Utilização de Aprendizagem por Reforço para Modelagem Autônoma do Aprendiz em um Tutor Inteligente

Modalidade Artigo Completo

Marcus V. C. Guelpeli[1], Carlos H. C. Ribeiro[1] e Nizam Omar[2]

[1] Divisão de Ciência da Computação

Instituto Tecnológico de Aeronáutica –12228-900, São José dos- Campos São Paulo-SP

[2] Departamento de Engenharia Elétrica

Universidade Presbiteriana Mackenzie, São Paulo-SP

guelpeli@uol.com.br, {carlos, omar}@comp.ita.br

Resumo.Este trabalho tem como meta apresentar um módulo de diagnóstico para ser incluído na arquitetura tradicional de Sistemas Tutores Inteligentes. Neste módulo, é aplicada uma técnica de Aprendizado por Reforço (algoritmo Q-Learning) que possibilita modelar autonomamente o aprendiz. O maior valor de utilidade é calculado baseado em uma tabela de pares estado-ação, a partir da qual o algoritmo estima reforços futuros que representam os estados cognitivos do aprendiz. A melhor política a ser usada pelo tutor para qualquer estado cognitivo do aprendiz é disponibilizada pelo algoritmo de Aprendizagem por Reforço.

Palavras-chave: Sistemas Tutores Inteligentes, Aprendizado por Reforço, Algoritmo Q-Learning.

Abstract.The goal of this paper is to present a diagnostic module to be included in an Intelligent Tutoring System (ITS) architeture. In this module, a Reinforcement Learning technique (Q-Learning algorithm) is applied, making it possible to autonomously model the learner. A maximum utility value is calculated based on a state-action table upon which the algorithm estimates future rewards which represent the cognitive states of the learner. The best action policy to be used by the tutor at any learner's cognitive state is made available by the Reinforcement Learning algorithm.

Keywords: Intelligent Tutoring Systems, Reinforcement Learning, Q-Learning algorithm.

1.INTRODUÇÃO

Um Sistema Tutor Inteligente (STI) é uma evolução de sistemas CAI (Computer-Assisted Instruction), aperfeiçoado com as técnicas de Inteligência Artificial (IA).

Os STI possibilitam ao estudante a capacidade de aprender com um tutor, que serve como guia no processo. Ele deve se adaptar ao aprendiz, e não o contrário, como acontece no método tradicional. Com isso, é necessário um modelamento do aprendiz, para que o STI possa saber o que ensinar, a quem ensinar e como ensinar. De acordo com [Marietto, 2000] o STI deve ser capaz de mudar o nível de entendimento para responder às entradas do aprendiz, em vários níveis, podendo mudar as estratégias pedagógicas, de forma individualizada (de acordo com o ritmo e as características de cada aprendiz).

Em um STI o ensino é construído sobre uma base de conhecimento (domínio) criada por um especialista. Através da interação com o aprendiz, um STI modifica suas bases de conhecimentos, percebe suas intervenções, e possui a capacidade de aprender e adaptar-se às estratégias de ensino, de acordo com o desenrolar do diálogo com o aprendiz [Leite, 1999].

[Burns, Parllet & Redfield, 1991] salientam que as pesquisas de STIs, principalmente as de ensino-aprendizagem, devem enfocar as estratégias de ensino em particular, dando ênfase aos estados cognitivos de cada aprendiz.

Para ser inteligente, um tutor deve ser flexível, isto é, ter capacidade para aprender com o meio ambiente e atualizar seu conhecimento [Viccari, 1990].

Para aumentar e facilitar a autonomia de um tutor, este artigo propõe um processo de definição de perfil e o uso de um algoritmo de Aprendizado por Reforço (algoritmo Q-learning), através da introdução de um módulo de diagnóstico em uma arquitetura de STI, com o propósito de incorporar ao tutor a capacidade de modelar autonomamente o aprendiz.

O trabalho está organizado como segue. Na Seção 2 apresentam-se os conceitos de Aprendizado por Reforço e o algoritmo Q-Learning. A Seção 3 contém a estrutura do sistema utilizando AR e todos os pontos relevantes para o funcionamento do sistema. A Seção 4 contém experimentos e resultados. Finalmente, a Seção 5 apresenta as vantagens e desvantagens do método proposto, e sugere trabalhos futuros.

2. TÉCNICA DE APRENDIZADO POR REFORÇO

Afirma [Hendley,1992] que o uso de técnicas de Inteligência Artificial na construção de software educacionais torna-se cada vez mais importante, uma vez que seu estudo fará com que os mesmos tenham mais qualidade.

A arquitetura descrita neste trabalho utiliza uma técnica de Inteligência Artificial que é o Aprendizado por Reforço (AR), para modelar o aprendiz de forma dinâmica e autônoma.

AR permite ao agente adquirir uma capacidade de conhecimento do ambiente que não estava disponível em tempo de projeto [Sutton & Barto, 1998]. AR é baseada na existência de um crítico externo ao ambiente, que avalia a ação tomada, mas sem indicar explicitamente a ação correta. Formalmente, AR utiliza uma estrutura composta de estados, ações e recompensas conforme mostra a Figura1.

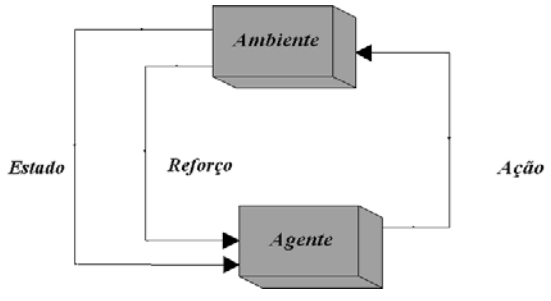

Figura 1. Um agente AR interagindo com seu ambiente.

O agente atua em um ambiente descrito por um conjunto de possíveis estados e pode executar, para cada estado, uma ação dentro de um conjunto de ações possíveis, recebendo um valor de reforços a cada vez que executa uma ação. Este reforço indica o valor imediato da transição estado-ação-novo estado. Ao longo do tempo, este processo produz uma seqüência de pares estado-ação, e seus respectivos valores de reforços. O objetivo do agente é aprender uma política que maximize uma soma esperada destes reforços a longo prazo.

Uma consideração importante a ser feita quando se lida com AR é conflito exploração X explotação. O agente deve lidar com o problema de haver um compromisso entre escolher a exploração de estados e ações desconhecidos, de modo a coletar nova informação, ou a explotação de estados e ações que já foram aprendidos e que irão gerar altas recompensas, de modo a maximizar seus reforços acumulados [Bellman, 1959]. Sendo assim, por um lado o agente deve aprender quais ações maximizam os valores das recompensas obtidas no tempo, e, por outro, deve agir de forma a atingir esta maximização, explorando ações ainda não executadas ou regiões pouco visitadas do espaço de estados. É importante, portanto, estabelecer uma política mista de explotação e exploração (política ε-greedy), que inclua a escolha intencional (com probabilidade ε) de se executar uma ação que não é considerada a melhor no estado atual, visando a aquisição de conhecimentos a respeito de estados ainda desconhecido ou pouco visitados. A escolha desta ação ocorre de forma aleatória. Em uma política de explotação pura (greedy) escolhem-se as ações que se julguem (talvez erroneamente, caso o algoritmo de AR ainda esteja longe da convergência) serem as melhores para resolver o problema.

No contexto deste trabalho, a técnica de AR terá como função modificar parâmetros e armazená-los em "esquemas de planos", que definem a forma de apresentar o material instrucional ao aprendiz. O grande desafio, neste caso, é escolher a melhor ação que - baseada no estado do aprendiz - possa mudar a estratégia de ensino, para obter resultados significativos, fornecedores de parâmetros indicativos de quão boa ou ruim está sendo uma determinada estratégia. O algoritmo utilizado (Q-learning) é baseado em estimativas de utilidades de pares estado-ação, e com estas estimativas podem-se gerar alternativas de novas estratégias pedagógicas. Neste caso, a aprendizagem das estratégias mais adequadas, que indicam a ação a escolher para cada estado do aprendiz, dar-se-á por reforço, através de uma estrutura de parâmetros adaptativos sobre os quais o algoritmo opera.

Formalmente, AR procura aproximar uma função que define a utilidade relativa dos pares estado-ação. Esta função de utilidade fornece indicações estimativas, mapeando os pares estado-ação em uma medida baseada na soma dos reforços esperados a longo prazo.

Para cada par estado-ação (s, a), é definido o reforço r(s, a), indicando uma conseqüência imediata da execução da ação a no estado s. O problema em AR é achar uma política ótima de ações (μ^*), ou seja, um conjunto de ações que maximizem, para cada estado s, os valores de utilidade Q(s, a). Com base nesses valores, o algoritmo de AR estima a ação de maior valor de utilidade – a ação que deverá ser executada pelo agente tutor.

2.1 Q-Learning

O algoritmo Q-Learning [Watkins,1989], pode ser usado para definir a escolha da melhor ação em AR. No Q-Learning, a escolha de uma ação é baseada em uma função de utilidade que mapeia estados e ações a um valor numérico. Neste artigo, o valor de utilidade Q(s, a) de um par (estado, ação), é calculado a partir de reforços medidos pela qualidade do estado cognitivo do aprendiz (modulo de diagnóstico). Cada valor Q(s, a) representa a soma de reforços esperada ao se executar a ação a no estado s, seguindo-se uma política ótima a partir de então. Portanto, uma vez que os valores Q(s, a) estejam bem estimados, a melhor ação a ser executada no estado s pode ser obtida simplesmente como arg $\max^a Q_t(s_{t+1},a)$.

O principal objetivo do algoritmo Q-Learning é estimar autonomamente, em cada estado s em que o aprendiz encontra-se, o valor Q(s, a) para cada possível ação a, e a partir daí permitir a obtenção da melhor ação (ou seja, a ação com maior valor de utilidade).

O Q-Learning normalmente usa uma tabela (Figura 2) para armazenar os valores de utilidade Q(s, a) estimados para os pares (estado, ação).

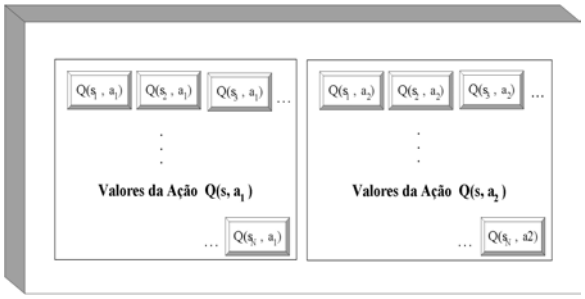

Figura 2. Tabela dos pares (estado - ação).

O algoritmo Q-Learning produz uma atualização dos valores Q da seguinte maneira:

Inicialize **Q(s, a)**.

Para cada instante **t** repita:

1- Observe estado $\mathbf{s_t}$ e escolha uma ação $\mathbf{a_t}$;

2- Observe o estado $\mathbf{s_{t+1}}$ e atualize $\mathbf{Q_t(s_t, a_t)}$ de acordo com

$$\mathbf{Q_{t+1}(s_t,a_t) = Q_t(s_t,a_t) + \alpha_t[r(s_t) + \gamma max_a Q_t(s_{t+1},a) - Q_t(s_t,a_t)];}$$

até **t** igual a limite de passos.

Onde:

- $Q_{t+1}(st,at)$ é o valor (qualidade) da ação a_t no estado s_t.
- $r(s_t)$ é o reforço imediato recebido no estado s_t.
- α é a taxa de aprendizado (normalmente definida entre 0 e 1).
- γ é a taxa de desconto temporal.
- t É uma seqüência discreta de passos no tempo, ou seja , t=0,1,2,3,....
- $max^aQ_t(s_{t+1},a)$ é o valor Q correspondente à ação com maior valor de utilidade no estado futuro.

Quanto mais próximo de 1 for o valor de γ, maior importância é dada aos reforços mais distantes no tempo.

No início, o algoritmo vai estar muito longe da utilidade ótima associada ao estado s, mas com o passar do tempo as estimativas de Q melhoram. De fato, os valores Q atualizados pelo algoritmo Q-Learning convergem para os corretos, desde que os pares (s,a) tenham sido visitados um número infinito de vezes [Watkins and Dayan, 1992]. Na prática, convergência para políticas de ação de boa qualidade é obtida apenas com exploração adequada do espaço de estados, durante um número razoável de iterações. A política de ações durante a execução do algoritmo nas fases iniciais de treinamento deve portanto garantir a exploração do espaço de estados (política ε-greedy).

3. ESTRUTURA DO SISTEMA

A Figura 3 ilustra a estrutura de funcionamento da Técnica de Aprendizado por Reforço para um STI. Todos os itens que fazem parte do sistema serão detalhados a seguir.

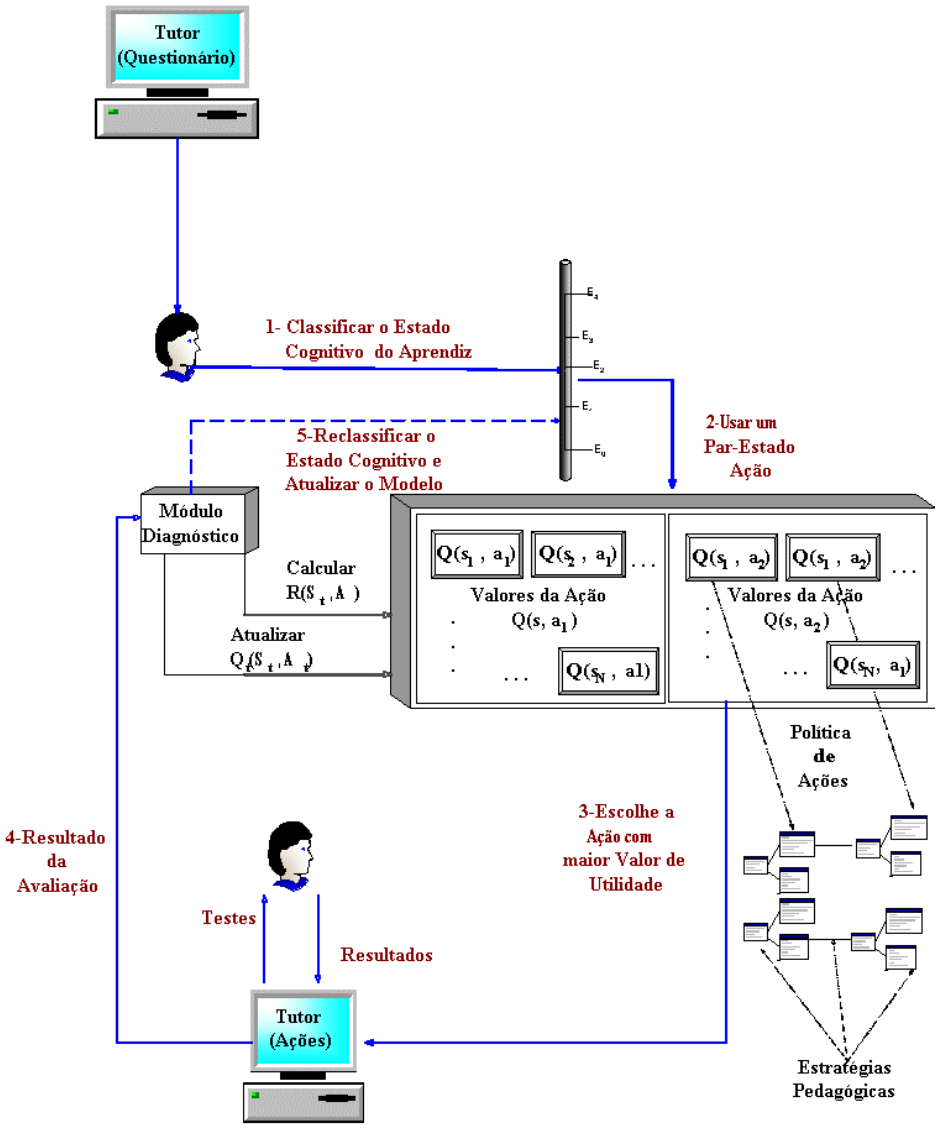

Figura 3. STI com a Técnica de AR.

3.1. Definição do Conjunto dos Estados do Estudante (Perfil)

A complexidade de estabelecer um perfil está na modelagem do aprendiz. O perfil vai indicar o estado cognitivo do mesmo a partir do comportamento observável e, com isso, poder indicar uma estratégia de ensino a ser utilizada.

Decisões Pedagógicas razoáveis são fatores de desafio para os tutores, já que as pessoas nem sempre conseguem representar seus próprios processos mentais e, algumas vezes, ficam confusas. De acordo com [Giraffa, 1999] um modelo realista do aluno implica em uma atualização dinâmica enquanto o sistema avalia o desempenho do estudante.

Este modelamento inicialmente dar-se-á através de questionários, onde é conhecido o perfil do aprendiz. Logo em seguida o sistema irá gerar um ciclo que classificará o aprendiz dentro das faixas distintas de conhecimentos (Figura 4), só que agora de forma dinâmica, usando as informações do módulo diagnóstico.

Como complementação, para construir um perfil do aprendiz, deve-se manter um log de respostas dadas por este anteriormente, para que ao retornar, seja desnecessária a repetição de tais questionários e possibilite a continuidade do processo.

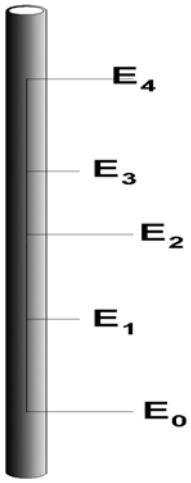

Figura 4. Classificação do Estado Cognitivo do Aprendiz.

A proposta de criação do módulo diagnóstico é de suma importância, pois ela definirá, através da qualidade de sua observação, como se dará o processo de aprendizado.

Este módulo diagnóstico produzirá - de acordo com a política de ações estabelecida – uma indicação de quão bem ou mal está sendo o desempenho do aprendiz em relação às ações produzidas pelo tutor. No contexto de AR, ele enviará reforços indicando o desempenho de modo intermitente, produzindo para cada par (s_t,a), um valor $r(s_t ,a)$, correspondendo à reforços positivos quando se produzem resultados favoráveis e reforços negativos para resultados desfavoráveis. Os valores de utilidade $Q_t(s_t,a_t)$ são atualizados na tabela de valores Q (Figura 2), para que com base nesses valores o tutor possa reclassificar o aprendiz.

O sistema classifica o estado cognitivo do Aprendiz através do resultado do questionário, usando uma escala de classificação (Figura 4) e obtendo seu perfil.

Com o perfil definido, o sistema usa uma tabela de pares estado-ação. Nesta, as ações representam uma estratégia pedagógica adotada pelo tutor para guiar o aprendiz.

O sistema escolhe a ação com o maior valor de utilidade baseado nos resultados do algoritmo Q-Learning.

Estes resultados são encaminhados para o Módulo Diagnóstico, que calcula reforços positivos ou negativos para o par estado-ação (s_t,a) e atualiza na tabela o valor de utilidade $Q_t(s_t,a_t)$.

Depois de atualizado o valor de utilidade na tabela, o sistema volta a reclassificar o aprendiz na escala.

4.EXPERIMENTOS

Os experimentos foram realizados sobre um protótipo de tutor, desenvolvido em linguagem C.

As simulações foram realizadas no ambiente com uma matriz 5x10 com elementos representados formalmente como:

- Um conjunto de estados $S=\{E_0, E_1, E_2, E_3, E_4\}$, cada um representando um possível estado cognitivo do aprendiz, em face da interação com o tutor. Estima-se que esse aprendiz tenha um grau cognitivo baseado nesses estados, onde:

E_0=> [0,2], E_1=>]2,4], E_2=>]4,6], E_3=>]6,8], E_4=>]8,10].

Os valores de 0 a 10, divididos nos cinco conjuntos em cada estado, representam uma métrica usada para diferenciar os estados, seguindo para isso uma escala crescente do grau evolutivo (Figura 4) do aprendiz na interação com o tutor.

- Um conjunto de Ações A={A_0, A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9} que podem ser escolhidas pelo tutor. Cada ação corresponde à aplicação de provas, exercícios, questionários, perguntas, trabalhos, testes, etc, ou combinações destes e outros dispositivos de avaliação, usados pelo tutor, seguindo as estratégias pedagógicas estabelecidas.
- Um conjunto de Reforços Instantâneos associados a cada estado visitado, ou seja,

E_0=>R=1-Ruim;
E_1=>R=3-Regular;
E_2=>R=5-Bom;
E_3=>R=7-Muito Bom;
E_4=>R=10-Excelente;

Foi definida uma metodologia de teste na qual foram criados três modelos não determinísticos: M_1(Ruim), M_2(Bom) e M_3(Excelente). Os modelos foram submetidos às simulações com 500, 1000 e 2500 passos, e de cada conjunto de simulação foi calculada uma média sobre vinte realizações, para obtenção dos dados finais. Note que estes modelos não são conhecidos *a priori* pelo tutor, que deverá estimar por AR uma política de ações em função dos reforços recebidos.

4.1. Resultados Obtidos

Os Resultados da Figura 5 mostram as médias dos reforços obtidos durante a simulação com 1000 passos da política pedagógica, para cada um dos modelos M_1, M_2 e M_3 (Não Deterministico) . Q-Learning consegue convergir para a melhor política de ação de cada modelo simulado, usando α=0.9 e γ=0.9.

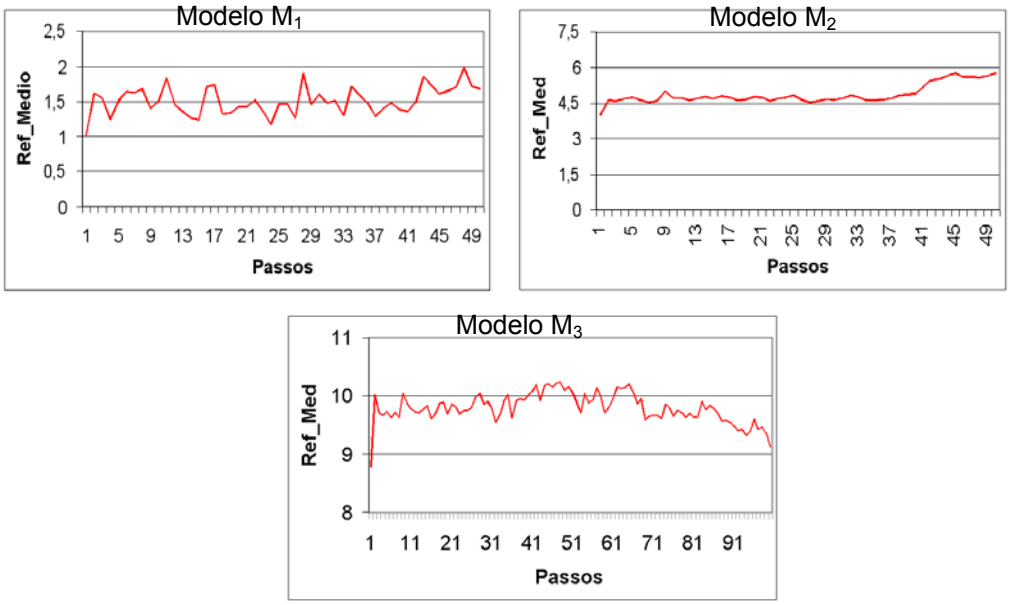

Figura 5. Gráficos dos Reforços Médios dos Modelos M_1, M_2 e M_3.

A Figura 6 demonstra o mesmo conjunto de simulação usado na Figura 5, porém o gráfico mostra os resultados das médias dos valores de utilidade Q(s,a).Com base nesses valores, o algoritmo consegue definir (através da maximização sobre Q(s,a)) a ação de maior valor de utilidade no estado s. Ao longo do tempo ocorreu convergência para uma política ótima de ações (μ) em cada modelo simulado.

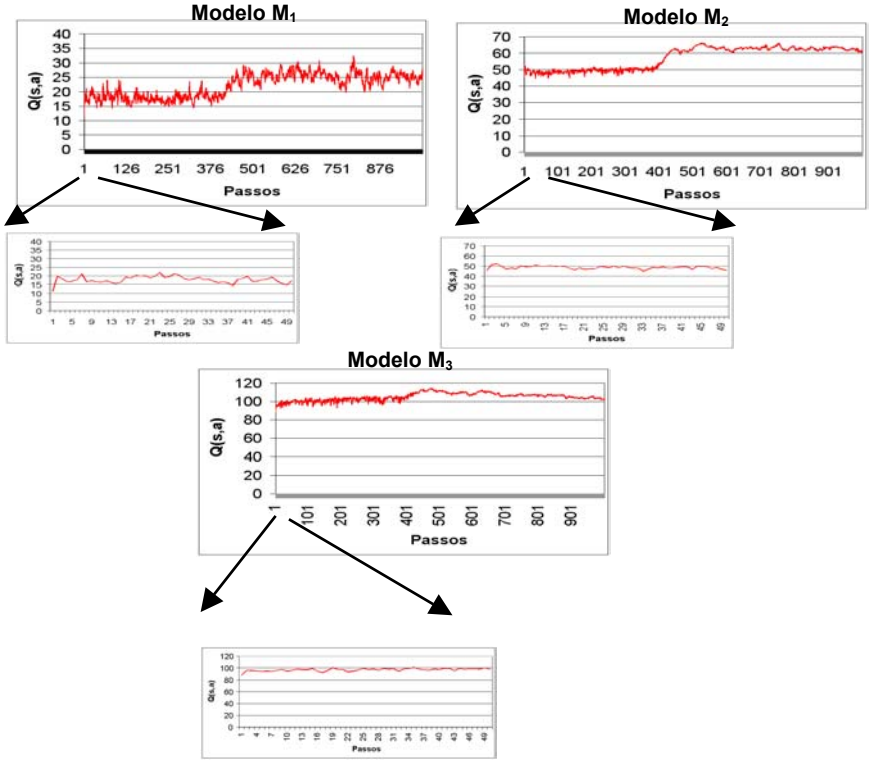

Figura 6. Gráficos dos Q(s, a) dos Modelos M_1, M_2 e M_3.

Na Figura 7, foi usado o mesmo conjunto de simulação da Figura 5 e 6. Os gráficos obtidos representam as médias das transições entre os estados cognitivos dos modelos simulados. Note que o STI não conhece os modelos: os resultados da Figura 7 são conseqüência da convergência do Q-Learning para uma política ótima de ações (μ*).

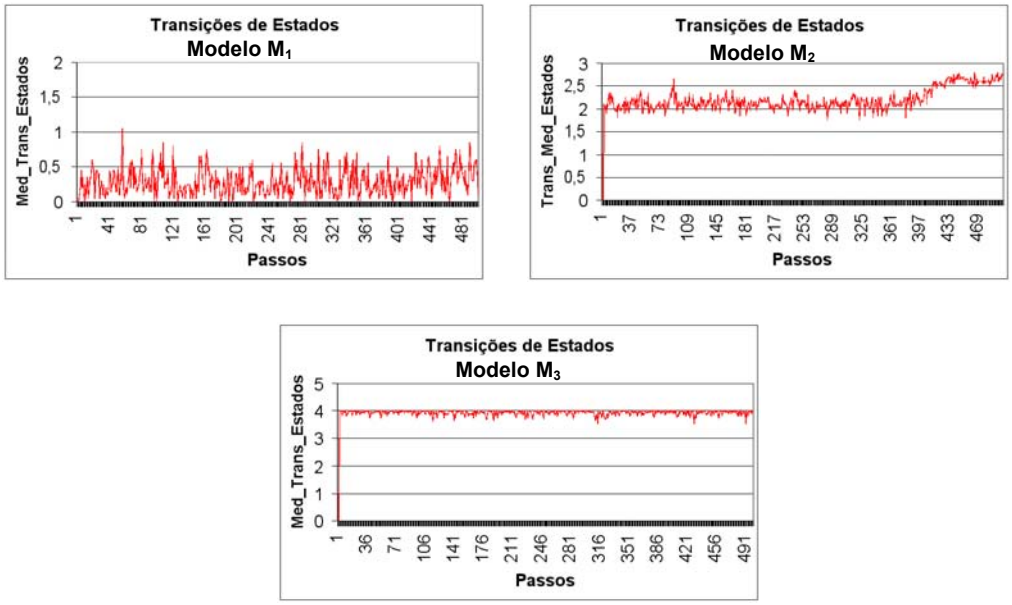

Figura 7. Gráficos da Média das Transições dos Estado dos Modelos M_1, M_2 e M_3.

Na Figura 8, observa-se que o algoritmo Q-Learning consegue através da política ótima de ações (μ), usando os modelos M_1, M_2 e M_3 (que representam os modelos aos quais se esperava obter com as simulações) visitar mais os estados E_0 (perfazendo um total de 86% das visitas), E_2 (perfazendo um total de 81% das visitas) e E_4 (perfazendo um total de 93% das visitas).

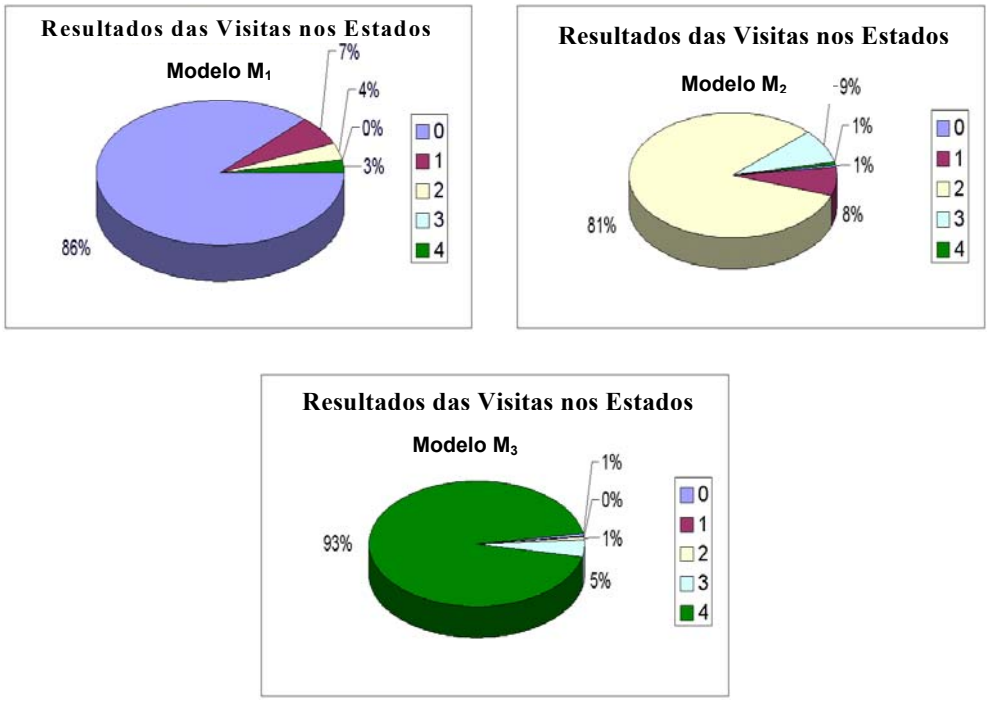

Figura 8. Gráficos das Visitas nos Estados para Modelo M_1 M_2 e M_3.

5. CONCLUSÃO

5.1. Caracterizando o Sistema com uma análise das Vantagens e Desvantagens.

A utilização do Sistema de Aprendizado por Reforço para Modelagem Autônoma do Aprendiz em um Tutor Inteligente pode oferecer vantagens e possíveis dificuldades que se tornariam desvantagens, dentre as quais:

Vantagens:

- Para implementação, não precisa do modelo do aprendiz;
- O sistema independe do conteúdo apresentado pelo tutor;
- Adaptação do sistema a várias estratégias pedagógicas;
- Garantia - através de um log de registro - da continuidade da tutoria do aprendiz a partir do ponto em que parou.

Desvantagens:

- Um número elevado de ações a serem tomadas em determinado estado cognitivo do estudante.
- Dependendo do estado podem existir diversas estratégicas pedagógicas a serem adotadas.
- Interfaceamento com tutores já existentes, devido à necessidade dos tutores receberem os parâmetros do sistema AR.
- Lentidão de convergência do algoritmo, que pode ser parcialmente compensada pelo uso de variantes baseadas em aproximações compactas [Tsitsiklis, J. & Bertsekas, D. 1996], estratégias de generalização da experiência [Ribeiro, C., 1998], ou planos de ação obtidos em simulação [Sutton & Barto, 1998].

5.2. Proposta para Trabalhos Futuros

Uma contribuição deste trabalho foi o resultado positivo obtido na simulação com o uso do algoritmo Q-Learning aplicado a um agente tutor que inclui na sua arquitetura o módulo diagnóstico, para que desta forma se consiga dar um passo importante na questão da modelagem autônoma do aprendiz em um tutor inteligente. Acredita-se que este trabalho venha contribuir para formalização computacional do problema de modelagem do estado cognitivo do aprendiz de forma dinâmica. Sendo assim, os resultados viabilizam a utilização de tutores inteligentes que utilizam o Sistema de AR em ambientes on-line, como por exemplo Internet.

Outra contribuição deste trabalho é, deixar dados para estudos de viabilidade de desempenho no que se refere a redução do tempo de convergência do algoritmo, para uso mais efetivo de tutores inteligentes com o Sistema de AR .

Além das contribuições citadas acima, podemos evidenciar possibilidades de trabalhos futuros, como:

- O interfaceamento de tutores inteligentes já existentes no mercado com o Sistema de AR.
- Desenvolvimento de um sistema de Tutoria Inteligente, onde se possa utilizar múltiplas estratégias, independente do domínio.
- Com a estrutura desenvolvida neste trabalho, adaptá-la para o uso de multiagentes.

REFERÊNCIAS BIBLIOGRÁFICAS

Burns,H and Parlett,J.W. and Redfield,C.L.: "Intelligent Tutoring Systems" N.J.: Lawrence Erlbaum-Verlag 1991.Bellman, R. E. Dynamic Programming. Princeton: Princeton University Press, 1957.

Giraffa, L.M.M.: "Uma arquitetura de tutor utilizando estados mentais", Rio Grande do Sul: PUC-RS,1999. Tese de Doutorado.

Hendley, R. and Jurascheck , N. "Cascade: Introducing AI intoCBT.Computers Education", London, v.18, n.1-3, p.71-76, 1992

Leite, A. S. "Um Modelo de Sistema Educativo Cognitivista Baseado em Tutoria Inteligente Adaptativa via Aderência Conceitual", São José dos Campos: ITA, 1999
Tese de Doutorado.

Marietto, M. G.:"Definição Dinâmica de Estratégias Instrucionais em sistema de tutoria Inteligente: Um Abordagem Multiagentes na WWW", São Jose dos Campos:ITA, 2000
Tese de Doutorado.

Sutton, R.and Barto, A. Reinforcement Learning: An Introduction. MIT Press, 1998.

Ribeiro, C.H.C. "Embedding a priori Knowledge in Reinforcement Learning", Journal of Intelligent and

Robotic Systems. Dordrecht, Holanda: , v.2, n.1, p.51 - 71, 1998

Viccari, R.M.: "Um Tutor Inteligente para a Programação em Lógica – Idealização,Projeto e Desenvolvimento",Portugal Universidade de Coimbra,1990,Tese de Doutorado.

Tsitsiklis, J. e Bertsekas, D. (1996). *Neuro-Dynamic Programming*. Athena Scientific, 1996.

Watkins, C. J. C. H.:" Learning from Delayed Rewards.", PhD thesis, University of Cambridge,1989.

Watkins, C. J.C.H., and Dayan, P. "Q-leaning". Machine Learning 8(3/4):279-292, 1992.