UTILIZAÇÃO DE SISTEMAS INTELIGENTES BASEADOS EM APRENDIZAGEM POR REFORÇO PARA A OTIMIZAÇÃO DO PROBLEMA DO GERENCIAMENTO DE SONDAS DE PRODUÇÃO TERRESTRE

Manoel Leandro de Lima Júnior[1], Jorge Dantas de Melo[2], Adrião Duarte Dória Neto[3]

[1] Universidade Federal do Rio Grande do Norte, Campus Universitário, Lagoa Nova, Natal-RN, mleandro@dca.ufrn.br
[2] Universidade Federal do Rio Grande do Norte, Campus Universitário, Lagoa Nova, Natal-RN, jdmelo@dca.ufrn.br
[3] Universidade Federal do Rio Grande do Norte, Campus Universitário, Lagoa Nova, Natal-RN, adriao@dca.ufrn.br

Resumo – Este trabalho mostra um algoritmo original, baseado na Aprendizagem por Reforço, para a otimização do Problema do Gerenciamento de Sondas de Produção Terrestre (OtimSPT). O OtimSPT consiste em encontrar o melhor itinerário para a frota de sondas de produção terrestre disponíveis, minimizando o custo total decorrente dos deslocamentos das sondas para o atendimento das solicitações, maximizando a produção diária de uma bacia petrolífera e reduzindo as perdas de produção de óleo devido a espera por serviço. O OtimSPT será tratado como um problema de computação *online*, mais especificamente, como uma aplicação do Problema dos k Servos Homogêneos Online (PKS). O OtimSPT foi modelado segundo a Aprendizagem por Reforço, e em seguida utilizado o algoritmo *Q-Learning* como método de solução. Experimentos com uma seqüência de solicitações criada de maneira aleatória foram gerados e o seu desempenho comparado com o um método de solução do PKS consagrado pela literatura e cujas taxa de competitividade c foi provada, visando comprovar a eficiência do algoritmo sugerido.

Palavras-Chave: Produção de Petróleo, Sondas de Produção, Aprendizagem por Reforço, Problemas dos k Servos, Otimização.

Abstract – This work presents an original algorithm, based on Reinforcement Learning, as a solution for the problem of scheduling workover rigs. It consists in finding the best schedule for the available workover rigs, so as to minimize the costs inherent in the travels of the workover rigs for servicing the requests, to maximize the daily rates of oil production and to minimize the production loss associated with the wells awaiting for servicing. The problem of schedule workover rigs was dealt like a problem of online computation, more specifically, like an Online Homogeneous k Server Problem, and was modelled according to Reinforcement Learning approach, and subsequently the *Q-Learning* algorithm was used as a means of solution. A series of experiments was conducted with the aim of evaluating the appropriateness and performance of the proposed solution. The results obtained show the efficiency of the suggested algorithm in comparison with other method of solving the k-server problem cited in the literature, whose rate of competitiveness have already been proven.

Keywords: Oil production, workover rigs, Reinforcement Learning, k-Server Problem, optimization

1. Introdução

Uma bacia de produção de petróleo é composta por uma série de campos petrolíferos contendo um número de poços variável. A imensa maioria destes poços não são *surgentes*, vale dizer, não possuem pressão natural suficiente para que os fluidos atinjam a superfície. Por causa disto, a elevação dos óleos contidos nesses poços ocorre de maneira artificial, através da utilização de equipamentos que atuarão junto aos poços realizando o bombeamento contínuo dos seus fluídos. A utilização permanente destes equipamentos, que estão sujeitos à falhas, faz com que, ao longo do tempo, intervenções sejam feitas para realizar serviços de manutenção. Nesse contexto, tem-se que as sondas de produção terrestre (SPT) são unidades móveis que realizam os serviços de intervenção em equipamentos de elevação artificial de óleo.

A PETROBRAS, por razões de ordem financeira, possui uma frota limitada de sondas de produção em relação ao número de equipamentos de bombeamento dos poços. Esta limitação no número de sondas pode acarretar perdas na produção do óleo devido à demora na intervenção para corrigir as falhas nos equipamentos de bombeamento, ocasionando perdas potenciais.

O Problema do Gerenciamento de Sondas de Produção Terrestre (OtimSPT) consiste em encontrar o melhor itinerário para a frota de sondas de produção terrestre disponíveis, visando minimizar o tempo de atendimento das solicitações e os custos correspondentes às perdas decorrentes de poços inativos por falta de manutenção, maximizando, assim, a produção total da bacia petrolífera.

O objetivo deste trabalho é propor uma solução original, baseada na Aprendizagem por Reforço, para otimizar o OtimSPT. Serão consideradas como critério de deslocamento das sondas, inicialmente, somente as distâncias entre os poços, de tal forma, que não existirão prioridades no atendimento de determinados poços, independente de qual seja o fator determinante da prioridade (taxa média de produção, localização geográfica, tempo de intervenção, entre outros). Esta restrição não impede que, em trabalhos futuros, outros critérios para a seleção das SPTs possam ser considerados.

Este artigo está organizado da seguinte forma: na Seção 2 é apresentada a modelagem do OtimSPT como o Problema dos k Servos Homogêneos Online. Na Seção 3 introduz-se o conceito de análise competitiva de algoritmos *online*. Na Seção 4 apresenta-se uma breve descrição da abordagem da Aprendizagem por Reforço e do método *Q-Learning* utilizado para a solução do OtimSPT. A modelagem do OtimSPT segundo a Aprendizagem por Reforço e os resultados computacionais obtidos em experimentos são apresentados na Seção 5. A conclusão do trabalho é apresentada na Seção 6.

2. O OtimSPT Visto como o Problema dos k Servos Homogêneos Online

A solução de problemas de computação *online* tem sido objeto de estudo há bastante tempo (Borodin et al., 1998). Dentre deste contexto, o OtimSPT pode ser modelado como um problema de computação *online* da seguinte forma: dada uma seqüência de solicitações $\sigma = \{\sigma_1, \sigma_2, \cdots, \sigma_m\}$ de intervenção nos equipamentos de bombeamento dos poços, um algoritmo OtimSPT *online* deve atender cada uma dessas solicitações sem o conhecimento prévio das solicitações subseqüentes, ou seja, em um dado instante *t*, não se conhece em qual poço irá surgir a próxima intervenção (solicitação), no instante de tempo t', ou seja, $\sigma_{t'}$ é desconhecido se $t' > t$. Desde que o atendimento das solicitações implica em custos, o objetivo buscado é a minimização desses custos. Formalmente, a bacia petrolífera é caracterizada por um grafo ponderado $G = \{N, A\}$, onde *N* é o conjunto de pontos (o qual representa os poços), com $|N| = n$, e *A* é o conjunto de arestas (i, j) (cada qual representando o caminho que interliga dois poços). A cada aresta (i, j) é associada uma ponderação $d(i, j)$, representando o custo de deslocamento das sondas de produção por este caminho.

Simplificadamente, o OtimSPT pode ser visualizado como o Problema dos k Servos Homogêneos[1] Online (PKS) (Manasse et al., 1988). O modelo do PKS é bastante estudado na literatura e serve como abstração para uma ampla gama de problemas *online* (Borodin et al., 1998). O PKS modelado de acordo com o OtimSPT pode ser definido formalmente da seguinte maneira: Seja *K* um conjunto de SPTs (servos), com $|K| = k$, localizados em *k* poços, necessariamente distintos, da Bacia Petrolífera *G*, e seja $\sigma = \{\sigma_1, \sigma_2, ..., \sigma_m\}$, uma seqüência de solicitações, as quais podem surgir em um poço qualquer, de intervenção nos poços. Para atender a cada uma dessas solicitações σ_i em um dado instante t_i ($i = 1,2,...,m$), um dos SPTs deve ser movido de sua posição atual para o poço σ_i. Associado a esse deslocamento de um nó *i* para um nó *j*, existe um custo de atendimento proporcional à distância percorrida $d(i, j)$. O objetivo em questão é atender a toda a seqüência de solicitações, minimizando o custo total, $\sum_{l=1}^{m} d_l(i, j)$.

[1] O termo Homogêneo é utilizado para caracterizar que todos os servos são idênticos e que as solicitações por serviço podem ser atendidas com o deslocamento de qualquer um dos servos aos locais onde as solicitações surgiram.

3. **Análise Competitiva**

Uma solução ótima para o problema é normalmente possível se a seqüência inteira é conhecida *a priori*. Nesse caso, o algoritmo de solução do problema é dito ser *offline*. No problema *online*, não se conhecem as solicitações futuras em um dado instante *t*, ou seja, σ_t é desconhecido se $t' > t$. Um algoritmo *online* A é dito ser *c-competitivo* se existirem constantes *c* e α, tal que $C_A(\sigma) \leq c \cdot C_{OPT}(\sigma) + \alpha$ para qualquer seqüência σ. Nessa expressão, $C_A(\sigma)$ e $C_{OPT}(\sigma)$ correspondem aos custos associados ao algoritmo A e ao algoritmo ótimo *offline*, respectivamente. A constante *c* é chamada razão de competitividade de *A*, considerando que *A* é determinístico.

4. **Aprendizagem por Reforço – O Algoritmo *Q-Learning***

O OtimSPT pode ser visto como um processo de decisão em múltiplas etapas, onde a cada instante $(i = 1,2,\cdots,m)$ considerado, uma decisão deve ser tomada sobre qual SPT deve ser deslocada para atender a solicitação de intervenção σ_i. Esse processo é *markoviano*, uma vez que a decisão a ser tomada no instante t_i depende apenas das informações disponíveis nesse instante. Desta forma, considera-se o OtimSPT um problema de controle ótimo em processos de decisão *markovianos*. Para a solução de problemas de decisão *markovianos*, uma abordagem bastante eficiente é a Aprendizagem por Reforço (Sutton et al., 1998). Trata-se de um processo de aprendizado não-supervisionado, que ensina como mapear situações para ações, de modo a maximizar (minimizar) um sinal numérico de reforço. O aprendizado baseia-se na iteração de um agente de aprendizagem com o seu ambiente. O agente irá interagir com o ambiente para realizar o aprendizado. O ambiente é representado por um conjunto finito de estados $S = \{s_1, s_2, \ldots, s_k\}$, cujos elementos s_t representam as localizações das SPTs na bacia petrolífera *G* num instante de tempo discreto t [2]. Para cada estado, está associado um conjunto $A(s_t)$ finito de ações a_t, onde cada ação representa o deslocamento de uma SPT para atender uma solicitação. Para cada ação tomada e caso uma transição de estados $s_t \rightarrow s_{t+1}$ tenha ocorrido, uma avaliação do resultado desta ação, na forma de um sinal numérico r_{t+1}, denominado *reforço*, é apresentada ao agente pelo ambiente. O processo de aprendizagem tem por finalidade orientar o agente a tomar as ações que venham a maximizar (minimizar) o somatório dos sinais de reforços recebidos ao longo do tempo, denominado *retorno*, o que nem sempre significa maximizar o reforço imediato a receber. Matematicamente (para Tarefas de Horizonte Infinito[3]), tem-se:

$$R_t = r_{t+1} + \gamma \cdot r_{t+2} + \gamma^2 \cdot r_{t+3} \ldots = \sum_{k=0}^{\infty} \gamma^k \cdot r_{t+k+1} \tag{1}$$

onde R_t representa o retorno (somatório dos reforços) recebido ao longo do tempo, r_{t+k} o sinal de reforço imediato, e γ é o fator de desconto, $0 \leq \gamma \leq 1$, utilizado para garantir que R_t seja finito.

Uma política, expressa por π, representa o comportamento que o agente deve seguir para alcançar a maximização (minimização) do retorno. Uma política $\pi(s,a)$ é um mapeamento de estados *s* em ações *a*, tomadas naquele estado, e representa a probabilidade de seleção de cada uma das ações possíveis, de tal forma que as melhores ações correspondem às maiores probabilidades de escolha.

Para se saber se um agente está conseguindo escolher as melhores ações, alguma medida de qualidade dessas ações precisa ser considerada. Para tanto, tem-se o conceito de *Função de Valor estado-Ação Q(s,a),* a qual é um número que representa uma estimativa do quão bom é para o agente estar em um dado estado *s* e tomar uma dada ação *a*, quando se está seguindo uma política π qualquer. O termo representa o valor esperado de retorno para o estado $s_t = s$ (estado atual), isto é, o somatório dos reforços aplicando-se a taxa de desconto γ. Matematicamente, tem-se:

$$Q^{\pi}(s,a) = E_{\pi}\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \middle| s_t = s, a_t = a\right\} \tag{2}$$

A questão central da Aprendizagem por Reforço pode então ser colocada:

- Dada uma política $\pi(s,a)$, qual a melhor forma de estimar $Q(s,a)$?

[2] O termo estado é sinônimo de *Configuração das SPTs*.

[3] Tarefas de Horizonte Infinito são problemas em que a iteração entre o agente e o ambiente continua sem limite, de maneira contínua. Como por exemplo, o aprendizado para realizar o controle de sensores em uma indústria.

- Conhecendo-se uma resposta afirmativa para a questão anterior, de que forma essa política pode ser modificada, tal que $Q(s,a)$ aproxime o valor ótimo dessa função, e a conseqüente política ótima correspondente possa ser obtida?

Vários são os resultados encontrados na literatura que aportam uma resposta a estas questões, notadamente aqueles baseados em *Programação Dinâmica* (Bellmann, 1957), *Métodos de Monte Carlo* e *Diferenças Temporais* (Sutton et al., 1998). Neste trabalho, optou-se por utilizar o algoritmo *Q-Learning*, desenvolvido por Watkins et al. (1992). Dentre as vantagens desse algoritmo está o fato dele aproximar diretamente o valor ótimo de $Q(s,a)$, independentemente da política utilizada. Os valores de $Q(s,a)$ são atualizados segundo a equação:

$$Q(s,a) = Q(s,a) + \alpha\left[r_{t+1} + \gamma \max_a Q(s_{t+1},a) - Q(s,a)\right] \tag{3}$$

onde α é a taxa de aprendizagem.
O algoritmo que implementa o *Q-Learning* está mostrado na Figura 1.

Inicialize $Q(s,a)$ arbitrariamente;
Repita (para cada episódio)
  Inicializar s
  Repita (para cada instante de tempo)
    Escolher *a* para *s* usando a política π ;
    Dado a ação *a*, observe *r*, s' ;
    $Q(s,a) = Q(s,a) + \alpha\left[r_{t+1} + \gamma \max_a Q(s_{t+1},a) - Q(s,a)\right]$;
    $s \rightarrow s'$;
  até condição de parada estabelecida

Figura 1. Algoritmo *Q-Learning*.

Dado que a convergência do algoritmo só é garantida se todos os pares estado-ação forem visitados infinitas vezes, a escolha da política a ser utilizada no *Q-Learning* deve garantir que todos os pares tenham uma probabilidade não nula de serem visitados. Isto pode ser alcançado utilizando-se uma política ε-*gulosa*, definida por:

$$\pi(s,a) = \begin{cases} 1 - \varepsilon + \dfrac{\varepsilon}{|A(s)|}, \text{ se } a = a^* = \arg\max_a Q(s,a) \\ \dfrac{\varepsilon}{|A(s)|}, \ a \neq \text{a}^* \end{cases} \tag{4}$$

5. Modelagem do OtimSPT Segundo a Aprendizagem por Reforço

O objetivo a ser alcançado pela modelagem da Aprendizagem por Reforço para a solução do OtimSPT é encontrar uma política π que indique os movimentos das SPTs de modo que o custo total nestes deslocamentos seja o menor possível.

Do ponto de vista da Aprendizagem por Reforço, o problema pode ser modelado como segue: o estado do ambiente é representado por uma configuração possível das *k* SPTs, ou seja, por k-tuplos do tipo $s = \{SPT_1, SPT_2, \ldots, SPT_k\}$.

As ações correspondem aos deslocamentos permitidos das SPTs em um estado válido. Em cada estado, e considerando o surgimento de uma intervenção σ_j em um dos *n* poços da bacia petrolífera, um dos k SPTs será deslocado. Deste modo, tem-se que o número de *ações permitidas* para atender σ_j é *k*. Neste trabalho, só será considerado o surgimento de uma intervenção por vez, deixando a análise de múltiplas intervenções para trabalhos futuros.

O número total de estados possíveis é dado pela combinação fatorial do número de poços pelo de SPTs, de acordo com a expressão $C_{n,k} = \dfrac{n!}{(n-k)!k!}$, onde *n* indica o número de poços e *k* o número de sondas disponíveis. Diante disto, pode-se inferir que para armazenar os valores da função Q, uma estrutura de dimensão $C_{n,k} \cdot k \cdot n$ deve ser utilizada, onde $C_{n,k}$ representa o número total de estados válidos e $k \cdot n$ o número de ações permitidas por estado.

Deve-se observar que a dimensão da estrutura de armazenamento utilizada pela Aprendizagem por Reforço cresce em função do número de poços e das SPTs. Ao se analisar esse crescimento, percebe-se que o mesmo ocorre de maneira exponencial. Este problema, denominado *Maldição do Dimensionamento*, foi introduzido por Belmann (1957) e significa a impossibilidade de execução de um algoritmo para certas instâncias de um problema, pelo esgotamento de recursos computacionais para obtenção de sua saída.

Infere-se, conseqüentemente, que para que se possa aplicar a solução baseada no *Q-Learning* a um número significativo de poços, algum mecanismo deve ser criado para contornar o problema do dimensionamento inerente ao método da Aprendizagem por Reforço.

Para superar o problema supracitado, uma solução hierarquizada baseada na Aprendizagem por Reforço e no Método Guloso foi proposta. A idéia geral é aplicar a Aprendizagem por Reforço a um número reduzido de poços, selecionados seguindo um critério específico, do conjunto de poços da bacia e tentar generalizar o aprendizado obtido neste treinamento para outros pares estado-ação não visitados. Quando a generalização não for possível, utiliza-se o critério guloso para selecionar o servo a ser deslocado.

Os passos do algoritmo que implementa a solução hierarquizada em Português Estruturado são apresentados na Figura 2:

1. Divida o conjunto de poços em grupamentos de proximidade;
2. Para cada grupamento formado
    a. Escolha um nó para representar o grupo – nós-centro;
    b. Execute o algoritmo da Aprendizagem por Reforço no conjunto de nós que compõem o grupo;
3. Execute o algoritmo da Aprendizagem por Reforço nos nós escolhidos no passo anterior - nós-centro;
4. Se o par estado-ação não foi visitado no passo anterior e a generalização não puder ser feita, utilize o critério guloso para escolher o servo a ser deslocado.

Figura 2. Algoritmo Hierarquizado.

Para realizar a divisão do conjunto de *n* poços da bacia em *x* grupos ($x < n$), utilizou-se o Algoritmo de *Boruvka* (Goldbarg at al., 2000), a partir da *Árvore Geradora Mínima* (Goldbarg at al., 2000) da bacia (grafo). O número de grupos formados é fornecido como parâmetro.

O critério de escolha dos poços que representam cada grupo, os quais foram denominados *nós-centro*, é a média do custo para deslocar uma SPT, considerando sempre como ponto inicial o poço em questão, a todos os demais poços do seu grupo. Em outras palavras, o poço selecionado será aquele que possuir, em média, o menor custo para deslocar uma sonda para todos os outros poços do seu grupo, considerando sempre como ponto inicial de partida o poço em questão.

Nos primeiros passos do algoritmo hierarquizado, o conjunto de *n* poços do problema foi dividido em *x* grupamentos de proximidade, e para cada grupamento um nó-centro foi selecionado, totalizando *x* nós-centro. A Aprendizagem por Reforço será aplicada neste conjunto de *x* nós-centro e em cada conjunto de poços que compõem os *x* grupamentos, mantendo-se constante o total de *k* SPTs. A execução da Aprendizagem por Reforço, neste conjunto reduzido de nós, mantém as mesmas características da execução se a mesma ocorresse no conjunto de *n* nós que compõem a bacia petrolífera completa. Porém, deve-se observar que os servos e os possíveis locais de demanda estarão localizados somente nos nós selecionados durante cada execução.

A redução da estrutura de armazenamento usada pela Aprendizagem por Reforço para armazenar os valores da função Q ocorrerá em função de um *fator de redução* δ, de acordo com a expressão, $C_{\frac{n}{\delta},k} \cdot k \cdot \frac{n}{\delta}$. O valor de δ é função do número de grupos formados.

Quando os servos e as demandas pertencerem a grupos distintos, todos eles, e os mesmos não estiverem nos nós-centros dos seus respectivos grupos, serão considerados como se estivessem. A partir desta suposição, utiliza-se o conhecimento obtido durante a execução da Aprendizagem por Reforço para escolha do servo a ser deslocado, já que o aprendizado com os servos e as demandas localizados nos respectivos nós-centros dos seus grupos já foi realizado. Esta transposição da posição dos servos ou da requisição faz com que o aprendizado possa ser utilizado em pares estado-ação que não foram visitados, generalizando o conhecimento obtido durante o aprendizado.

Quando o par estado-ação não for visitado pela Aprendizagem por Reforço e a generalização do conhecimento não puder ser feita, o critério para o deslocamento dos servos será o guloso, ou seja, o servo a ser deslocado será o que apresentar o menor custo para deslocar a sonda até o ponto que está solicitando intervenção.

Para verificar o desempenho da solução apresentada, experimentos foram realizados e o desempenho comparado com o Algoritmo *Harmonic* de solução do PKS, cuja taxa de competitividade já foi prova por Bartal et al. (1991). O *Harmonic* foi modelado de acordo com o OtimOPT. Foram realizados 3 experimentos. Em cada experimento, foram apresentadas 100 seqüências de intervenções. Uma seqüência é composta por 500 pedidos de intervenção σ_j em poços, sendo cada intervenção gerada randomicamente. Para cada experimento, alterou-se o número de grupos formados. Registrou-se o custo ao se deslocar as sondas, segundo os critérios de seleção estabelecidos pelos algoritmos.

No final dos experimentos, anotou-se, o menor e o maior custo de deslocamento para atender à seqüência de requisições, o valor médio dos custos e quantas vezes cada algoritmo foi o melhor em comparação ao outro, vale dizer, quantas vezes conseguiu obter o menor custo total decorrente dos deslocamentos das SPTs para atender a seqüência de requisições. O treinamento do algoritmo da Aprendizagem por Reforço nos nós-centro foi feito a partir de uma seqüência de 300.000 requisições apresentadas ao mesmo, considerando $\alpha = 0.1$, $\gamma = 0.9$ e $\varepsilon = 0.1$. Os resultados são apresentados na Tabela 1.

Tabela 1. Resultados obtidos nos testes.

	n = 100 k = 3 grupos = 18 it. = 3e05			
Algoritmos	Min	Max	μ	Vit.
Hierárquico	6010	6794	6421.53	65
Harmonic	6045	6879	6465.77	35
	n = 100 k = 3 grupos = 20 it. = 3e05			
Algoritmos	Min	Max	μ	Vit.
Hierárquico	6096	6867	6403.64	80
Harmonic	6176	6884	6498.21	20
	n = 100 k = 3 grupos = 24 it. = 3e05			
Algoritmos	Min	Max	μ	Vit.
Hierárquico	6055	7104	6517.07	75
Harmonic	6370	7699	6980.03	25
Legenda	μ - média dos Custos Vit. – Vitórias obtidas			

Pela análise dos resultados apresentados, constatou-se que o desempenho do algoritmo hierárquico foi mais eficiente do que o do *Harmonic*. Em seu melhor caso, obteve 80% das vitórias. Em seu pior caso, 65%. Em todos os casos, obteve mais vitórias do que o outro algoritmo.

6. Conclusão

A Aprendizagem por Reforço mostrou-se uma ferramenta eficaz no desenvolvimento de algoritmos para a otimização do Problema do Gerenciamento de Sondas de Produção Terrestre (OtimSPT). O OtimSPT foi tratado como uma aplicação online do Problema dos k Servos Homogêneos, e em seguida modelado segundo as características da Aprendizagem por Reforço, utilizando-se o *Q-Learning* como método de solução. Observou-se que o desempenho do algoritmo proposto foi melhor do que o de algoritmos consagrados para a solução do PKS e cujas taxas de competitividade já foram provadas. A redução dos custos associados aos deslocamentos das sondas e a redução da perda na produção de óleo decorrente da espera por serviços é o principal atrativo para se buscar novos métodos para o agendamento das SPTs. Para tanto, visa-se para trabalhos futuros a incorporação de novas restrições ao deslocamento das sondas, adequando o problema a condições reais dos deslocamentos das SPTs, de tal forma que se consiga verificar o desempenho da solução proposta para instâncias de problemas reais.

7. Referências

Bartal, Y. and Grove, E., *The Harmonic K-Server Algorithm is Competitive*. 23rd ACM Symposium on Theory of Computation, pages 260-266,1991.

Bellman, R., *Dynamic Programming*, Princeton University Press, 1957.

Borodin, A. and El-Yaniv, R., *Online Computation and Competitive Analysis*, Cambridge University Press, 1998.

Goldbarg, M. C. and Luna, H. P. C., *Otimização Combinatória e Programação Linear: Modelos e Algoritmos*, Editora Campus, 2000.

Manasse, M. S., McGeoch, L. A. and Sleator, D. D., *Competitive Algorithms for On-Line Problems*, In: Proc. 20th Annual ACM Symposium on Theory of Computing, pages 322-33, 1988.

Sutton, R. S. and Barto, A. G., *Reinforcement Learning: An Introduction*,The MIT Press, 1998.

Watkins, C. J. C. H. and Dayan, P., *Q-Learning: Machine Learning*, Kluwer Academic Publishers, pages 279-92, 1992.