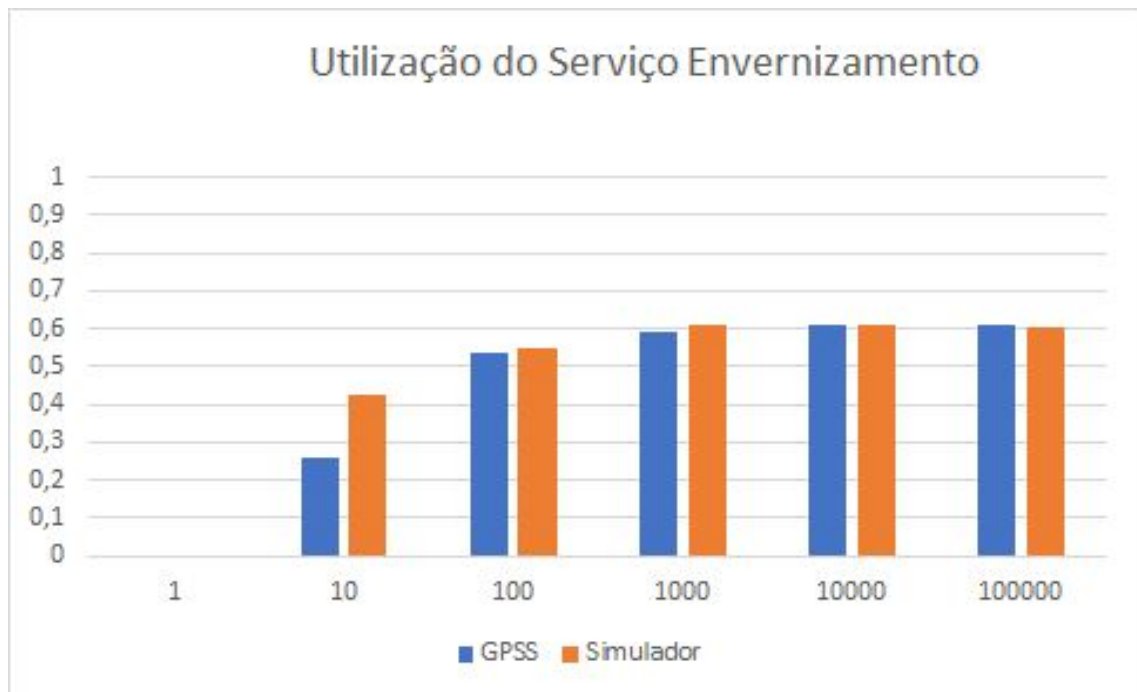












Análise de comparações

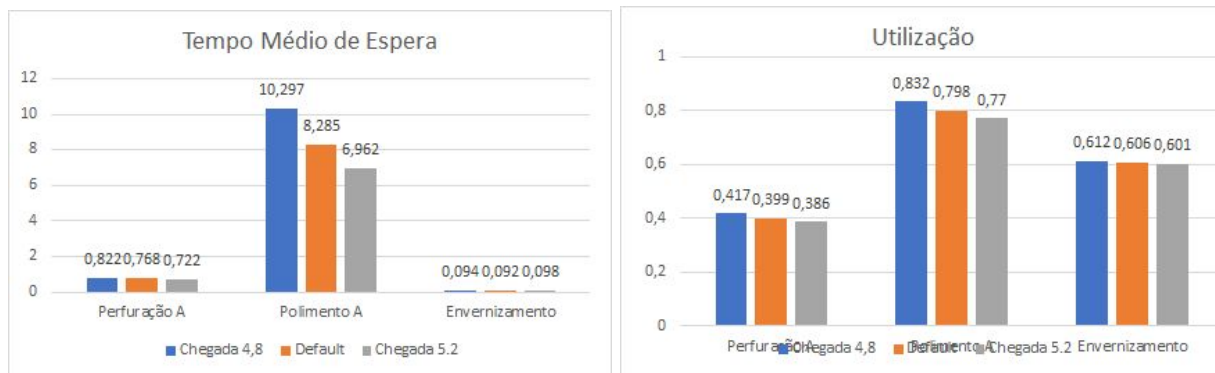
Ao analisar os gráficos de comparação, facilmente se identifica semelhanças entres os dois. Também é de realçar que os valores dos dois vão se aproximando com o avanço do tempo. Isto deve-se a uma estabilização relacionada com aleatoriedades exponenciais de normais. Onde existe bastante discrepância é nos resultados da simulação com um minuto. Isto acontece porque o simulador criado em python gera sempre um cliente de cada no instante 0, enquanto o simulador de GPSS não está implementado assim, criando valores *undefined* os quais não podemos utilizar nas comparações.

● Análise de parâmetros

Outra forma de validação utilizada foi a análise de parâmetros. Para proceder a esta análise, testámos a sensibilidade dos parâmetros de médias (quer de serviço, quer de chegada), comparando os valores “default” (do enunciado) com valores 0.2 minutos acima e abaixo.

Média chegadas A:

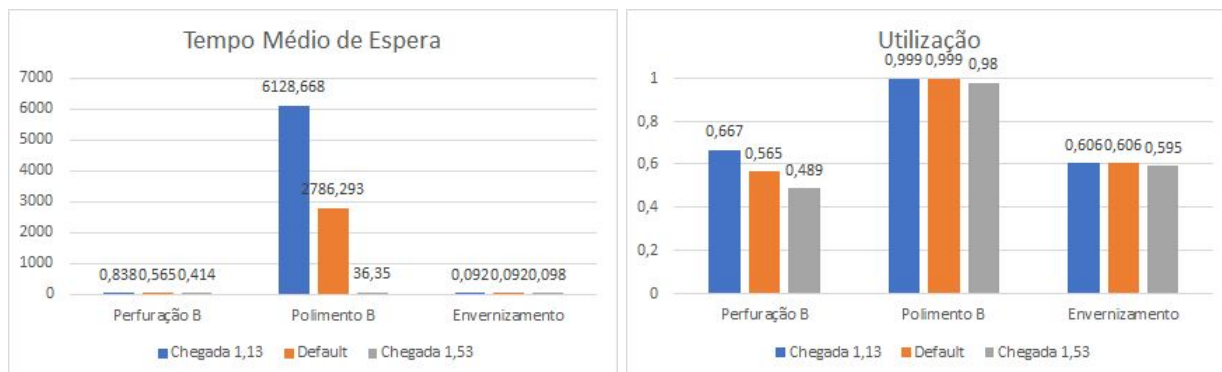
Ao diminuirmos a média de chegada das peças A, vamos ter um número muito maior de peças criadas. Assim, consequentemente, espera-se um aumento das taxas de utilização e dos tempos médios de espera nos serviços pelos quais passam peças A, sendo mais evidente nas filas onde passam exclusivamente peças A. Ao aumentarmos esta média, espera-se um efeito exactamente contrário.



Estes gráficos feitos a partir dos resultados das simulações confirmam a hipótese equacionada.

Média chegadas B:

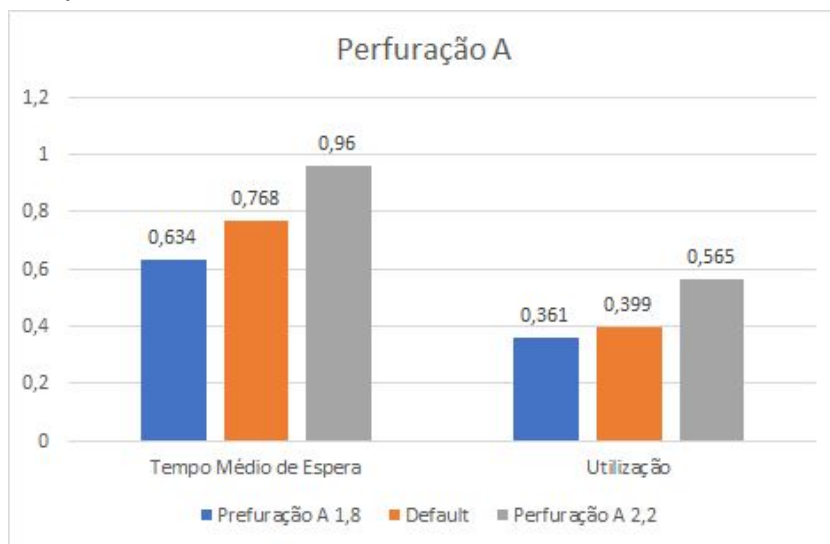
Os efeitos esperados ao ocorrerem alterações na média de chegada das peças B são os mesmos que mencionados quanto às peças A.



Mais uma vez, os gráficos confirmam o que estava estimado.

Média perfuração A:

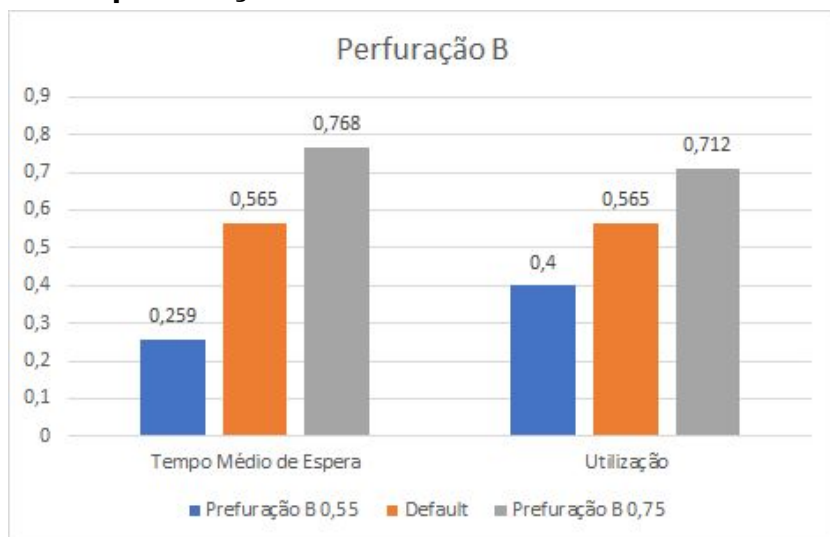
No que toca a diminuir a média deste serviço, o esperado é que haja uma significativa diminuição das taxas de utilização e dos tempos médios de espera deste mesmo serviço. Os serviços pelos quais a peça A passa posteriormente também terão ligeiras diminuições dos mesmos parâmetros, embora sejam muito menos significativos, pelo que não serão analisados. No caso de um aumento, os efeitos serão os opostos.



O gráfico mostra com bastante evidência os efeitos que se previam.

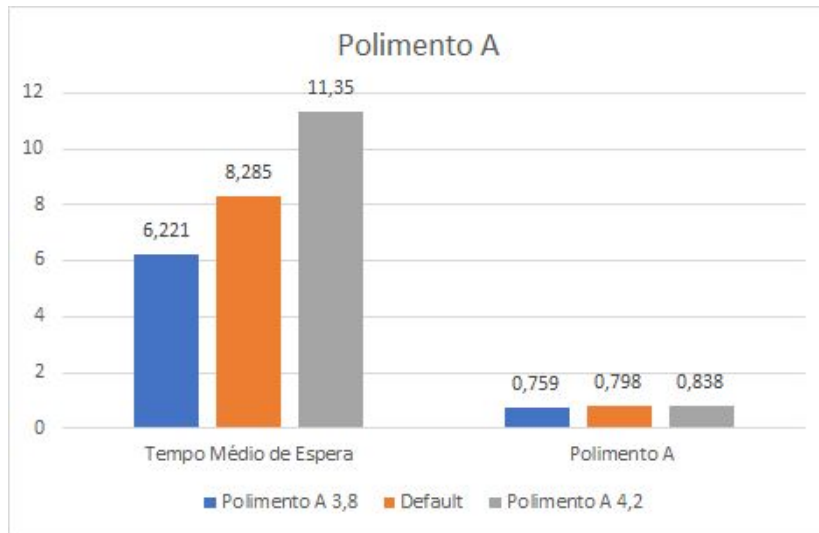
Em todas as restantes médias espera-se as mesmas consequências que as enunciadas.

Média perfuração B:



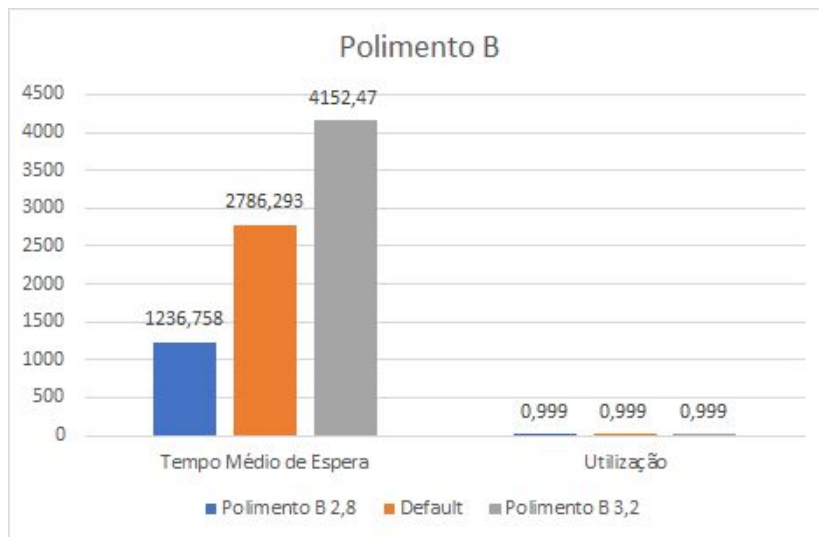
Neste caso também se comprova o que era estimado.

Média polimento A:



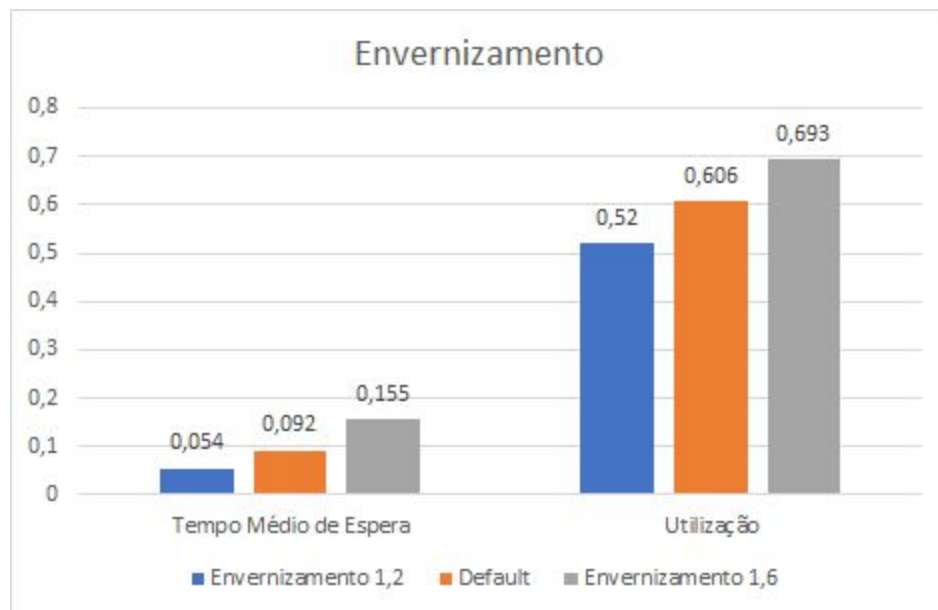
Mais uma vez, os efeitos são evidentes, pelo que a hipótese é comprovada.

Média polimento B:



Neste caso, os efeitos no que toca às taxas de utilização não são tão evidentes, mas sendo que no caso original a taxa já se aproximava dos 100%, as alterações das médias não são suficientes para se notar as consequências. No entanto o tempo médio de espera confirma o esperado.

Média Envernizamento:



No último caso, os efeitos são também os esperados.

Sistema atual e possíveis alterações

Foi nos proposto na alínea c) realizar uma alteração na zona crítica do sistema. Ao simular o cenário com as condições iniciais, facilmente se detecta que esta zona se encontra no serviço do *Polimento B*. Ao simular até à estabilização do sistema (50000 minutos), apresentam-se valores na taxa de utilização a rondar o 0.99%, e filas de milhares de peças e milhares de minutos. Valores muito superiores a qualquer outro serviço.

Assim sendo, foi nos apresentado duas soluções:

- i) Acrescentar mais uma máquina no *Polimento B* com média e desvio padrão iguais às máquinas actuais.
- ii) Melhorar as duas máquinas existentes no serviço para estas terem uma média de 1.7 minutos, mas com um custo superior à primeira solução.

Testámos os dois casos com vários valores temporais (100,9600,115200,50000).

Os resultados obtidos apresentam, para os dois casos, valores iguais (ou com uma discrepância aleatória de algumas peças), para o número de clientes atendidos no envernizamento. Isto é, o número de peças que concluíram o percurso na fábrica, foram iguais. O resto dos valores, tiveram algumas diferenças, no entanto, insignificantes para este para este estudo.

Assim sendo, para esta situação, seria preferível optar pela primeira opção visto que é financeiramente mais económica.

Em baixo encontram-se as tabelas para cada caso para 9600 minutos (1 mês) e 115200 minutos (1 ano).

Resultados						
Tempo da Simulação: 9600.202199152374						
	Tempo Médio de Espera	Comp Médio de Espera	Utilização de Serviço	Nº de Clientes Atendidos	Nº de Clientes na Fila	Nº de Clientes em Serviço
Perfuração A	0.759851586249693	0.15592459344700685	0.41233003157031467	1970	0	0
Perfuração B	0.560415278721165	0.4210031107984263	0.5653844305943526	7212	0	0
Polimento A	10.234160858737614	2.099024823938974	0.8224565096858582	1969	0	1
Polimento B	0.9965954916209205	0.7484689744577733	0.7531796459845659	7210	0	2
Envernizamento	0.2331112928245777	0.2228351330391802	0.669469934171973	9177	0	2

Resultados para o caso i) para 1 mês

Resultados						
Tempo da Simulação: 9600.920121494832						
	Tempo Médio de Espera	Comp Médio de Espera	Utilização de Serviço	Nº de Clientes Atendidos	Nº de Clientes na Fila	Nº de Clientes em Serviço
Perfuração A	0.759851586249693	0.15591293396562822	0.4122991990106866	1970	0	0
Perfuração B	0.560415278721165	0.4209716296970671	0.5653421531761817	7212	0	0
Polimento A	10.234160858737633	2.0988678663974705	0.8224697857507396	1969	0	1
Polimento B	0.646483370990674	0.4855567517718213	0.6765884211153358	7211	0	1
Envernizamento	0.2736469463373556	0.26164981434985646	0.6695816506693033	9180	0	0

Resultados para o caso ii) para 1 mês

Resultados						
Tempo da Simulação: 115200.06754518539						
	Tempo Médio de Espera	Comp Médio de Espera	Utilização de Serviço	Nº de Clientes Atendidos	Nº de Clientes na Fila	Nº de Clientes em Serviço
Perfuração A	0.7323574816873007	0.14581025478471563	0.39910236769244256	22936	0	0
Perfuração B	0.5693337178984524	0.42839866087087314	0.568132568451204	86683	0	0
Polimento A	7.790610270191211	1.5510203279763874	0.7940297578103231	22935	0	1
Polimento B	1.03944003404176	0.7821240295320769	0.7538238042444841	86682	0	1
Envernizamento	0.23375956734680262	0.22242656207357986	0.6662163384373595	109615	0	2

Resultados para o caso i) para 1 ano

Resultados						
Tempo da Simulação: 115200.06754518539						
	Tempo Médio de Espera	Comp Médio de Espera	Utilização de Serviço	Nº de Clientes Atendidos	Nº de Clientes na Fila	Nº de Clientes em Serviço
Perfuração A	0.7323574816873007	0.14581025478471563	0.39910236769244256	22936	0	0
Perfuração B	0.5693337178984524	0.42839866087087314	0.568132568451204	86683	0	0
Polimento A	7.790610270191145	1.5510203279763743	0.7940297578103231	22935	0	1
Polimento B	0.6864157531119125	0.5164918005617394	0.6758196278311898	86682	0	1
Envernizamento	0.278969923488216	0.26544747572383093	0.6662257441290598	109616	0	1

Resultados para o caso ii) para 1 ano

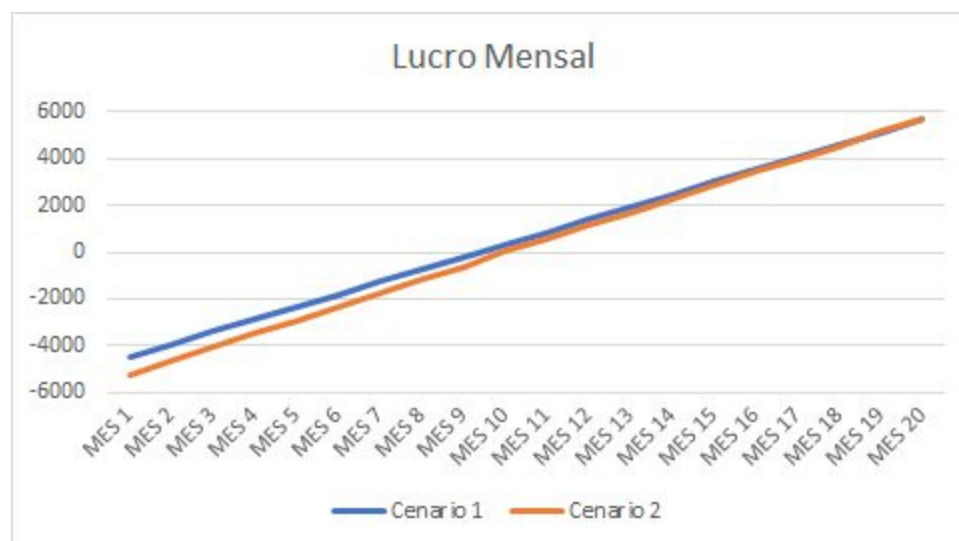
Alínea D - Análise de lucros

Nesta alínea, o objectivo era idealizar uma melhoria na produtividade da fábrica, mas mantendo a eficiência alcançada com as alterações da alínea anterior.

Para isso, para cada cenário, diminuámos o valor de chegada de peças B à fábrica, fazendo com que o número de peças que chega aumente. Testando vários valores com uma casa decimal, encontrámos um valor óptimo para cada cenário, em que o tempo médio de espera e o comprimento médio da fila do *Polimento B* é aceitável (entre 4 e 5 minutos). Os valores encontrados foram:

- i) Valor de produção de peças B – 1.1
- ii) Valor de produção de peças B – 1.0

É nos dito que o custo da implementação do primeiro cenário é de 5000€, e para o segundo cenário 5800€. É também fornecida a informação de que o lucro de cada peça é de 0.05€. Usando esta informação, calculámos o tempo necessário para o lucro obtido implementando o segundo cenário seja superior ao lucro implementando o primeiro cenário. Fizemos os testes de 9600 em 9600 minutos, ou seja de mês a mês.



	PEÇAS 1	LUCRO 1	PEÇAS 2	LUCRO 2
MÊS 1	10709	-4464.5€	11580	-5221€
MÊS 2	21284	-3935.8€	23024	-4648.8€
MÊS 3	31902	-3404.9€	34553	-4072.35€
MÊS 4	42465	-2876.75€	46035	-3498.25€
MÊS 5	53203	-2339.85€	57633	-2918.35€
MÊS 6	63895	-1805.25€	69081	-2345.95€
MÊS 7	74509	-1274.55€	80679	-1766.05€
MÊS 8	85127	-743.65€	92213	-1189.35€
MÊS 9	95812	-209.4€	103676	-616.2€
MÊS 10	106465	323.25€	115207	39.65€
MÊS 11	117120	856€	126783	539.15€
MÊS 12	127789	1389.45€	138306	1115.3€
MÊS 13	138520	1926€	149808	1690.4€
MÊS 14	149177	2458.85€	161560	2278€
MÊS 15	160007	3000.35€	172994	2849.7€
MÊS 16	170576	3528.8€	184515	3425.75€
MÊS 17	181171	4058.55€	195820	3991€
MÊS 18	191761	4588.05€	205474	4473.7€
MÊS 19	202338	5116.9€	218879	5143.95€
MÊS 20	213085	5654.25€	230351	5717.55€

Analisando os resultados obtidos, verificamos que é aproximadamente a partir do 19º mês, que o lucro obtido no segundo cenário, é superior ao obtido, na mesma altura, pelo primeiro cenário.

Divisão de trabalho

A cada elemento do grupo coube uma parte fulcral do projecto dividida no início da resolução do trabalho. Além disso foca-se a divisão do relatório, que foi realizado por todos, assim como a entreaajuda na realização do código.

Ao Tiago Gomes coube a realização da maior parte do código do simulador e tudo o que correspondia a essa mesma parte no relatório. O Leonardo Vieira fez todo o código da interface gráfica e descrição da mesma no relatório. Também desenvolveu código em tudo que se referia a aleatoriedade. Por último o João Afonso realizou o levantamento de dados e o tratamento da informação no relatório.

Conclusão

O simulador desenvolvido apresenta um comportamento realista e com bastante consistência quando comparado com um simulador já validado. Com as confirmações realizadas, da quais foram obtidos os valores esperados, podemos afirmar que o simulador se comporta de maneira realista tornando assim possível, através da alteração dos vários parâmetros, a simulação de vários cenários possíveis para o sistema em questão.

Por outro lado, convém destacar que o simulador não é perfeito. Não é possível simular a avaria de uma máquina nem a falta de material necessário ao funcionamento do sistema. No entanto todos estes imprevistos são externos ao sistema. Ainda assim, é inquestionável a relevância que uma simulação destas pode ter para uma empresa na medida em que ajuda a estudar a forma óptima de estruturar o sistema e ajuda a dissipar os riscos associados a um investimento e garantir que não se opta por uma solução mais dispendiosa e menos eficiente.