CONTROLE ÓTIMO DE DESCARREGADORES DE NAVIOS UTILIZANDO APRENDIZADO POR REFORÇO

Leonardo Azevedo Scárdua[*]
scardua@lac.usp.br

José Jaime da Cruz[*]
jaime@lac.usp.br

Anna Helena Reali Costa[*]
anna.reali@poli.usp.br

[*]Escola Politécnica da USP, Av. Prof. L. Gualberto, Trav. 3, No. 158, Cidade Universitária, São Paulo, SP

ABSTRACT

This paper describes the use of Reinforcement Learning to the computation of optimal trajectories and anti-swing control of a ship unloader. The unloading cycle is divided into six phases and an optimization problem is defined for each of them. A TD(0) algorithm together with a multilayer perceptron neural network as a value function approximator is used in the optimization. The results obtained are compared to Optimal Control results.

KEYWORDS: Reinforcement Learning, Optimal Control, Anti-Swing Control, Ship Unloaders, Neural Networks.

RESUMO

Este trabalho descreve o uso de Aprendizado por Reforço para a obtenção de trajetórias ótimas e controle anti-balanço de um descarregador de navios. O ciclo de descarga é dividido em seis etapas e, para cada uma delas, é definido um problema de otimização. Para a solução deste são utilizados um algoritmo TD(0) juntamente com uma rede neural do tipo perceptron multicamada como um aproximador da função valor. Os resultados obtidos são comparados com resultados de Controle Ótimo.

PALAVRAS-CHAVE: Aprendizado por Reforço, Controle Ótimo, Controle Anti-Balanço, Descarregadores de Navios, Redes Neurais.

Artigo submetido em 20/12/2000
1a. Revisão em 5/12/2001; 2a. Revisão 8/8/2003
Aceito sob recomendação dos Eds. Associados Profs. Fernando Gomide e Takashi Yoneyama

1 INTRODUÇÃO

Um problema importante nas operações de descarga de navios é a otimização do movimento entre o navio e a moega, respeitando restrições impostas pelos equipamentos e satisfazendo condições de contorno específicas. O descarregador de navios é basicamente um sistema carro-pêndulo em que o comprimento do pêndulo pode ser variado, independentemente do movimento do carro (figura 1). Uma caçamba localizada na extremidade do cabo é usada para transportar o material a ser descarregado.

O problema da otimização do movimento da caçamba pode ser encarado como um problema de decisão seqüencial em tempo discreto, no qual um controlador deve decidir, em cada época de decisão, qual a melhor ação a executar, considerando seu objetivo de longo prazo. A dinâmica do movimento da caçamba pode ser modelada na forma de um sistema de equações diferenciais.

Há diversas soluções propostas para esse problema. Em Sakawa e Shindo (1982), os autores consideram a minimização da oscilação de containers sob determinadas condições de contorno. Um algoritmo para calcular a lei de controle ótimo de tempo mínimo é apresentado. Auernig e Troger (1987) abordam o problema de transferência de uma carga içada em tempo mínimo por meio do Princípio do Mínimo

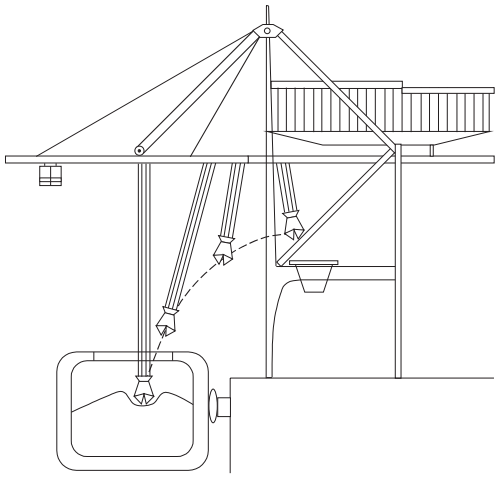

Figura 1: Esquema do descarregador de navios.

de Pontryagin. A modelagem mecânica e elétrica do sistema é realizada em detalhes. Tanto o movimento do carro porta-caçamba, como o movimento de içamento são determinados sujeitos à restrição de que a caçamba deve estar em repouso no início e no final do movimento. O problema de cálculo das trajetórias de tempo mínimo para guindastes é tratado por Golafshani e Aplevich (1995). Um modelo matemático do sistema é construído, sendo obtidas as trajetórias ótimas sujeitas a restrições nos estados e no controle. Uma abordagem heurística, baseada em lógica fuzzy, é apresentada por Liang e Koh (1997) para reduzir as oscilações no problema do movimento de cargas.

Há formas alternativas de tratar este problema que não se traduzem na solução de um sistema de equações diferenciais com condições no contorno. Pode-se modelar o problema de otimização do movimento do descarregador de navios como um Processo de Decisão de Markov (PDM). Em vista da dimensão do problema, é impraticável utilizar Programação Dinâmica ou técnicas de Aprendizado por Reforço (AR) que utilizam representação tabular para a função valor. Como alternativa pode-se utilizar uma forma compacta de representar esta função. É um fato bastante conhecido que as Redes Neurais (RNs) do tipo Perceptron Multi-Camada (PMC), treinadas com o algoritmo de retro-propagação são aproximadores universais de funções contínuas (Cybenko, 1989). Os PMCs têm sido usados com sucesso em aplicações de AR que envolvem espaços de estados de dimensões elevadas, tais como o jogo de gamão (Tesauro, 1995) e o controle de sistemas de elevadores (Crites e Barto, 1998).

Neste trabalho um algoritmo de AR baseado em diferenças temporais (Sutton e Barto, 1998) com um aproximador de funções PMC é utilizado para lidar com o número grande de possíveis estados do sistema. A idéia básica em AR é que o sistema de aprendizado esquematizado na figura 2 é capaz de aprender a resolver uma tarefa complexa por meio de repetidas interações com o ambiente. As únicas informações

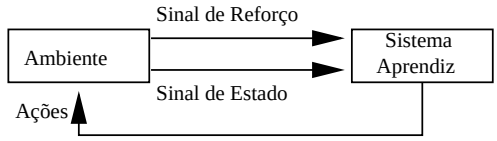

Figura 2: Esquema de um sistema de AR.

disponíveis são o estado do ambiente e um sinal escalar de reforço (recompensa), que é gerado pelo ambiente em resposta a transições de estado produzidas pelas ações executadas pelo sistema aprendiz. O objetivo de um aprendiz AR consiste em aprender uma política de escolha de ações, em cada estado, de forma a maximizar o reforço cumulativo no tempo.

O restante deste artigo está estruturado da seguinte forma: a seção 2 descreve a tarefa do descarregador de navios e o modelo usado para ela. A tarefa foi dividida em seis subtarefas, sendo fornecidos, para cada subtarefa, uma descrição e seu respectivo conjunto de condições de contorno. A seção 3 apresenta a abordagem AR desenvolvida para solucionar a tarefa. O ambiente experimental assim como os algoritmos desenvolvidos são descritos na seção 4. A seção 5 apresenta os resultados obtidos pelo uso da abordagem AR aplicada à otimização do movimento do descarregador e, finalmente, na seção 6 as conclusões são apresentadas.

2 O MODELO DO DESCARREGADOR DE NAVIOS

Esta seção baseia-se integralmente no modelo proposto por Auernig e Troger (1987). O descarregador pode ser modelado como um sistema carro-pêndulo conforme ilustra a figura 3. σ denota a posição do carro em relação a uma origem fixada, λ representa o comprimento do cabo[1] e θ, o ângulo de desvio do pêndulo em relação à vertical.

Em condições usuais de operação[2], $\hat{\theta}$ é um ângulo suficientemente pequeno de maneira que as equações de movimento podem ser linearizadas em torno de zero. Aplicando-se a formulação de Lagrange, obtém-se[3]:

$$\hat{\sigma}'' + \frac{\hat{m}_L}{\hat{m}_T}(\hat{\sigma}'' - \hat{\lambda}\hat{\theta}'' + 2\hat{\lambda}'\hat{\theta}' - \hat{\lambda}''\hat{\theta}) \quad = \quad \frac{\hat{F}_T}{\hat{m}_T} \qquad (1)$$

[1] O comprimento do cabo é definido como aquele entre o ponto de suspensão e o centro de massa da caçamba.

[2] O acento circunflexo é utilizado para indicar variáveis em forma dimensional; sua ausência indica que a variável se encontra adimensionalizada com relação à base definida a seguir. $\hat{\theta}$, medido em radianos, é a única exceção.

[3] Os símbolos $\hat{x}'$ e $\dot{x}$ denotam derivadas com relação aos tempos dimensional e adimensional, respectivamente.

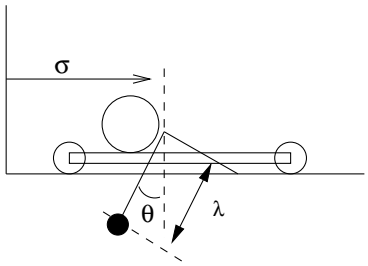

Figura 3: Esquema do sistema carro-pêndulo.

$$\hat{\lambda}\hat{\theta}'' + 2\hat{\lambda}'\hat{\theta}' + \hat{g}\hat{\theta} \quad = \quad \hat{\sigma}'' \qquad (2)$$

para o caso em que a política de içamento da caçamba é fixada a priori, isto é, a função $\hat{\lambda}(\hat{t})$ é dada. Nestas equações, $\hat{m}_L$ é a massa da caçamba, $\hat{m}_T$ é a massa total do carro e $\hat{F}_T$ é a força de tração do carro.

Há duas formas de controle do carro: na primeira, a força de tração é a variável de controle; na segunda, considera-se que a posição do carro é controlada e a aceleração $\hat{u}_T$ deste é a variável de controle. Este último caso é o considerado neste trabalho e, conforme Auernig e Troger (1987), fazendo-se $\hat{m}_L/\hat{m}_T = 0$ na equação 1, obtém-se:

$$\hat{\sigma}'' = \hat{u}_T, \qquad (3)$$

que, em conjunto com a equação 2, constitui o modelo do sistema na forma dimensional.

Definindo-se o seguinte conjunto de variáveis adimensionais:

$$\begin{array}{ll} u_T = \hat{u}_T/\hat{u}_{Tmax} & t = \hat{t}\sqrt{\hat{g}/\hat{\lambda}_{min}} \\ \sigma = (\hat{\sigma}\hat{g})/(\hat{u}_{Tmax}\hat{\lambda}_{min}) & \theta = (\hat{g}/\hat{u}_{Tmax})/\hat{\theta} \\ \lambda = \hat{\lambda}/\hat{\lambda}_{min} & v_T = (\hat{v}_T\sqrt{\hat{g}/\hat{\lambda}_{min}})/\hat{u}_{Tmax}, \end{array}$$

em que $\hat{u}_{Tmax}$ é a aceleração máxima do carro, $\hat{\lambda}_{min}$ é o valor mínimo do comprimento do cabo ao longo de todo o ciclo, $\hat{v}_T$ é a velocidade do carro, isto é, $\hat{v}_T = \hat{\sigma}'$ e $\hat{g}$ é a aceleração da gravidade, obtém-se as equações do sistema na forma adimensional:

$$\ddot{\sigma} \quad = \quad u_T \qquad (4)$$

$$\lambda\ddot{\theta} + 2\dot{\lambda}\dot{\theta} + \theta \quad = \quad u_T. \qquad (5)$$

O ciclo de descarga foi dividido em seis fases, a saber, *Partida do navio, Translação em direção à moega, Chegada à moega, Partida da moega, Translação em direção ao navio* e *Chegada ao navio*. As três primeiras fases são descritas em detalhes a seguir e constituem a primeira metade do ciclo. As três restantes, por serem similares, são descritas apenas brevemente. Para cada fase foi adotada uma origem para a medida da posição e uma para a variável tempo. Convenciona-se que $\dot{\sigma} > 0$ coresponde ao movimento do carro no sentido do navio para a moega.

2.1 Partida do Navio

Esta fase é definida da seguinte forma. Inicialmente, tanto o carro como a caçamba são supostos em repouso num dado ponto acima do navio; o comprimento do cabo tem seu valor máximo (λ_{max}). O içamento do cabo e o movimento do carro iniciam-se simultaneamente no instante que se convenciona como sendo $t = 0$. O içamento da caçamba é realizado a velocidade constante v_λ até que um dado valor λ_{min} do comprimento do cabo seja alcançado. Sendo assim, a duração desta fase é dada por:

$$t_{pn} = \frac{\lambda_{max} - \lambda_{min}}{v_\lambda} \qquad (6)$$

e a função $\lambda(t)$ resulta expressa por

$$\lambda(t) = \lambda_{max} - v_\lambda t \qquad (0 \leq t \leq t_{pn}), \qquad (7)$$

que deve ser usada na equação 5. Desta forma, a dinâmica do sistema é dada por um conjunto de duas equações diferenciais lineares de 2a. ordem, sendo a segunda delas (5), variante no tempo.

O objetivo desta fase é maximizar a distância percorrida pelo carro durante o içamento da caçamba. Admite-se que o intervalo de tempo t_{pn} seja suficiente para o carro atingir sua velocidade máxima v_{Tmax}.

No instante em que o içamento termina, deseja-se que a caçamba esteja em repouso com relação ao carro e na posição vertical. Além disso, a velocidade do carro deve ter atingido seu valor máximo v_{Tmax}.

A descrição acima conduz ao seguinte conjunto de condições de contorno $\sigma(0) = 0, \dot{\sigma}(0) = 0, \theta(0) = 0, \dot{\theta}(0) = 0, \dot{\sigma}(t_{pn}) = v_{Tmax}, \theta(t_{pn}) = 0, \dot{\theta}(t_{pn}) = 0$.

Note que a condição $\sigma(0) = 0$ estabelece o ponto de partida do carro acima do navio como a origem para a medida de posições. Além disso, por simplicidade, as seguintes restrições são impostas:

1. a aceleração do carro u_T pode tomar apenas os valores 0 ou 1;
2. a velocidade do carro v_T não pode ser superior a v_{Tmax}.

Note que a restrição 1 impede o carro de se movimentar para a frente e para trás. Ela reduz também o número de ações de controle possíveis em cada época de decisão. A restrição 2 representa um limite natural para o movimento do carro.

2.2 Translação em Direção à Moega

Durante esta fase o carro se move com velocidade máxima v_{Tmax} em direção à moega e o comprimento do cabo é man-

tido constante e igual a λ_{min}. Uma vez que o movimento do carro é retilíneo e uniforme, a caçamba permanece no mesmo estado em que se encontrava ao final da etapa *Partida do navio*, a saber, em repouso com relação ao carro e na posição vertical.

A duração desta fase depende dos resultados da próxima fase, uma vez que esta determina a distância necessária para o carro desacelerar e parar sobre a moega.

2.3 Chegada à Moega

Nesta fase, supõe-se o carro se movendo inicialmente em direção à moega com velocidade v_{Tmax} e a caçamba é admitida em repouso com relação ao carro e na posição vertical. O objetivo é minimizar a distância de frenagem necessária para parar o carro de maneira que, no final desta fase, a caçamba esteja novamente em repouso na vertical. O comprimento do cabo é mantido constante e igual a λ_{min}.

A duração desta fase não é conhecida a priori e deve ser obtida pelo controlador. Denotando-a por t_{cm} e adotando a origem de tempo $t = 0$ como o instante em que a fase se inicia, tem-se o seguinte conjunto de condições de contorno:$\sigma(0) = 0, \dot{\sigma}(0) = v_{Tmax}, \theta(0) = 0, \dot{\theta}(0) = 0, \dot{\sigma}(t_{cm}) = 0, \theta(t_{cm}) = 0, \dot{\theta}(t_{cm}) = 0$.

Note-se que a condição $\sigma(0) = 0$ fixa o ponto inicial da frenagem como a origem para a posição do carro. Em outras palavras, a distância de frenagem $\sigma(t_{cm})$ define o ponto em que o carro deve estar no início desta fase de modo que, no final da fase, a caçamba esteja exatamente na posição de descarga desejada acima da moega.

Além disso, por simplicidade, supõe-se que as seguintes restrições devam ser satisfeitas:

1. a aceleração do carro u_T pode tomar apenas os valores 0 ou -1;
2. a velocidade do carro v_T não pode ser inferior a 0.

Uma vez que λ é constante durante esta fase, tem-se $\dot{\lambda} = 0$ na equação 5.

2.4 Demais Fases

As fases restantes do ciclo de descarga são bastante similares às descritas acima.

A etapa *Partida da moega* é análoga à *Chegada à moega*. O objetivo é acelerar o carro de volta em direção ao navio desde o repouso até sua velocidade máxima. Considera-se a caçamba inicialmente em repouso na posição vertical e deseja-se que ela resulte nessa mesma condição em relação ao carro no instante final da etapa. O cabo é mantido com comprimento constante e igual a λ_{min}. A solução que se obtém nesse caso, obviamente, é a mesma da fase *Chegada à Moega*, exceto pela troca dos papéis entre os instantes inicial e final, assim como das condições de contorno.

A etapa *Translação em direção ao navio* é idêntica à fase *Translação em direção à moega*, pois o movimento do carro é retilíneo e uniforme.

Por fim, no que se refere à troca de papéis entre instantes e condições iniciais e finais, a *Chegada ao navio* e a *Partida do navio* guardam entre si a mesma semelhança que existe entre as duas primeiras etapas mencionadas no início desta seção. Contudo, como os movimentos verticais da caçamba têm sentidos opostos nas duas etapas, as soluções são distintas, pois dependem de $\dot{\lambda}$ (que é negativa para a *Partida do navio* e positiva para a *Chegada ao navio*).

3 O APRENDIZADO POR REFORÇO

De acordo com (Crites e Barto, 1998), da perspectiva da teoria de controle, as técnicas de AR são formas de obter soluções para problemas de controle ótimo estocástico em que o agente que toma as decisões é um controlador e o ambiente é o sistema a controlar. Dentro deste esquema, o objetivo é maximizar um dado índice de desempenho. O problema de decisão seqüencial a resolver é modelado como sendo um PDM em que o objetivo do agente aprendiz é maximizar o reforço recebido durante sua vida.

Formalmente, um PDM é definido como sendo (S, A, R, T) (Kaelbling et al., 1996), onde:

- S é o conjunto de estados;
- A é o conjunto de ações;
- R é uma função de reforço escalar, $R : S \times A \rightarrow \Re$;
- T é uma função de transição de estado, $T : S \times A \rightarrow \Pi(S)$, onde um membro de $\Pi(S)$ é uma distribuição de probabilidade sobre S. $T(s, a, s')$ representa a probabilidade de alcançar o estado s' a partir de s, executando a ação a.

A tarefa de um agente AR consiste em aprender, por tentativa e erro, a política $\pi : S \rightarrow A$ que mapeia o estado corrente s na ação $a \in A$ desejada a ser executada em $s \in S$.

O aspecto mais importante no contexto de PDMs é a *Propriedade de Markov*. Diz-se que um problema de decisão seqüencial tem a *Propriedade de Markov* se a decisão no início do passo t pode ser tomada como uma função apenas do estado no mesmo instante s_t. Em outras palavras, a *Propriedade de Markov* estabelece que a função de transição de estados e a função de reforço são independentes de qualquer

estado prévio do ambiente ou das ações prévias executadas pelo agente aprendiz.

O problema de controle das oscilações do pêndulo apresenta a *Propriedade de Markov* uma vez que, dado o estado composto por $\Theta, \dot{\Theta}, \lambda, \sigma, \dot{\sigma}$, não é necessário considerar a história passada do sistema para tomar uma decisão num dado instante.

4 SOLUÇÃO DO PROBLEMA

Esta seção descreve os passos envolvidos na solução do problema utilizando AR. Deve-se notar que a solução da *Partida da moega* é simétrica no tempo com relação à da *Chegada à moega*. Por outro lado, a solução da *Chegada ao navio* guarda certa similaridade com a da *Partida do navio*. Em vista disso, apenas as soluções correspondentes às fases *Partida do navio, Translação rumo à moega* e *Chegada à moega* são apresentadas.

4.1 Implementação

Para tratar o problema de controle apresentado como um PDM, o sistema de AR ilustrado na figura 2 deve ser implementado como um sistema a tempo discreto. O sistema é então simulado com uma ação de controle escolhida em cada passo de discretização. O aprendizado é realizado independentemente para cada fase do movimento, ou seja, cada fase é simulada até que um resultado satisfatório seja obtido.

O bloco *Ambiente* da figura 2 é implementado por um simulador que descreve as respostas dinâmicas do sistema a ações de controle e o bloco *Sistema Aprendiz* é implementado na forma de um algoritmo de aprendizado TD(0) (Sutton e Barto, 1998). No que se segue, os componentes do sistema de AR são descritos em detalhes.

4.1.1 O Simulador

No início de cada episódio de simulação, iniciam-se as variáveis de estado do sistema com o valor inicial correspondente àquela fase. Em seguida, o simulador executa o seguinte ciclo:

1. Lê a ação a executar. Esta ação deve ter sido selecionada pelo *Sistema Aprendiz*.
2. Avança uma unidade de tempo de simulação integrando as equações diferenciais que descrevem o sistema.
3. Armazena o estado do sistema para o próximo passo de discretização.
4. Calcula o reforço imediato que se segue à transição de estados.

Cálculo do Reforço Associado à Transição de Estados. Para alguns problemas formulados como PDMs, há uma escolha natural para o sinal de reforço (Singh e Bertsekas, 1996). Infelizmente, neste caso, não existem quantidades associadas à tarefa que possam ser diretamente usadas como reforço. A solução adotada envolveu a definição de um tempo limite t_{gs} para finalizar um episódio de simulação. Quando o tempo de simulação atinge este limite pré-estabelecido, o estado do descarregador é comparado ao estado final desejado da subtarefa correspondente. O reforço é então determinado através da diferença entre o estado corrente e o estado final considerado, indicando ocorrência de erros. O procedimento dedicado ao cálculo deste reforço pode ser descrito como:

1. Se o limite de tempo t_{gs} foi atingido:
   (a) Se o sistema não está no estado final, então ocorreu uma violação de restrição: retorne como reforço o negativo do valor absoluto da diferença escalar entre os valores atual e desejado da primeira variável de estado que não satisfaz a condição desejada para o estado final.
   (b) Se o estado final desejado foi atingido: termine a simulação.
2. Retorne 0 (zero) como reforço.

Observe que este procedimento avalia somente as situações extremas, aplicando uma punição caso o sistema não tenha chegado ao estado final dentro do limite de tempo pré-estabelecido ou terminando a simulação caso o objetivo tenha sido alcançado. Para todas as outras transições intermediárias, zero é retornado como reforço, significando que o simulador não sabe como avaliar tais transições. Esta falta de conhecimento causa um impacto negativo no tempo de aprendizado, uma vez que o algoritmo TD(0) não pode adquirir conhecimento a partir de tais transições.

Quando ocorre uma violação de restrição, o simulador inicia uma nova simulação desde seu início, considerando como condições iniciais do procedimento de integração aquelas correspondentes à fase que está sendo otimizada.

Otimização do tempo. O procedimento de atribuição de reforço verifica se o sistema atingiu o estado final desejado apenas após o decurso de um intervalo tempo t_{gs} dado pelo usuário. O processo de otimização do tempo para cada fase é executado expondo o controlador a uma seqüência de tempos de simulação decrescentes, conforme descrito a seguir:

1. Leia o valor inicial de t_{gs}.
2. Atribua ao estado inicial o valor correspondente à fase atual.

3. Inicie o controlador sem qualquer conhecimento prévio.
4. Inicie o contador de tempo.
5. Enquanto o contador de tempo for inferior a 20 minutos (valor empírico):
   (a) Avança um passo na simulação.
   (b) Se o estado do sistema for igual ao estado desejado:
      i. Mantenha a solução obtida como candidata à solução final.
      ii. Reduza t_{gs}.
      iii. Calcule o reforço baseado na diferença entre o estado atual e o final desejado.
      iv. Atualize o conhecimento do controlador.
6. Retorne a solução candidata como solução final.

4.1.2 O Sistema Aprendiz

No início de cada passo de simulação, o algoritmo TD(0) escolhe a ação que ele avalia como sendo aquela capaz de gerar a maior recompensa a longo prazo. Para cada par estado-ação possível, esta estimativa é traduzida por um número real chamado valor Q (Watkins e Dayan, 1992). O valor Q de um par estado-ação é atualizado após cada transição de estados. Em essência, tudo que for aprendido estará codificado nos valores de Q. O modelo de horizonte infinito descontado foi usado para o valor cumulativo esperado adquirido por seguir uma política arbitrária π a partir do estado inicial s_i.

O estado do simulador é composto por $\Theta, \dot{\Theta}, \lambda, \sigma, \dot{\sigma}$. Estas são todas variáveis contínuas, o que leva a um número infinito de estados possíveis. Entretanto, para fins de simulação, o número de estados possíveis é finito. Este fato pode ser observado facilmente considerando que:

1. Para cada fase do movimento os episódios de simulação sempre têm uma duração máxima de tempo e iniciam com as mesmas condições iniciais.
2. O número de ações de controle possíveis em cada época de decisão é limitado (finito).
3. Ao tomar a ação a_t quando no estado s_t, o sistema é conduzido ao estado s_{t+1} com probabilidade 1, onde s_{t+1} é determinado pela dinâmica do sistema.

Note, entretanto, que esse número cresce drasticamente com o tempo de simulação. Utilizando um passo de 0,05s e considerando simulações de 6s, há 120 épocas de decisão. Como o número de ações de controle possíveis em cada época de decisão é 2, então há 2^{120} seqüências de controle possíveis.

O número de estados possíveis torna impraticável o uso de uma representação tabular para armazenar os valores Q. A solução adotada utiliza uma RN para representar de forma compacta esses valores. A RN escolhida é um PMC com 7 nós na camada de entrada (5 nós para as variáveis de estado e 2 nós para as duas possíveis ações de controle), 20 nós sigmoidais na única camada escondida e um nó na camada de saída, o qual soma as saídas de todos os nós escondidos.

A RN é treinada com o algoritmo de retro-propagação (Haykin, 1999). O conjunto de pesos iniciais é escolhido aleatoriamente no intervalo [-1,1]. A mesma semente é utilizada, o que leva à mesma distribuição inicial de pesos para todas as fases do movimento.

Com o propósito de aumentar a velocidade de aprendizado, cada fase do movimento é controlada por uma RN, o que significa que um conjunto diferente de pesos é aprendido para cada fase da tarefa do descarregador de navios.

5 RESULTADOS

Esta seção mostra os resultados obtidos da otimização do movimento do descarregador de navios. As simulações foram realizadas para diversos valores de t_{gs}, sendo que nos interessam apenas os resultados correspondentes ao menor desses valores, já que o objetivo é minimizar a duração do ciclo de descarga. Os valores de t_{gs} muito maiores do que o ótimo podem levar a um tempo de processamento demasiadamente grande. Valores muito pequenos de t_{gs} correspondem a situações em que é impossível atingir o estado final desejado. Sendo assim, uma boa estimativa inicial para o valor de t_{gs} pode acelerar a convergência do procedimento. Caso seja muito difícil obter essa boa estimativa, deve-se utilizar o procedimento descrito a seguir.

Procedimento de dificuldade crescente. Para cada fase, em lugar de fornecer ao sistema de AR o estado final desejado como o estado objetivo, um enfoque diferente foi adotado. Estados finais com dificuldades crescentes para serem atingidos foram fornecidos ao sistema de AR, isto é, as condições do estado objetivo final foram progressivamente apertadas. Após o sistema aprender a atingir um objetivo "relaxado" num intervalo de tempo satisfatório, ele recebe como objetivo um novo estado, mais próximo do estado final realmente desejado, e assim sucessivamente até que ele aprenda a atingir o estado desejado.

Os resultados para as fases *Partida do navio*, *Translação em direção à moega* e *Chegada à moega* são apresentados a seguir. Foi utilizado um computador com 128 MB de RAM, processador Pentium II duplo de 266 MHz e sistema operacional Linux Red-Hat 5.2.

Os parâmetros dimensionais utilizados na simulação foram: $\hat{v}_{Tmax} = 3,83$m/s ,$\hat{\lambda}_{min}$ =9m, $\hat{\lambda}_{max} = 30$m, $\hat{u}_{Tmax} = 2,5m/s^2$ e $\hat{v}_{\lambda}$ =2m/s. A distância horizontal entre o navio e a moega é de 39,45m (Cruz et al., 1999).

5.1 Partida do Navio

Após alguns experimentos com t_{gs}, chegou-se ao valor de 5,8s.

O estado objetivo foi escolhido de início como $\hat{t}_{gs} = 5,8s, \hat{v}_T \geq 0,95\hat{v}_{Tmax}, \hat{v}_T \leq 1,05\hat{v}_{Tmax}, |\hat{\theta}| \leq 4°, |\hat{\dot{\theta}}| \leq 4°/s, \hat{\lambda} = \hat{\lambda}_{min}$.

O controlador necessitou simular 306 episódios para atingir o estado final desejado.

O estado objetivo foi então apertado para $\hat{t}_{gs} = 5,8s, \hat{v}_T \geq 0,95\hat{v}_{Tmax}, \hat{v}_T \leq 1,05\hat{v}_{Tmax}, |\hat{\theta}| \leq 2°, |\hat{\dot{\theta}}| \leq 2°/s, \hat{\lambda} = \hat{\lambda}_{min}$.

O controlador necessitou de 17 episódios para atingir um estado final satisfazendo estas restrições, a saber, $\hat{t}_{gs} = 5,8s, \hat{v}_T = \hat{v}_{Tmax}, |\hat{\theta}| \leq 0,2°, |\hat{\dot{\theta}}| \leq 0,2°/s, \hat{\lambda} = \hat{\lambda}_{min}$.

O tempo total de processamento foi de 234s.

Um procedimento heurístico foi introduzido para realizar uma sintonia fina na variável de controle, de maneira a reduzir as oscilações residuais. Após a última ação de controle bang-bang, o procedimento localiza o primeiro pico da velocidade angular do pêndulo superior a $0,1°/s$ e comanda, durante três passos de simulação, uma aceleração 25 vezes menor do que a aceleração máxima admissível.

Os resultados obtidos foram então $\hat{t}_{gs} = 5,8s, \hat{v}_T = 1,005\hat{v}_{Tmax}, |\hat{\theta}| \leq 0,03°, |\hat{\dot{\theta}}| \leq 0,03°/s, \hat{\lambda} = \hat{\lambda}_{min}$ e a distância percorrida pelo carro foi de 28,10m.

Este resultado foi considerado suficientemente próximo do estado objetivo dado pelas condições de contorno descritas na seção 2.1.

5.2 Chegada à Moega

Após alguns experimentos com valores de t_{gs} chegou-se a 5,8s.

Forneceu-se então ao controlador o seguinte estado objetivo: $\hat{t}_{gs} = 5,8s, \hat{v}_T = 0,1m/s, |\hat{\theta}| \leq 4°, |\hat{\dot{\theta}}| \leq 4°/s, \hat{\lambda} = \hat{\lambda}_{min}$.

Foram necessários nove episódios até atingir um estado final satisfazendo o primeiro objetivo.

O estado objetivo foi então apertado para $\hat{t}_{gs} = 5,8s, \hat{v}_T = 0,1m/s, |\hat{\theta}| \leq 2°, |\hat{\dot{\theta}}| \leq 2°/s, \hat{\lambda} = \hat{\lambda}_{min}$.

Foram necessários 126 episódios para atingir um estado final satisfazendo essas restrições, tendo-se obtido $\hat{t}_{gs} = 5,8s, \hat{v}_T = 0,035m/s, |\hat{\theta}| \leq 0,15°, |\hat{\dot{\theta}}| \leq 0,15°/s, \hat{\lambda} = \hat{\lambda}_{min}$.

O tempo total de processamento foi de 56s.

Um procedimento heurístico foi também adotado aqui para reduzir as oscilações residuais. Quando o ângulo do pêndulo atinge o pico, comanda-se, durante um passo de simulação, uma aceleração 25 vezes menor que a máxima admissível.

Os resultados então obtidos foram $\hat{t}_{gs} = 5,8s, \hat{v}_T = 0,024m/s, |\hat{\theta}| \leq 0,05°, |\hat{\dot{\theta}}| \leq 0,05°/s, \hat{\lambda} = \hat{\lambda}_{min}$ e a distância percorrida pelo carro foi de 7,52m.

5.3 Translação em Direção à Moega

A duração desta fase foi tomada como sendo aquela exatamente necessária para que, considerando o conjunto das três fases, o carro percorresse um total de 39,45m e parasse no ponto de descarga desejado, acima da moega. Considerando as distâncias percorridas nas fases de *Partida do navio* e *Chegada à moega*, resulta, portanto, que a extensão desta fase deve ser de 3,83m. Como o percurso se realiza à velocidade máxima, a sua duração é de 1,0s.

5.4 O Comportamento Resultante

O cabo, com 30m de comprimento inicial, foi recolhido à velocidade $\hat{v}_{\lambda}$ até atingir o comprimento de 10m, conforme ilustra a figura 4.

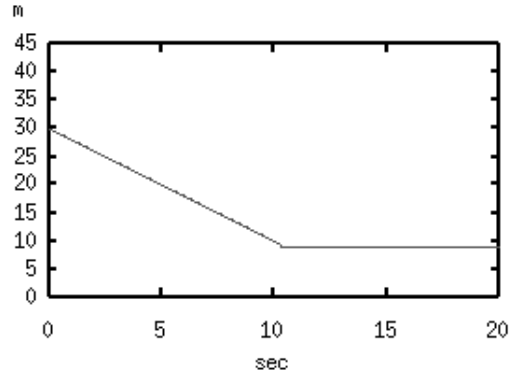

Figura 4: Comprimento do cabo em função do tempo.

As figuras 5 a 9 mostram a evolução temporal das variáveis do sistema para o semi-ciclo constituído pelas três fases des-

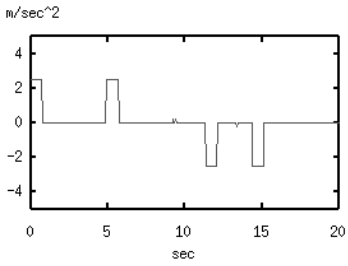

Figura 5: Ações de controle em função do tempo.

.

critas acima.

A seqüência de ações de controle em função do tempo é mostrada na figura 5. A fase *Partida do navio* tem uma duração de aproximadamente 5,7s e contém dois pulsos de aceleração. De maneira similar, a fase *Chegada à moega* dura 3,8s e a seqüência de ações de controle é composta de dois pulsos de desaceleração. A velocidade do carro e a sua posição em função do tempo são apresentadas nas figuras 6 e 7, respectivamente.

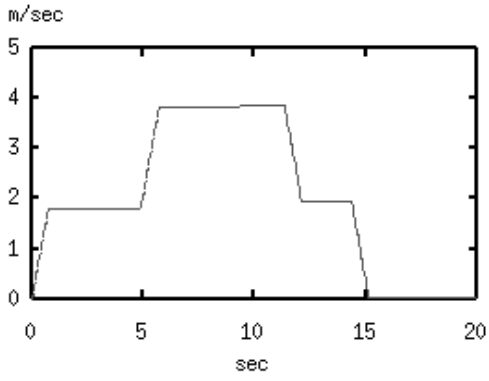

Figura 6: Velocidade do carro em função do tempo.

As figuras 8 e 9 mostram, respectivamente, o ângulo de desvio do pêndulo em relação à vertical e sua velocidade angular em função do tempo. Nota-se, pois, que ocorrem movimentos angulares para trás e para a frente, respectivamente, durante as fases *Partida do navio* e *Chegada à moega*.

Os resultados apresentados acima reproduziram aqueles obtidos por meio de uma abordagem mais tradicional, baseada na discretização do problema, conforme consta em (Cruz et al., 1999). Em particular, o valor numérico ali relatado de 15,23s para a duração do semi-ciclo foi reproduzido exatamente pelo procedimento aqui descrito.

Deve-se notar que a abordagem baseada em AR utiliza o modelo matemático do problema considerado apenas para simu-

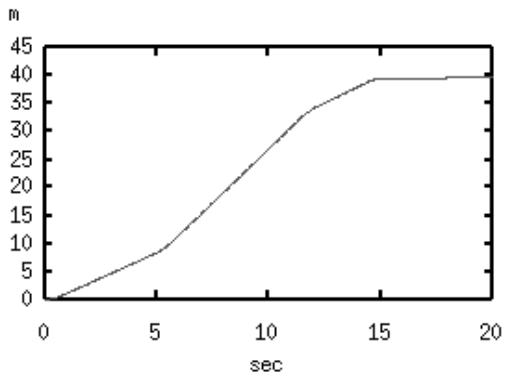

Figura 7: Posição do carro em função do tempo.

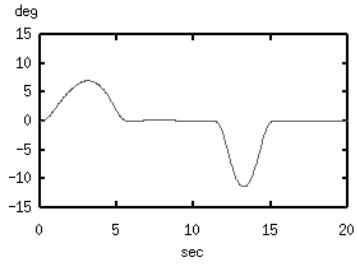

Figura 8: Ângulo do pêndulo em função do tempo.

lação. Sendo assim, a simplicidade talvez seja a vantagem mais significativa da abordagem baseada em AR em relação às formas tradicionais de obtenção numérica da solução de problemas de controle ótimo com restrições. Para ter uma idéia das dificuldades nestes casos veja, por exemplo, a referência (Auernig e Troger, 1987), para o problema específico do descarregador de navios, e a referência (Kamien eSchwartz, 1981), para problemas de controle ótimo de forma geral.

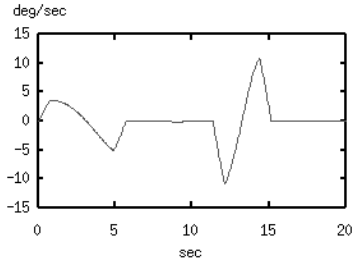

Figura 9: Velocidade angular do pêndulo em função do tempo.

6 CONCLUSÕES

O desempenho do esquema de AR depende fortemente da função de reforço adotada. Para alguns problemas de controle formulados como PDMs há uma escolha natural para o sinal de reforço, como se pode ver em (Singh e Bertsekas, 1996). Infelizmente este não é o caso do problema aqui tratado. Não há uma variável que possa ser utilizada de imediato como um sinal de recompensa para cada transição de estados. Para superar esta dificuldade, uma função de reforço foi construída considerando todas as variáveis de estado do sistema. É importante salientar que o desempenho acima poderá ser significativamente alterado se uma função de recompensa diferente for adotada.

Todo o conhecimento disponível a respeito do sistema a controlar deve ser utilizado para a definição da função recompensa. Neste trabalho admite-se que não haja conhecimento adequado para avaliar transições de estado intermediárias. Este fato tem um impacto negativo no tempo necessário para o aprendizado.

Outro fator que tem influência no desempenho do controlador é o conjunto inicial de pesos da RN. Os tempos de convergência dependem fortemente desse conjunto, como sugerem os diferentes tempos de convergência obtidos em correspondência às diversas fases do movimento.

Foram definidos estados objetivo de dificuldades crescentes. Esta abordagem produziu melhores resultados do que a forma tradicional de definir apenas o objetivo final para o controlador. Este comportamento pode ser explicado pelo fato de que um objetivo mais difícil de atingir acarreta punições mais severas, o que pode fazer com que os pesos da RN oscilem ou mesmo atinjam a saturação, uma vez que não está sendo utilizado passo decrescente de aprendizado.

Como comentário final, deve-se notar que o esquema de AR produziu um semi-ciclo de duração 15,23s, o que coincide com o valor obtido em (Cruz et al., 1999). Este trabalho mostra assim que controladores baseados em AR podem ser utilizados de maneira efetiva em problemas práticos de controle.

AGRADECIMENTOS

L.A. Scardua agradece ao CNPq (Proc. No. 141802/97-9), J.J. Cruz, ao CNPq (Proc. No. 304071/85-4) e à Fapesp (Proc. No. 97/04668-1) e A.H. Reali Costa, à Fapesp (Proc. No. 01/14588-2).

REFERÊNCIAS

Auernig, J. e Troger, H. (1987). Time optimal control of overhead cranes with hoisting of the load, *Automatica* **23**(4): 437–447.

Crites, R. H. e Barto, A. G. (1998). Elevator group control using multiple reinforcement learning agents, *Machine Learning* **33**: 235–262.

Cruz, J. J., Moraes, C. C. e Amaral, J. A. D. (1999). Modelagem do sistema de transporte carro-caçamba de descarregadores de minério em área portuária, *Relatório Interno*, USP.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function, *Mathematics of Control, Signals and Systems* **2**: 303–314.

Golafshani, A. e Aplevich, J. (1995). Computation of time-optimal trajectories for tower cranes, *Proc. IEEE CCA*, pp. 1134–1139.

Haykin, S. (1999). *Neural Networks: a comprehensive foundation 2nd ed*, Prentice-Hall.

Kaelbling, L.P., Littman, M.L. e Moore, A.W. (1996). Reinforcement Learning: A Survey, *Journal Of Artificial Intelligence Research* **4**: 237–285.

Kamien, M.I. e Schwartz, N.L. (1981). *Dynamic Optimization*, North-Holland.

Liang, Y. e Koh, K. (1997). Concise anti-swing approach for fuzzy crane control, *Electronics Letters* **33**(2): 167–168.

Sakawa, Y. e Shindo, Y. (1982). Optimal control of container cranes, *Automatica* **18**(3): 257–266.

Singh, S. e Bertsekas, D. (1996). Reinforcement learning for dynamic channel allocation in cellular telephone systems, *Advances in Neural Information Processing Systems: Proceedings of the 1996 Conference*, MIT Press, Cambridge, MA, pp. 974–980.

Sutton, R. S. e Barto, A. G. (1998). *Reinforcement Learning: an introduction*, Addison Wesley, MIT Press.

Tesauro, G. (1995). Temporal difference learning and td-gammon, *Communications of the ACM* **38**(3): 815–825.

Watkins, C. e Dayan, P. (1992). Q-learning, *Machine Learning* **8**: 279–292.