Sistemas Computacionais de Apoio à Decisão

Problema de Servidores e Roubo de Bicicletas

January 14, 2017

UNIVERSIDADE DE ÉVORA

André Figueira, m37280

João Calhau, m36764

José Pimenta m37158

1 Exercício 1 - Problema de N Servidores

1.1 Resolução e Gráficos

Neste problema de N servidores, pretendem-se manter ligados um número de servidores ao longo de 24 horas (1 dia) mas devido a várias variáveis, entre as quais custos e número de cliente, deve-se fazer uma gestão de quantos servidores se devem manter ligados em cada instante da decisão.

No caso base do nosso trabalho necessitamos de utilizar a fórmula:

$$r(x_t) = -Ex_t - Nmax(0, Q_t - Cx_t) + Smin(Q_t, Cx_t)$$

sendo,

$x_t \rightarrow$ Número de servidores
$Q_t \rightarrow$ Clientes
N $\rightarrow$ Custo estimado por cliente infeliz
S $\rightarrow$ Valor médio gasto por cada cliente
C $\rightarrow$ Clientes por hora
E $\rightarrow$ Custo energético

Deste modo podemos calcular a recompensa esperada.

Para ilustrar melhor o problema, e com base nos slides dados na aula, é realizado um conjunto de gráficos que representam as recompensas (eixo dos yy) em função dos clientes Q (eixo dos xx) e servidores x_t.

É necessário agora conforme a quantidade de clientes (com base numa função de seno) verificar a quantidade de servidores que se devem manter ligados ou não em cada instante da decisão, e com base nas outras variáveis verificar também a utilidade esperada a partir de cada hora até ao final do dia.

Dada a matriz de transição da ação 0 ser sempre igual (apenas mudando o número de colunas e linhas, consoante o número de servidores), conseguimos calcular a matriz de transição da ação seguinte (com uma simples convolução entre matrizes, neste caso a matriz de ação 0 e a matriz de probabilidades inicial) e por sua vez com a nova ação calculada conseguimos calcular a matriz de transição da ação seguinte.

$$(a * v)[n] = \sum_{m=-\infty}^{\infty} a[m]v[n-m]$$

Caso base com os seguintes parâmetros:

$x_t = 3$
$Q_t = 800$
N = 0.5
S = 1
C = 200
E = 100

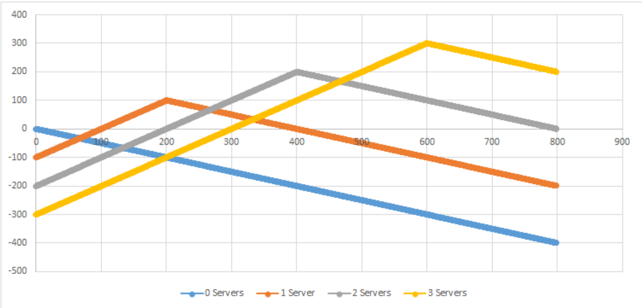


Figure 1: $r(x_t)$ em função dos clientes Q_t e servers x_t

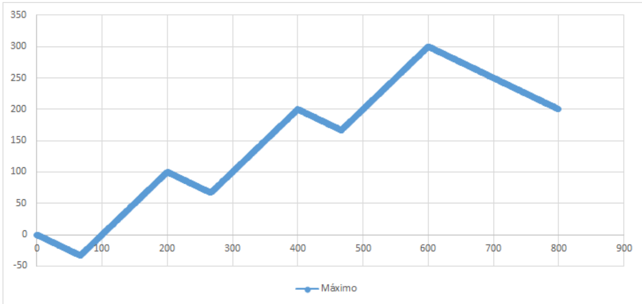


Figure 2: Máximo do $r(x_t)$ do gráfico anterior em função dos clientes Q_t

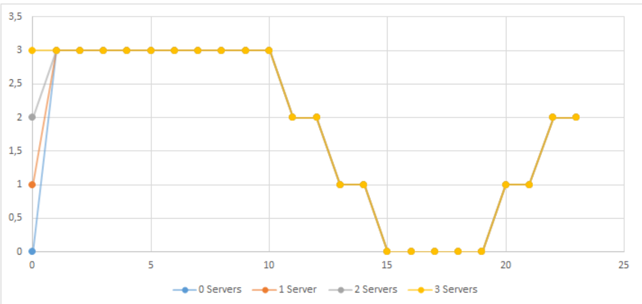


Figure 3: Path de servidores para obter maior utilidade

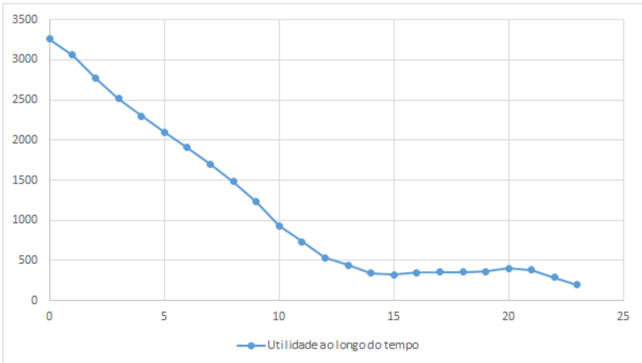


Figure 4: Utilidade esperada de cada ponto até ao fim

Como se pode verificar, na Figure 1, é um gráfico na mesma onda do gráfico realizado nas aulas, sendo a Figure 2 o máximo do gráfico da Figure 1.
Em relação à Figure 3, este demonstra quantos servidores devem estar ligados a um dado momento (hora) de modo a poder ter a melhor utilidade, sendo a Figure 4 a representação da possibilidade de ter utilidade de 1 dado momento até ao fim do dia, neste caso. Verifica-se para o caso base então uma Utilidade total estimada para 1 dia de cerca de 3250.

Utilizando agora os valores anteriores, mas modificando a quantidade de servidores de 3 para 5, com os seguintes parâmetros:

$x_t = 5$
$Q_t = 800$
N = 0.5
S = 1
C = 200
E = 100

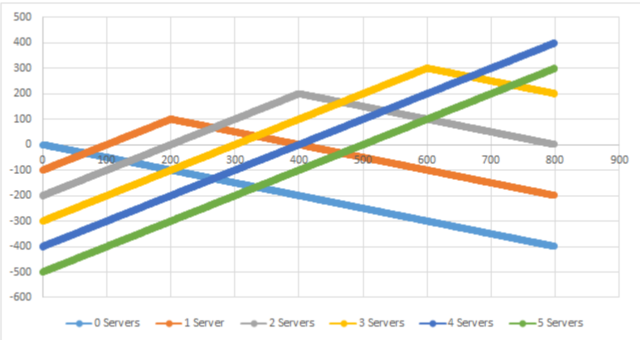


Figure 5: $r(x_t)$ em função dos clientes Q_t e servers x_t

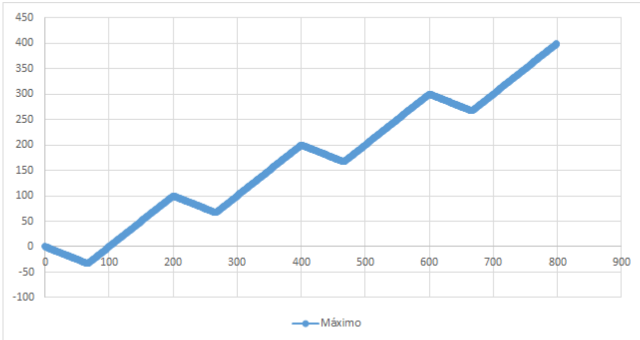


Figure 6: Máximo do $r(x_t)$ do gráfico anterior em função dos clientes Q_t

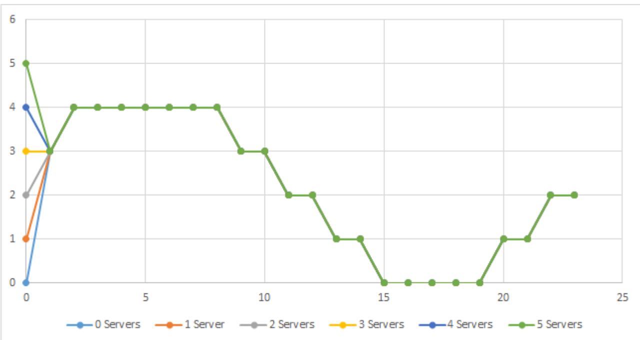


Figure 7: Path de servidores para obter maior utilidade

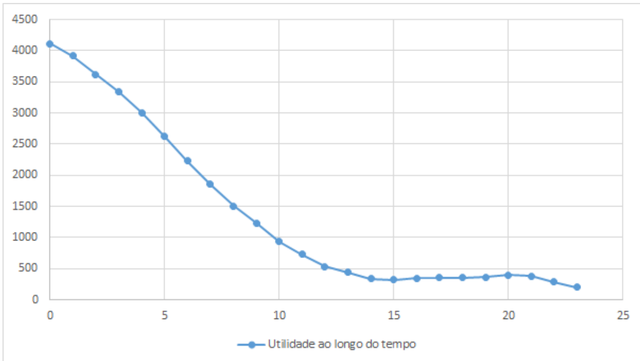


Figure 8: Utilidade esperada de cada ponto até ao fim

Neste caso, e comparando com o caso base, a Figure 5, mostra que podemos continuar a servir clientes, quando anteriormente não podíamos devido a termos apenas 3 servidores disponíveis no máximo. Isto faz com que se verifique na Figure 7 que se utilizem os 4 servidores em vez de 3 no inicio, quando verificam uma maior afluência, podendo os 4 servidores ser suficientes para 800 clientes, pois 4x200 = 800. Além do mais, a utilidade que se espera para um dia inteiro passou de 3250 para 4100 aproximadamente.

Utilizando agora os valores de base, mas modificando a quantidade de clientes de 800 para 1600, com os seguintes parâmetros:

$x_t = 3$
$Q_t = 1600$
N = 0.5
S = 1
C = 200
E = 100

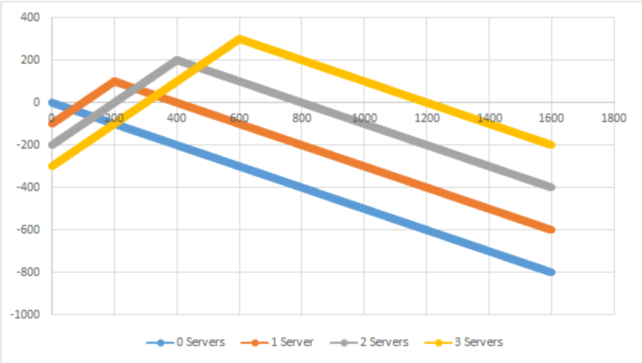


Figure 9: $r(x_t)$ em função dos clientes Q_t e servers x_t

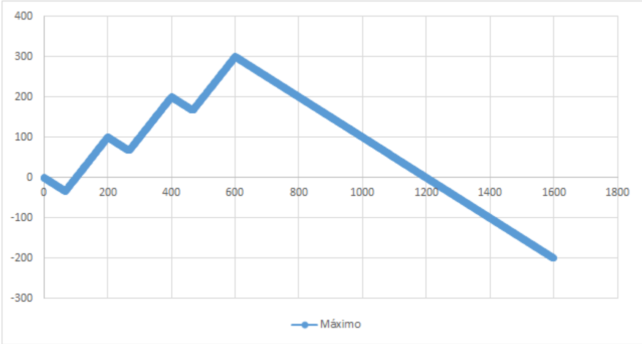


Figure 10: Máximo do $r(x_t)$ do gráfico anterior em função dos clientes Q_t

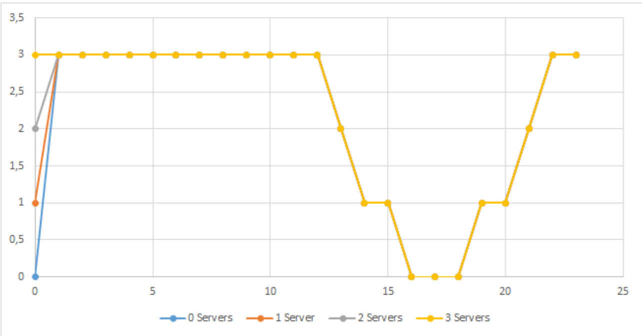


Figure 11: Path de servidores para obter maior utilidade

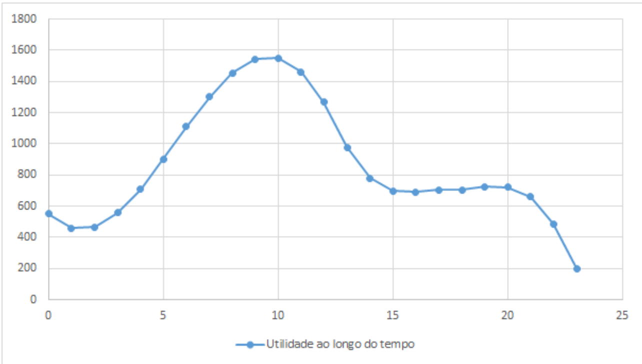


Figure 12: Utilidade esperada de cada ponto até ao fim

No caso agora apresentado, como a quantidade de clientes aumentou para o dobro, 1600, apenas 3 servidores não conseguem responder à grande procura, e assim verifica-se que se perde muito por não ter mais servidores disponíveis. Na Figure 12, sobretudo, verifica-se que quando existe mais afluência, a não disponibilidade prejudica muito a utilidade esperada a partir do início do dia, apenas sendo útil entre as 7-12 horas do dia.

Utilizando agora os valores de base, mas modificando a quantidade de N (custo por cliente infeliz) de 0.5 para 0.95, com os seguintes parâmetros:

$x_t = 3$
$Q_t = 800$
N = 0.95
S = 1
C = 200
E = 100

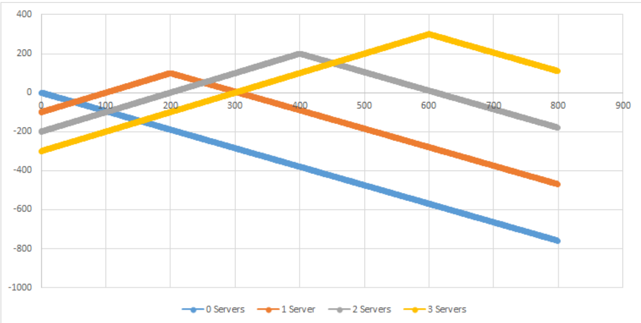


Figure 13: $r(x_t)$ em função dos clientes Q_t e servers x_t

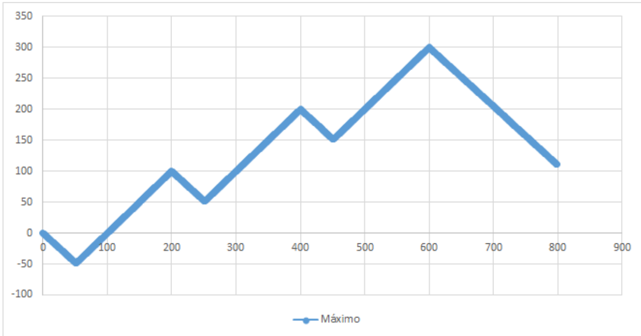


Figure 14: Máximo do $r(x_t)$ do gráfico anterior em função dos clientes Q_t

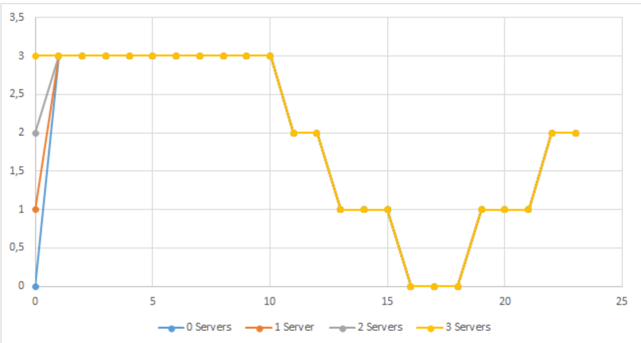


Figure 15: Path de servidores para obter maior utilidade

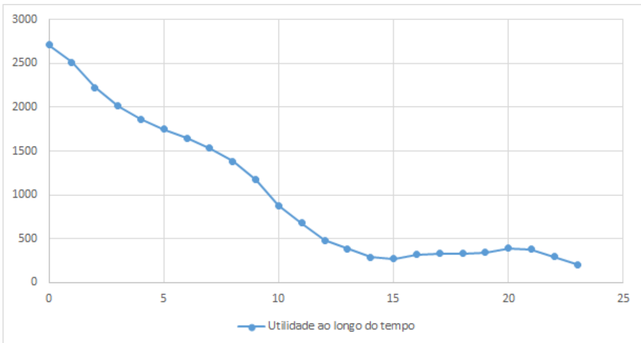


Figure 16: Utilidade esperada de cada ponto até ao fim

Nesta situação, a Figura 13 comparada com a Figura 1, consegue demonstrar o que se pretendia, que ao haver clientes que não são servidos, e por N estar com 1 valor maior, o "dono" dos servers acaba por perder mais utilidade nesses casos. Assim a utilidade para o dia todo passa de cerca de 3250 para 2750, pois nas primeiras horas, não se consegue satisfazer todos os clientes que precisam dos servidores, e assim afeta em grande parte a utilidade inicial. No restante caso, como a quantidade de servidores é mais que suficiente, não se nota uma perda significativa.

Utilizando agora os valores de base, mas modificando a quantidade de S (valor gasto por cliente) de 1 para 1.5, com os seguintes parâmetros:

$x_t = 3$
$Q_t = 800$
N = 0.5
S = 1.5
C = 200
E = 100

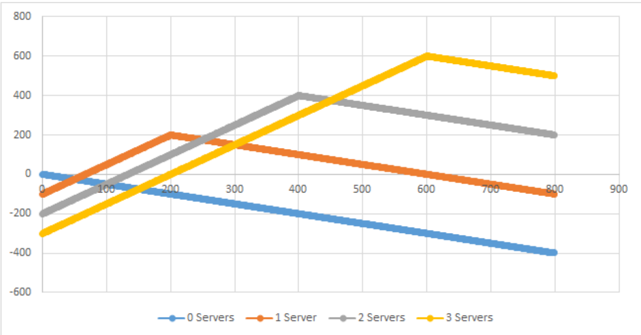


Figure 17: $r(x_t)$ em função dos clientes Q_t e servers x_t

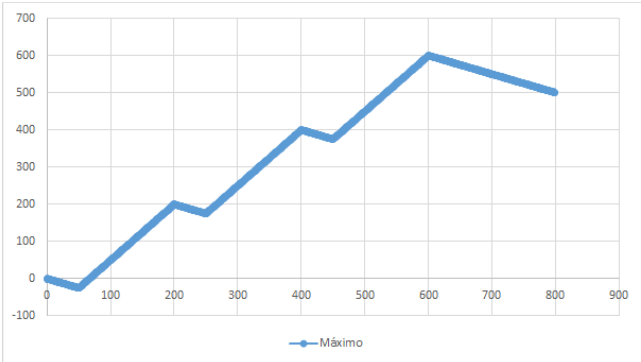


Figure 18: Máximo do $r(x_t)$ do gráfico anterior em função dos clientes Q_t

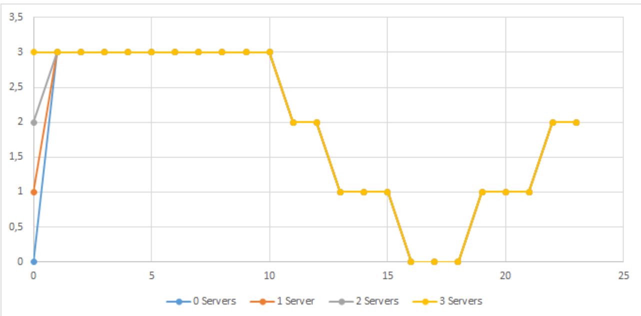


Figure 19: Path de servidores para obter maior utilidade

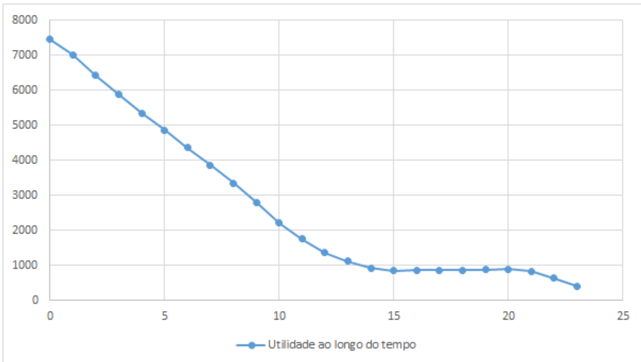


Figure 20: Utilidade esperada de cada ponto até ao fim

Neste caso, verifica-se que a utilidade conforme a quantidade de servidores, passou aproximadamente para o dobro do caso base, vendo-se que a Figura 17 é muito idêntica à Figura 1.
Em relação à Figura 20, e a utilidade esperada do inicio do dia ao fim, passou para aproximadamente 7500, o dobro do inicialmente esperado no caso base de aproximadamente 3250.

Utilizando agora os valores de base, mas modificando a quantidade de C (Clientes por hora) de 200 para 150, com os seguintes parâmetros:

$x_t = 3$
$Q_t = 800$
N = 0.5
S = 1
C = 150
E = 100

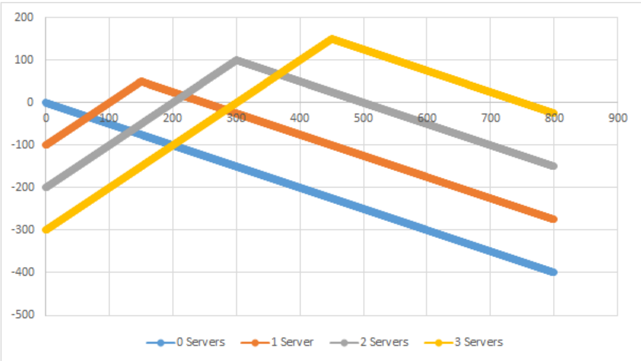


Figure 21: $r(x_t)$ em função dos clientes Q_t e servers x_t

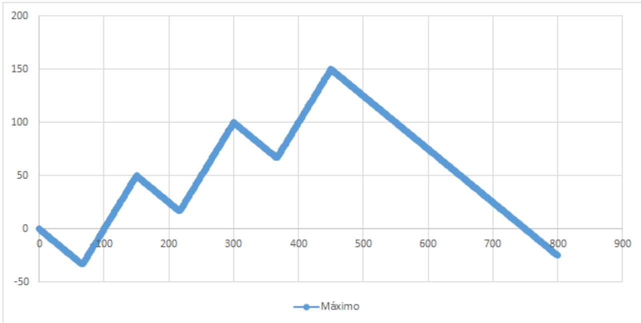


Figure 22: Máximo do $r(x_t)$ do gráfico anterior em função dos clientes Q_t

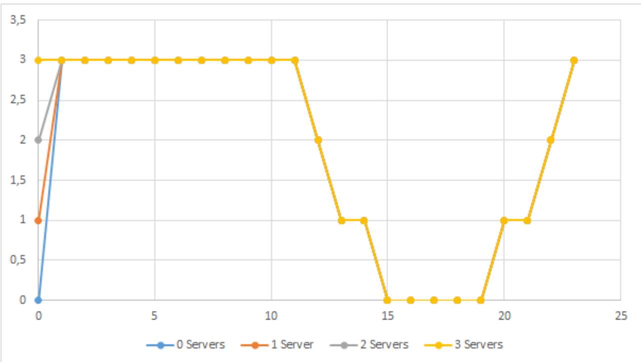


Figure 23: Path de servidores para obter maior utilidade

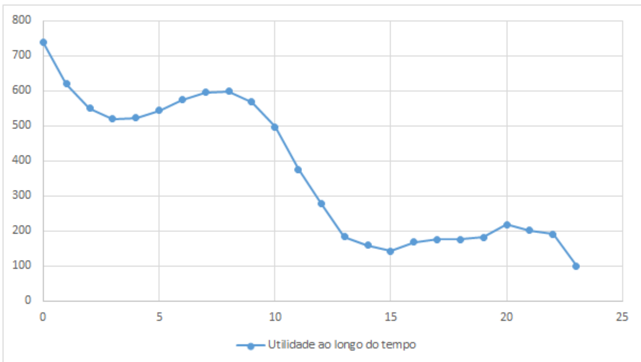


Figure 24: Utilidade esperada de cada ponto até ao fim

Nesta situação, não é possível suprimir as necessidades dos clientes, e com apenas capacidade de 150 por servidor, com apenas 3 servidores, 3x150=450, existem 800-450=350 clientes que não podem utilizar os servidores, fazendo com que a utilidade diminua, isto pois os existem mais clientes prejudicados.
Os servidores neste caso, acabam por dar muito menos utilidade, mantendo apenas os valores do final do dia estáveis como nos casos anteriores, altura de menos afluência.

Utilizando agora os valores de base, mas modificando a quantidade de E (Custo energético) de 100 para 50, com os seguintes parâmetros:

$x_t = 3$
$Q_t = 800$
N = 0.5
S = 1
C = 200
E = 50

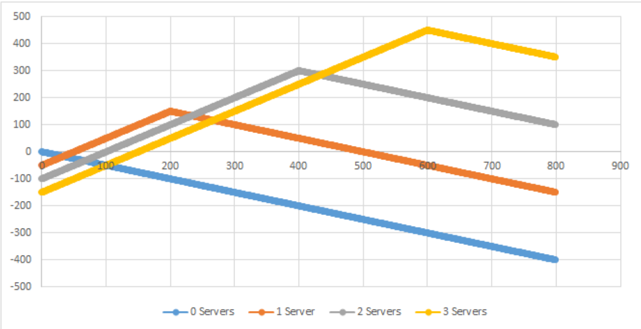


Figure 25: $r(x_t)$ em função dos clientes Q_t e servers x_t

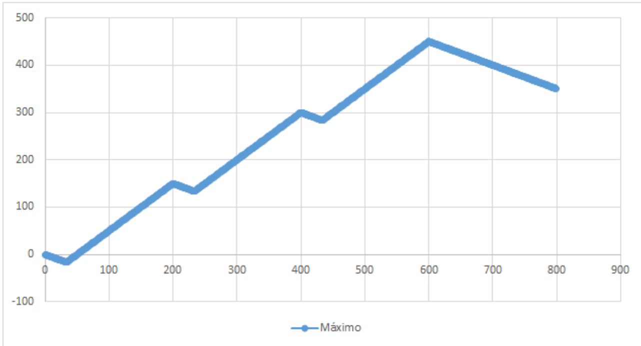


Figure 26: Máximo do $r(x_t)$ do gráfico anterior em função dos clientes Q_t

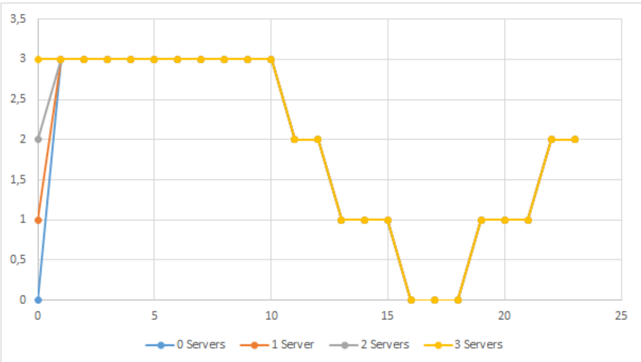


Figure 27: Path de servidores para obter maior utilidade

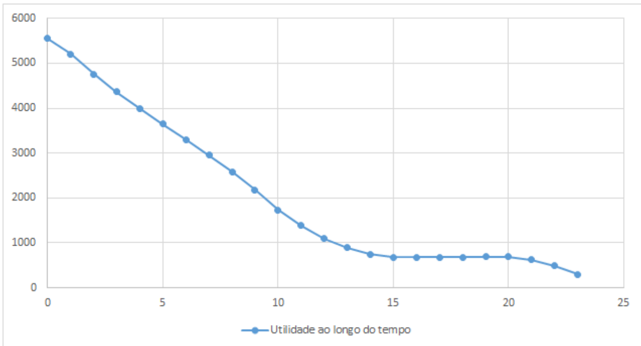


Figure 28: Utilidade esperada de cada ponto até ao fim

Neste caso, ao gastar-se menos no custo dos servers, a utilidade é muito maior, notando-se a diferença de 3250 para cerca de 5500, a contar um dia inteiro, e se contar a 2ª metade do dia, nota-se casos em que passa de cerca 400 para 800, isto porque no início do dia não se consegue satisfazer a quantidade de clientes, e assim não ter uma utilidade um pouco melhor.

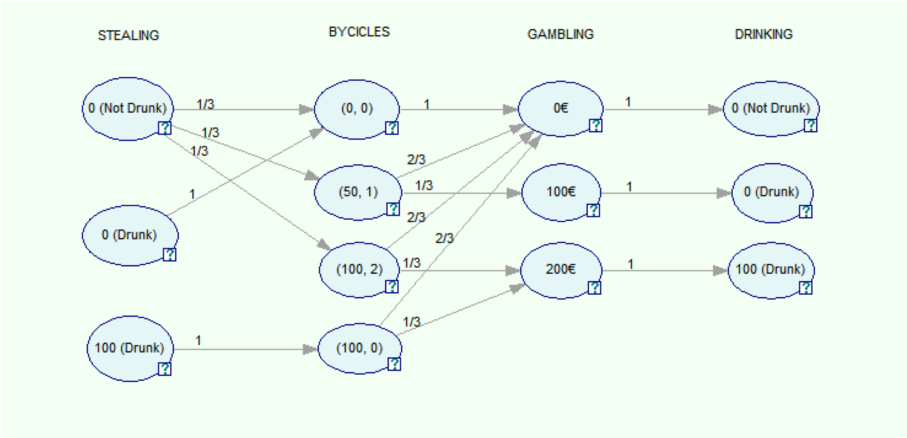
1.2 Conclusão

Após todas as referidas comparações, o que se pode concluir é que, ao longo do dia verifica-se uma situação de clientes a variar como uma função de seno, (podendo verificar-se até em casos de programas, tal como a Steam, visualizável no site a quantidade de utilizadores online a seguir aproximadamente essa função).
Sendo assim, a quantidade de servidores deve ser muito maior na 1ª metade do dia (devido à maior afluência, senão, tentar maximizar a quantidade de clientes por servidor. Ao garantir pelo menos estes dois casos, a quantidade de clientes infelizes diminui bastante, os clientes são assim mais satisfeitos.
Convém também assim dizer que quanto menor for o custo energético E, também será melhor em termos de utilidade, pois é menos dinheiro que se gasta, e o custo por cliente, quanto menor se gastar por cliente, também maior será a utilidade.
Em termos do "Path de servidores para obter maior utilidade", quando se verifica que a quantidade de clientes não excede a capacidade de determinada quantidade de servers, ou seja, com capacidade no caso base com 200 clientes por servidor, não faz sentido num dado momento com 500 clientes ter 4 ou mais servidores ativos, quando:
2x200=400, e assim 2 servidores são insuficientes, e 3x200=600, são 3 servidores e chegam para este caso. Os valores dos gráficos foram calculados com base em cálculos, que era o que se pretendia na resolução do trabalho, verificando-se no final dos cálculos conforme a escolha inicial para a quantidade de servidores ligados, quantos se deverá ter ligados no estado seguinte (com base em todos os valores descritos anteriores e variáveis). Assim pensamos ter chegado a uma boa conclusão dos dados, e conseguir assim num caso real, utilizar servidores conforme a exigência necessária.

2 Exercício 2 - Roubo de Bicicletas

2.1 Diagramas, Modelos e Tabelas

Com base nas dúvidas colocadas durante as aulas, com a ajuda do Docente, conseguimos chegar a este Diagrama, que representa o problema descrito no Enunciado.



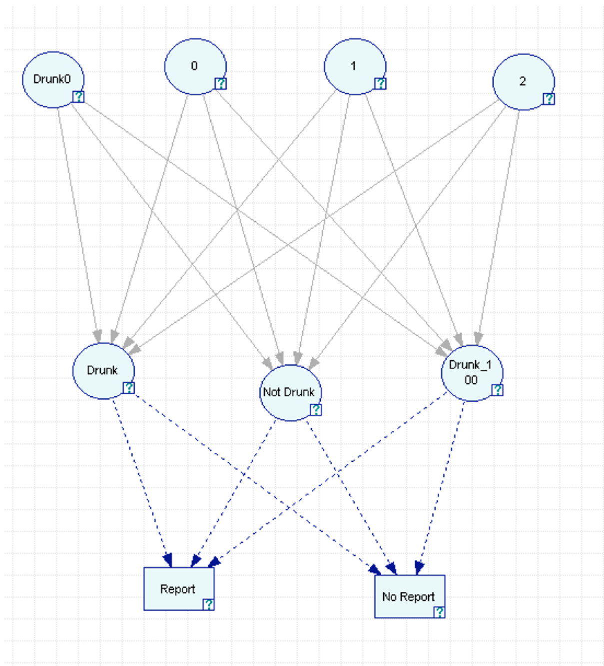
Com base neste Diagrama, podemos reparar nas seguintes observações. Através de certos caminhos podemos reparar certos comportamentos, que por exemplo, roubando entre 0 a 2 bicicletas, existem vários caminhos que chegam ao estado Not Drunk ou a Drunk, podendo sendo o estado Drunk de dois tipo, um estado Drunk de 0 euros (no caso deste ter roubado uma bicicleta e tendo ganho depois no jogo, gasto assim os 100 euros em bebidas alcoólicas), ou um estado Drunk de 100 euros, conseguindo unicamente chegar a este estado quando rouba duas bicicletas e ganha no jogo, duplicando o seu dinheiro, ou seja, fica com 200 euros, mas gasta 100 desses 200 na bebida.

No caso de chegar ao estado Not Drunk, este foi porque, ou não tinha roubado bicicletas, ou roubou uma ou duas bicicletas e perdeu o dinheiro no jogo ou ainda através de um caso especial em que tinha ficado no estado Drunk com 100 euros e tendo dinheiro mesmo bêbado e mesmo assim não roubou, mas como ainda tinha dinheiro, neste caso 100 euros, apostou no jogo. Caso este tenha sorte, e ganhe no jogo, duplicando o seu dinheiro, mas gastando na bebida volta a ficar com 100 euros, e volta a repetir até algum dia perder no jogo.

Estados	$Drunk_0$	$Drunk_{100}$	$NotDrunk$
0	0	0	1
1	$\frac{1}{3}$	0	$\frac{2}{3}$
2	0	$\frac{1}{3}$	$\frac{2}{3}$
$0'$	0	$\frac{1}{3}$	$\frac{2}{3}$

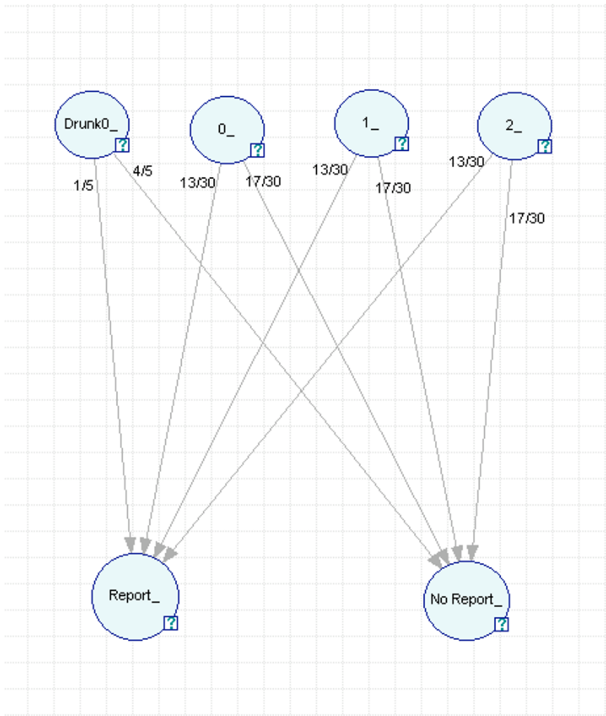
Também com o enunciado, consegue-se verificar as transições entre este indivíduo estar bêbado e ser reportado ou não o ser, ou este não esteja bêbado e ser reportado ou não.

Estados	$Report$	$NotReport$
$Drunk_0$	0.90	0.10
$Drunk_{100}$	0.90	0.10
$NotDrunk$	0.20	0.80



Com base nestas probabilidades e seguindo os caminhos do diagrama, conseguiu-se chegar à tabela de transições e diagrama seguinte:

Estados	$NotReport$	$Report$
0	$\frac{4}{5}$	$\frac{1}{5}$
1	$\frac{17}{30}$	$\frac{13}{30}$
2	$\frac{17}{30}$	$\frac{13}{30}$
$0'$	$\frac{17}{30}$	$\frac{13}{30}$



Tendo chegado às transições de observação $P(X_t- > Y_t)$, fomos seguidamente encontrar a tabela que contém as probabilidades de transição entre os estados da cadeia de Markov $P(X_{t+1}|X_t)$.
Para isso, vimos os caminhos, no primeiro diagrama de todos, dado pelo docente, de que levam, por exemplo roubar 0 bicicletas e voltar a roubar novamente 0 bicicletas, ou roubar 0 bicicletas e depois roubar uma, etc... E chegámos ao diagrama seguinte:

Estados	0	1	2	$0'$
0	$\frac{3}{9}$	$\frac{3}{9}$	$\frac{3}{9}$	0
1	$\frac{5}{9}$	$\frac{2}{9}$	$\frac{2}{9}$	0
2	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{3}{9}$
$0'$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{3}{9}$

2.2 Conclusão

Após a realização do código, e utilizando o algoritmo de viterbi (pois queremos encontrar a melhor sequência que explica as observações, escolhendo o número de bicicletas roubadas mais provável para cada estado). Dados os inputs, isto é, uma sequência de reports (yes or no report), procuramos obter a melhor explicação para tal sequência, sendo esta a seguinte, para este exemplo:

```
Reports: yes, no, yes, yes, no, yes, no, yes, no, yes, no, yes, no, no, yes

Bikes Stolen: [1, 2] ,0 ,[1, 2] ,3 ,0 ,[1, 2] ,0 ,[1, 2] ,0 ,[1, 2] ,0 ,[1, 2] ,0 ,0 ,[1, 2]
```

Com base nos resultados que verificámos, quando não há report, significa que, o caso mais provável foi que não roubou bicicletas, isto porque, não estando bêbado, a probabilidade ter um report falso é baixa, e a probabilidade de não estar bêbado dado que não roubou também é alta, isto também porque, roubando uma bicicleta ou duas, a probabilidade de não ficar bêbado, é mais baixa do que não roubar nada, e assim, no caso de existir um report, sendo que no dia anterior não houve report, o mais provável é roubar um ou duas bicicletas, isto porque, a probabilidade de ser reportado é muito alta. E porque, roubou uma bicicleta, e ganhou no jogo, ou roubou duas e ganhou no jogo.
No caso de haver um report, e no dia anterior também houve um report, o mais provável é que o individuo já tenha roubado anteriormente, e assim encontra-se no estado 0' (ou 3 como se encontra ilustrado na figura acima), isto porque, a probabilidade de estar no estado 0' e voltar a estar bêbado é de $\frac{1}{3}$, em quanto que, roubar uma ou duas bicicletas e estar bêbado é de $\frac{1}{9}$, pois tem de conseguir roubar e tem de conseguir ganhar no jogo.