



Universidade de Coimbra
Faculdade de Ciências e Tecnologia
Departamento de Engenharia Informática

Metodologias Experimentais em Informática
Meta 2

Docentes

Carlos Manuel Mira da Fonseca
Alexandre Daniel Borges de Jesus



Equipa

João Pedro Amaral Nogueira , nº2020151675
Ricardo David da Silva Briceño, nº2020173503
Tiago de Melo Ribeiro, nº2017252377

Índice

Âmbito do Problema	2
Identificação de variáveis	2
Formulação do cenário e de hipóteses experimentais	3
Ferramentas e procedimentos	4
Resultados e Análise Experimental	5
Conclusão	8

Âmbito do Problema

Pretendemos compreender como diferentes algoritmos de escalonamento se comportam dependendo do sistema em que estão inseridos, avaliando o desempenho de cada algoritmo para diferentes variações de dados. Podendo assim escolher o melhor algoritmo de escalonamento que apresenta os melhores resultados gerais.

Contudo, a partir dos dados obtidos no trabalho realizado na meta 1, chegamos à conclusão que os cenários e hipóteses estipuladas não estariam bem formuladas, pelo que para este trabalho decidimos alterar o cenário anteriormente estipulado.

Identificação de variáveis

Tendo em conta que se pretende medir o desempenho de um sistema computacional, e sendo este uma avaliação muito abstrata, passamos a avaliar o sistema através de uma variável denominada “Total Wait Time”, resultante da soma do tempo em fila de espera do CPU e do I/O. Resumidamente, $\text{Total Wait Time} = \text{CPU Wait Time} + \text{I/O Wait Time}$, que será a quantidade de tempo que um processo fica em fila de espera aguardando pelo processamento.

Por fim, o tipo de variável nível no nosso estudo experimental tomará os valores correspondentes aos algoritmos de escalonamento, First Come First Served, Round-Robin, Shortest Job First, Shortest Remaining Time First, ou seja, “FCFS”, “RR”, “SJF”, “SRTF”, respectivamente. Ainda relativamente ao algoritmo Round-Robin, este utiliza uma variável denominada Quantum que determina o tempo de processamento de cada burst real de CPU, contudo irá permanecer intacta em toda a experiência..

Com base nos requisitos do trabalho, de modo a fazer um estudo inferencial, decidimos então alterar, não só o cenário de estudo, mas também as variáveis que queremos avaliar. Dependendo do tipo e objetivo de estudo as variáveis desempenham um papel específico. No caso do estudo de comparações entre algoritmos (T-Test e ANOVA) a variável dependente em ambas técnicas foi o Total Wait Time em função dos grupos (algoritmos), que seria a nossa variável independente. No caso do estudo de escalabilidade, a variável dependente é o Total Wait Time e a variável independente é o número de processos.

Formulação do cenário e de hipóteses experimentais

Foi pensado um cenário melhorado em relação à primeira meta, onde idealizamos um servidor real de Jogos de Xadrez. Sendo que cada processo do servidor será um jogo de xadrez a decorrer, e o CPU Burst e o IO Burst serão as jogadas do sistema(Inteligência Artificial) e do jogador, respetivamente.

Com base neste cenário foram criados vários workloads onde o número médio de jogadas é 40 (número médio de jogadas de um jogo de xadrez, segundo a Lei de Shannon), as jogadas do jogador variam entre 5 e 10 unidades de tempo, e as jogadas do sistema variam entre 4 e 6 nas primeiras 10/12 jogadas e últimas 5, sendo que as jogadas intermédias o tempo é multiplicado por um valor entre 4 a 6, assemelhando-se a uma distribuição normal. Para se analisar o impacto do aumento de número processos no sistema foram criados 8 workloads diferentes com 10, 30, 40, 50, 60, 150, 500 e 1000 números de processos respectivamente.

Como ponto de partida para a geração de hipóteses usamos um dos BoxPlots da meta 1 para observação cuidada do comportamento dos algoritmos de escalonamento e, dessa forma, pensar em hipóteses de estudo interessantes e que abordassem diferentes técnicas de estudo inferencial.

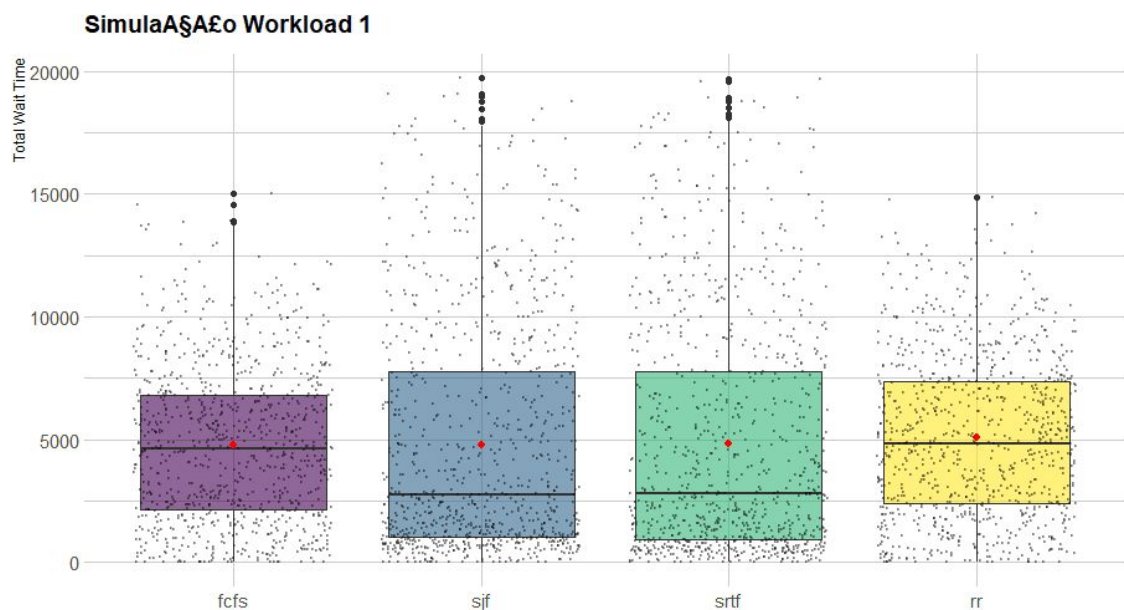


Figura 1.Dados do Workload 1

Com a discussão de grupo e observação da figura anterior, decidimos estipular três hipóteses nulas/alternativas, sendo as seguintes:

1. **H₀₁**: Não existe diferença na média do Total Wait Time do algoritmo FCFS e RR
H₁₁: Existe diferença na média do Total Wait Time do algoritmo FCFS e RR
2. **H₀₂**: Não existe diferença na média do Total Wait Time do algoritmo SRTF e SJF
H₁₂: Existe diferença na média do Total Wait Time do algoritmo SRTF e SJF
3. **H₀₃**: Não existe diferença na média Total Wait Time entre os 4 algoritmos.
H₁₃: Existe diferença na média Total Wait Time entre os 4 algoritmos.
4. **H₀₄**: Não existe relação entre o número de processos e o desempenho do sistema
H₁₄: Existe uma relação entre o número de processos e o desempenho do sistema

Ferramentas e procedimentos

Com a geração de Hipóteses nulas e alternativas, começou-se por fazer um Teste de Hipótese recorrendo ao T-Test na qual testamos os algoritmos 2 a 2 e se testou a similaridade. Para o primeiro teste comparou-se a média do Total Wait Time do algoritmo FCFS com o algoritmo RR. Para o segundo teste comparou-se a média do Total Wait Time do algoritmo SJF e SRTF.

Achou-se interessante também comparar as médias do Total Wait Time de todos os algoritmos em simultâneo recorrendo a um teste Two-way ANOVA, dado este teste ser o mais indicado para comparações com mais de 2 grupos (algoritmos). Este teste não foi feito com interação pois consideraram-se como fatores os algoritmos e as seeds usadas (30), a qual não há interação possível. Após este teste, prosseguiu-se com uma análise Post-hoc na qual se realizaram testes de normalidade com o Shapiro-Test e testes de homogeneidade com o Bartlett-Test para sabermos se podemos ou não confiar nos resultados da análise do teste da ANOVA e termos de recorrer a testes não paramétricos nos quais não é obrigatório que a normalidade e homogeneidade dos dados se cumpram.

Por último, achamos adequado estudar o comportamento dos algoritmos de escalonamento em questões de escalabilidade, visto que a variação do número de processos no workload teve um impacto direto na performance dos schedulers. Para tal, aplicou-se uma regressão linear aos dados e desta maneira pretendemos observar a relação entre a variação do número de processos e o tempo total de espera, Total Wait Time.

Foi estudada qual o tipo de relação entre estas duas variáveis usando a técnica de regressão seguindo o modelo $y = mx + b$, onde y é a nossa variável dependente (total wait time), x é a variável independente (número de processos) e m e b são os parâmetros de regressão. Para isso, foi inicialmente calculado o coeficiente de correlação, que representa o nível de dependência linear entre as variáveis, sendo um valor que varia entre 1 e -1 e onde qualquer valor acima de 0.2 ou abaixo de -0.2 sugere uma forte relação. Após se verificar a existência dessa relação, a melhor forma para se analisar que tipo de regressão existe entre variáveis é observando o gráfico das mesmas, onde se concluiu que em todos os schedulers existe um crescimento linear entre o total wait time e o número de processos. Por fim, foi feito o cálculo da regressão usando a função `lm()` do R, usando o resultado para .

Relativamente a todo o processo experimental, foram gerados 8 Workloads no total no qual os workloads com 30, ... foram usados para o estudo de comparações entre os algoritmos (T-Test e ANOVA) e os restantes workloads para o estudo do comportamento de escalabilidade (Regressão Linear)

<i>Workload</i>	<i>WI10</i>	<i>WI30</i>	<i>WI40</i>	<i>wI50</i>	<i>WI60</i>	<i>WI150</i>	<i>WI500</i>	<i>WI1000</i>
<i>Número de processos</i>	10	30	40	50	60	150	500	1000

Tabela 1. Dados da Regressão.

Resultados e Análise Experimental

Admitindo que, para todos os testes realizados, foi considerado um intervalo de confiança de 95% ($\alpha = 5\%$), começou-se por analisar os resultados obtidos dos testes de

hipótese (T-Test) a qual chegamos a conclusão que podemos rejeitar a hipótese nula numa das comparações e não na outra e, comprou-se também, que as variáveis são similares.

```
> testHypothesis("fcfs","rr5","total_wait_time","wl1")
Test Hypothesis

As variaveis são similares.
Estamos em condições de rejeitar H0.
Estamos em condições de rejeitar H0.
P-Value: 0.0212321
> source('C:/Users/ricar/Desktop/Ricardo/Mestrado/1º An
> testHypothesis("fcfs","rr5","total_wait_time","wl4")
Test Hypothesis

As variaveis são similares.
Estamos em condições de rejeitar H0.
Estamos em condições de rejeitar H0.
P-Value: 0.00361013
>
```

Figura 2.

```
> testHypothesis("sjf","srtf","total_wait_time","wl1")
Test Hypothesis

As variaveis são similares.
Não podemos rejeitar H0.
Não podemos rejeitar H0.
P-Value: 0.9693228
> testHypothesis("sjf","srtf","total_wait_time","wl4")
Test Hypothesis

As variaveis são similares.
Não podemos rejeitar H0.
Não podemos rejeitar H0.
P-Value: 0.9407785
> |
```

Figura 3.

Como se pode observar nas figuras anteriores, desenvolveu-se uma pequena função que consegue computar o T-Test passando-lhe 2 algoritmos e um workload. Podemos rejeitar os casos em que o p-value é inferior a 0.05 (5%) pelo que estamos em condições de rejeitar a hipótese nula (H_{01}) e concluir que o Total Wait Time dos algoritmos FCFS e RR5 (Quantum 5) são diferentes. Contudo, na comparação entre SJF e SRTF não podemos rejeitar a hipótese nula (H_{02}) pelo que não existe diferença na média do Total Wait Time do algoritmo SRTF e SJF.

Para observar a relação entre o Total Wait Time com os algoritmos, gerou-se um BoxPlot antes de realizarmos o teste Anova, com os parâmetros que irão ser relacionados. Podendo entender a associação entre os dois dados.

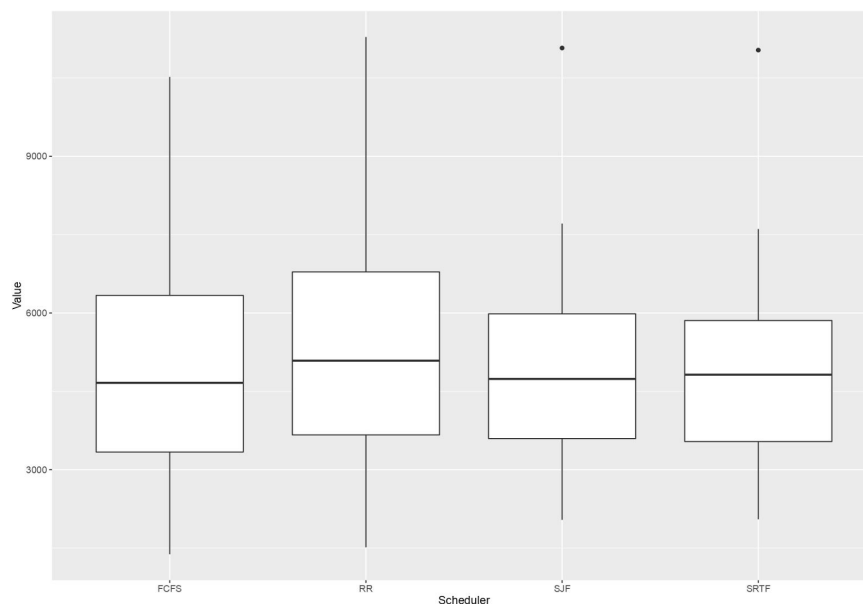


Figura 4. Relação do Total Wait Time com o Scheduler para o Workload1.

Relativamente aos resultados do Two-way ANOVA na qual se compararam os 4 algoritmos, os resultados obtidos foram os seguintes.

```
> testOneAnova("wl1","total_wait_time")
One-way ANOVA


```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
finalSeeds	29	442298540	15251674	226.04	< 2e-16 ***
finalGroups	3	2430794	810265	12.01	1.19e-06 ***
Residuals	87	5870069	67472		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> testOneAnova("wl2","total_wait_time")
One-way ANOVA


```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
finalSeeds	29	775207997	26731310	260.96	< 2e-16 ***
finalGroups	3	4433046	1477682	14.43	1.04e-07 ***
Residuals	87	8911751	102434		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Figura 5. ANOVA - WL1

```
> testOneAnova("wl3","total_wait_time")
One-way ANOVA


```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
finalSeeds	29	1.213e+09	41815809	281.47	< 2e-16 ***
finalGroups	3	9.814e+06	3271264	22.02	1.06e-10 ***
Residuals	87	1.292e+07	148561		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> testOneAnova("wl4","total_wait_time")
One-way ANOVA


```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
finalSeeds	29	1.704e+09	58753441	269.5	< 2e-16 ***
finalGroups	3	1.687e+07	5623436	25.8	4.97e-12 ***
Residuals	87	1.897e+07	218002		

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> |
```

Figura 6. ANOVA - WL3

Como se pode observar nas figuras anteriores, e considerando os Grupos (algoritmos) e as Seeds como fatores, podemos concluir que nos 4 testes podemos rejeitar a hipótese nula e afirmar com 95% de certeza que as médias do Total Wait Time dos algoritmos são diferentes e por sua vez há diferença entre os algoritmos.

Depois de realizar o teste Anova, foi necessário testar a normalidade (Shapiro) e a homogeneidade (Bartlett) para saber se podemos confiar nos resultados da ANOVA, isto é, se existe a necessidade de realizar testes não paramétricos.

```
shapiro-wilk normality test

data:  aov.out$res
W = 0.99614, p-value = 0.9867

> |
```

Figura 7. Shapiro-Test WL3

```
shapiro-wilk normality test

data:  aov.out$res
W = 0.99352, p-value = 0.8551

> |
```

Figura 8. Shapiro-Test WL4

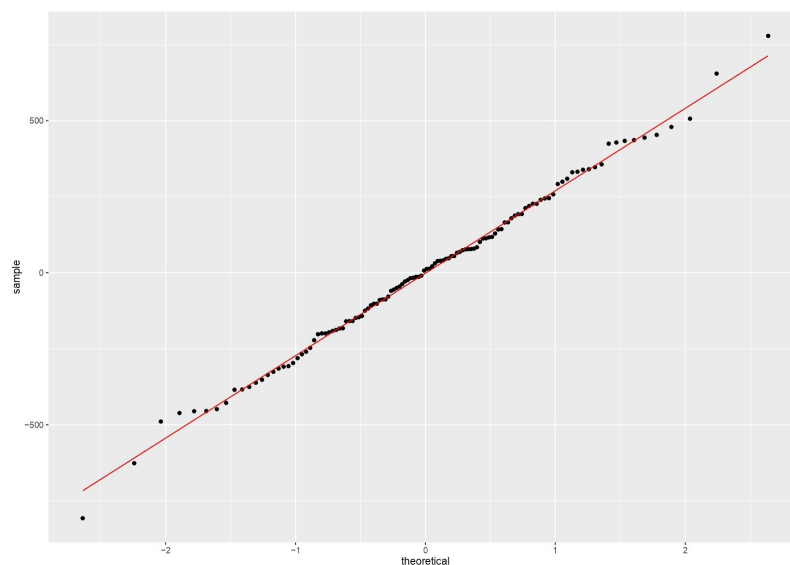


Figura 9. Teste de Normalidade WL1.

Apesar de não se representarem todas as figuras, os resultados foram semelhantes em todos os workloads, na qual é possível observar que o p-value é superior a 5% o que implica que a distribuição dos dados segue uma distribuição normal e que se pode confirmar na representação gráfica.


```

Bartlett test of homogeneity of variances
data:  finalMeans by finalGroups
Bartlett's K-squared = 0.4647, df = 3, p-value = 0.9266
>

```

Figura 10. Bartlett-Test WL1

```

Bartlett test of homogeneity of variances
data:  finalMeans by finalGroups
Bartlett's K-squared = 0.52, df = 3, p-value = 0.9145
>

```

Figura 11. Bartlett-Test WL2

```

Bartlett test of homogeneity of variances
data:  finalMeans by finalGroups
Bartlett's K-squared = 0.25403, df = 3, p-value = 0.9684
>

```

Figura 12. Bartlett-Test WL3

```

Bartlett test of homogeneity of variances
data:  finalMeans by finalGroups
Bartlett's K-squared = 0.24696, df = 3, p-value = 0.9697
>

```

Figura 13. Bartlett-Test WL4

É também possível confirmar a homogeneidade das variâncias dado que todos os testes de Bartlett retornaram um p-value superior a 5%.

Uma análise Post-Hoc foi realizada (figura abaixo) nos testes de ANOVA, descobrindo assim que pares de Schedulers são estatisticamente diferentes.

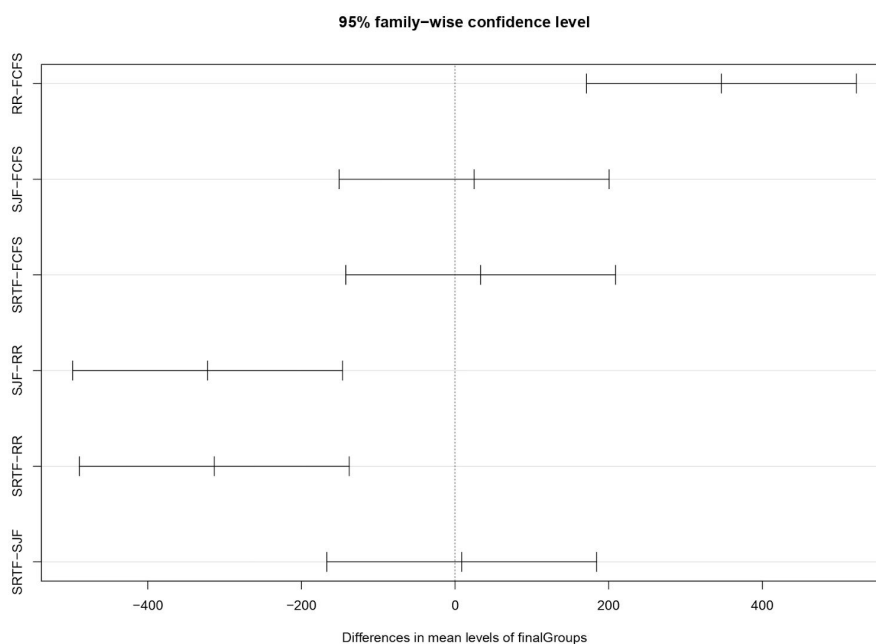


Figura 14. Tukey HDS WL1.

Como se pode observar na figura anterior, as médias do Total Wait Time entre o SRTF-SJF, SRTF-FCFS e SJF-FCFS não apresentam diferenças significativas. Contudo, entre o RR-FCFS o RR apresenta uma grande diferença, assim como nos pares SRTF-RR e SJF-RR. Sempre que é subtraída a média do RR, a diferença tem valor negativo, o que significa que o RR tem uma média de Total Wait Time acima dos outros algoritmos. No caso da diferença entre RR-FCFS, o valor é positivo com grande diferença, o que significa que o FCFS tem uma média bastante inferior em comparação ao RR.

Regressão linear entre Total Wait Time e Número de processos para cada Scheduler

First Come First Served

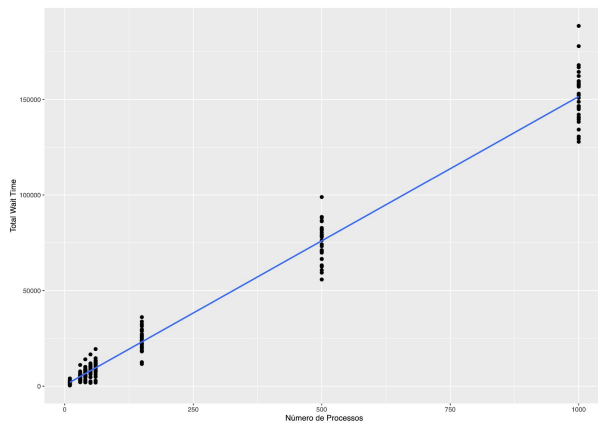


Figura 15. Regressão Linear “FCFS”

Shortest Job First

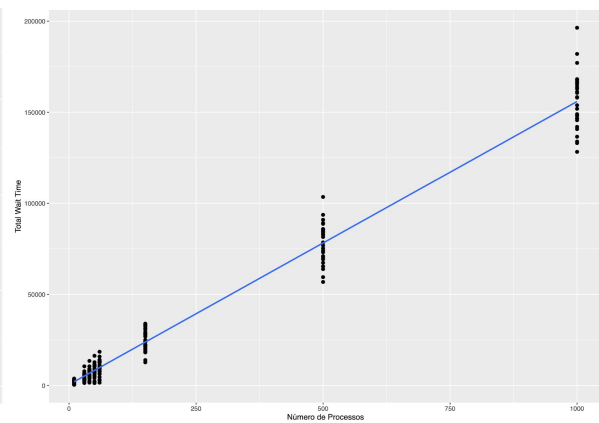


Figura 16. Regressão Linear “SJF”

Algoritmo	Coefficiente de correlação	Expressão
First Come First Served	0.9902637	$y = 524.885 + 155.393x$
Shortest Job First	0.9911326	$y = 487.255 + 151.063x$
Shortest Remaining Time First	0.9910951	$y = 502.012 + 151.088x$
Round Robin	0.9904211	$y = 561.481 + 165.661x$

Tabela 2. Dados da Regressão.

A partir dos resultados obtidos, há uma clara relação linear entre tempo total de espera e o número de processos no sistema, podendo afirmar-se que o Total Wait Time é diretamente proporcional ao número de processos.

Conclusão

Com base nos resultados obtidos e análise feita com as diferentes técnicas de análise inferencial podemos concluir que, neste cenário e estudo realizado, o algoritmo com melhor desempenho é o FCFS, tendo sido este o que apresenta uma média inferior em comparação com os outros algoritmos e que no cenário real se traduz a que os processos estejam menos tempo à espera para serem processados. O algoritmo com pior desempenho é o RR com um quantum de valor 5, apresentando resultados fracos nas condições aplicadas para estudo. Quanto aos algoritmos SJF e SRTF são estatisticamente semelhantes, tal como esperado, apresentando uma média de Total Wait Time semelhante e sem diferenças significativas.

Estes resultados confirmam os resultados da representação gráfica da figura 4, na qual é claramente visível como o Total Wait Time se comporta nos diferentes algoritmos de escalonamento. Pode-se também concluir que há uma clara relação entre o desempenho (Total Wait Time) e o número de processos, apresentando proporcionalidade direta.