Um Algoritmo para Extração de um Plano BDI que Obedece uma Política MDP Ótima

Diego R. Pereira, Graçaliz P. Dimuro

[1]Mestrado em Ciência da Computação
Programa de Pós-Graduação em Informática – Universidade Católica de Pelotas (UCPEL)
Rua Félix da Cunha, 412 – 96.010-000 – Pelotas – RS – Brasil

{dpereira,liz}@ucpel.tche.br

Abstract. *This paper presents an analisys of the hybrid approach BDI-MDP found in the literature, and introduces an algorithm that gets plans for BDI agents that obey an optimal policy, presenting an application example. Policies are mapped into BDI agents plans, following an intention, considering that plans extracted from an optimal policy are the ones that the BDI agent selects the plan with the greatest utility, and has an optimal strategy reconsideration. We also present an implementation, in a form of a search algorithm that follows a given policy through the state space.*

Resumo. *Neste artigo, apresenta-se uma análise da proposta de trabalhar com uma abordagem híbrida BDI-MDP encontrada na literatura, e introduz-se um algoritmo que obtém planos para agentes BDI que obedecem uma política ótima, apresentando um exemplo ilustrativo de sua aplicação. Políticas são mapeadas para planos de agentes BDI, de acordo com uma dada intenção, considerando que os planos derivados de uma política ótima são aqueles adotados pelo agente BDI que seleciona o plano com a maior utilidade, e uma reconsideração de estratégia ótima. Apresenta-se também uma versão computacional deste mapeamento, na forma de um algoritmo de busca que segue uma dada política através do espaço de estados.*

1. Introdução

Diversas linhas de pesquisa da Inteligência Artificial, que atuam na área de multiagentes, avançaram na construção de times de agentes para atuar em ambientes complexos e dinâmicos. Especificamente, duas abordagens têm obtido grandes resultados. Primeiro temos os Problemas de Decisão de Markov Parcialmente Observáveis (Partially Observable Markov Decision Problems - POMDPs) que se concentram na coordenação de times e na existência de incerteza de ações e observações em domínios do mundo real [Boutilier et al. 1999]. E temos também arquitetura de agentes BDI (Beliefs, Desires and Intentions - Crenças, Desejos e Intenções), inspirados pela lógica e pela psicologia, que são abordagens simbólicas que definitivamente permitem melhor entendimento da metodologia aplicada [Rao and Georgeff 1992].

Apesar de terem sido alcançados excelentes avanços em cada uma dessas áreas, existe uma grande falta de modelos híbridos que permitam interações entre os modelos descritos acima, permitindo um auxiliar o outro em seus pontos fracos. Por exemplo,

as abordagens atuais BDI não possuem ferramentas para análise quantitativa de performance sobre ambientes com incerteza. Já os POMDPs executam bem tal tarefa, mas a complexidade de se encontrar uma política ótima, em modelos onde o espaço de estados é muito grande, é quase intratável. Felizmente, abordagens híbridas BDI-POMDP prometem uma solução para este problema, onde os planos BDI são explorados para melhorar a tratabilidade dos POMDPs, e POMDPs melhoram a performance dos planos dos agentes BDI [Simari and Parsons 2006, Paruchuri et al. 2006].

O objetivo deste artigo é apresentar um algoritmo e um exemplo concreto que demonstre a relação entre MDPs e arquiteturas BDI, onde planos de agentes BDI podem ser extraídos através de políticas de MDP de tal forma que tais planos "obedecem" essas políticas. Neste trabalho preliminar, no sentido de simplificar o exemplo apresentado, não estarão sendo considerados modelos parcialmente observáveis. Entretanto, existem vários trabalhos que apontam que modelos híbridos BDI-POMDP podem de fato trazer várias vantagens, como, por exemplo, a melhora de performance [Paruchuri et al. 2006, Nair and Tambe 2005, Gupta et al. 2007].

Este artigo está organizado da seguinte forma: na Seção 2 estabelecemos uma relação entre a arquitetura BDI e os modelos MDP; na Seção 3 a relação entre políticas e planos é discutida, com base nos trabalhos existentes sobre o tema; na Seção 4 vemos como extrair um plano a partir de uma política ótima; na Seção 5 vemos os algoritmos utilizados para a resolução do exemplo; na Seção 6 apresentamos um exemplo para estudo de caso; na Seção 7 temos uma discussão dos resultados e a conclusão.

2. O problema de relacionar MDPs e arquitetura BDI

Processos de Decisão de Markov (Markov Decision Processes - MDP) pode ser a abordagem ideal para a implementação de agentes inteligentes, devido o fato de poderem dar valores às utilidades de cada estado, e probabilidades de transição entre estes estados. Com estes valores podemos utilizar algoritmos como a interação de valor para obter uma política ótima, mapeando cada estado para a melhor ação para aquele estado. Mas a própria natureza destes algoritmos não consegue tratar espaço de estados muito grande, devido à sua alta complexidade, obtendo assim uma solução apenas aproximada. Até mesmo as melhores abordagens disponíveis não são capazes de lidar com problemas muito complexos ou muito grandes.

Em contraste, a arquitetura BDI (beliefs, desires, intentions) tem agentes construídos com um conjunto de crenças sobre o estado do mundo, e um conjunto de desejos que, de maneira geral, identificam quais estados do mundo são objetivos para o agente. Através da deliberação o agente formula uma ou mais intenções. Então o agente constrói um plano para alcançar suas intenções.

Uma abordagem BDI para um problema pode ter um desempenho bem inferior a uma abordagem MDP, desde que o problema seja tratável pelo mesmo [Simari and Parsons 2006]. Entretanto, um modelo BDI pode solucionar problemas que estão além do escopo dos modelos MDP, e pode ainda ser mais eficiente do que modelos MDP aproximados, para problemas relativamente pequenos.

Em [Simari and Parsons 2006], encontra-se uma discussão sobre a relação formal entre as duas abordagens, em particular as seguintes questões:

- Dada uma política que é uma solução para um MDP, como extrair uma descrição BDI que pode ser usada para aproximar esta solução?
- Dada uma descrição completa BDI, como obter uma política que um agente baseado em MDP possa usar para controlar suas ações?

O interesse particular deste trabalho está nas respostas para a primeira questão. Pretende-se, ao desenvolver um mecanismo para resolver esta questão, possibilitar sua aplicação a diferentes tipos de processos de decisão, como, por exemplo, os Processos de Decisão Fracamente acoplados [Parr 1998, Meuleau et al. 1998], que é o nosso principal tema de trabalho.

Primeiramente descreveremos como ações, estados e funções de transição para um BDI e MDP podem ser relacionados. Ambas as descrições consistem de um espaço de estados S, um conjunto de ações A, e uma função de transição T que depende do estado corrente e da ação a ser realizada. Uma descrição MDP ainda inclui:

- Uma função de recompensa R,
- Uma distribuição de probabilidades P sobre o conjunto de estados, e
- Um conjunto de políticas Π, em que cada membro do conjunto identifica a melhor ação a ser tomada em cada estado.

Denotamos um MDP pela tupla:

$$< S, A, T, R, P, \Pi >$$

Um agente que usa a descrição MDP para decidir como agir será chamado de agente MDP.

Uma descrição BDI consiste, em adição ao espaço de estados, ações e função de transição, de:

- Um conjunto de crenças B, um de desejos D, e um de intenções I;
- Um componente de deliberação Del; e
- Um componente de raciocínio meio-fim M

Denotamos um BDI pela tupla:[1]

$$< S, A, T, B, D, I, Del, M >$$

Um agente que usa a descrição BDI para decidir como agir será chamado de agente BDI.

Para um agente em um dado ambiente consideramos que S, A e T são os mesmos para ambas as descrições (BDI e MDP). Além disso consideraremos que B e P representam a mesma idéia - identificam qual o estado o agente está atualmente. Com estas equivalências definidas, temos de um lado recompensas e políticas e por outro temos desejos, deliberação, raciocínio meio-fim, e intenções. Na verdade, como recompensas são meios de se determinar políticas, e desejos são um passo para se determinar intenções, estes componentes podem ser ignorados. Finalmente a relação que devemos considerar com detalhes seriam políticas e intenções.

[1]Embora a notação adotada não seja a padrão, optou-se por utilizá-la para possibilitar a discussão da relação entre agentes BDI e agentes MDP, conforme apresentado em [Simari and Parsons 2006].

3. Intenções, planos e políticas

A semântica da intenção de um agente normalmente varia de acordo com a literatura, os significados mais comuns são: "o foco atual do agente", um plano que o agente toma para alcançar um determinado estado. Neste artigo, intenção é o estado que o agente se comprometeu a alcançar, e usaremos o termo plano-intenção (i-plan), para denotar uma seqüência de ações construídas para alcançar um determinado estado, ou seja para alcançar uma determinada intenção. O i-plan vai depender do estado atual do agente e de sua intenção. Um agente pode criar diferentes i-plans para alcançar a mesma intenção, dependendo de seu estado inicial, da mesma forma criará diferentes i-plans a partir do mesmo estado para alcançar diferentes intenções.

Os i-plans são a chave para se construir agentes BDI. Na prática um agente escolhe uma intenção, pesquisa entre uma série de planos pré-compilados, e determina o melhor para alcançar a sua intenção. Um bom sistema BDI consiste em uma boa biblioteca de planos construídos para operar em determinado ambiente no qual o agente opera.

Para analisar a questão de nosso interesse, discutiremos a proposta apresentada em [Simari and Parsons 2006] para obter i-plans a partir de políticas. Pretende-se obter i-plans através da solução de um MDP.

Um i-plan será denotado como ψ , indexado por $\psi^{i,s}$, onde i é a intenção do agente e s seu estado atual. I-plans são seqüências de ações e ψ_i denota a iésima ação em ψ, enquanto s_i^{ψ} denota o iésimo estado que o agente planeja visitar enquanto executa ψ . Portanto, para um i-plan ψ de tamanho p, o agente começa no estado s_0^{ψ} e planeja visitar os estados $s_1^{\psi}, s_2^{\psi}, \ldots, s_p^{\psi}$. O agente pode se desviar desta seqüência de estados através de ações não-determinísticas ou mudanças no ambiente. Quando isto ocorre, o i-plan não tem o efeito desejado, mas ainda pode levar o agente para sua intenção. Quando o desvio ocorre, o agente decide se precisa de uma nova intenção ou um novo i-plan. Este processo é chamado de reconsideração de intenção, que definirá se o agente precisa de uma nova intenção ou não. Se uma nova intenção for adotada o agente também irá precisar de um novo i-plan.

Deliberação, reconsideração e geração de i-plans podem ser feitos em teoria. A utilidade esperada de um i-plan pode ser obtida da mesma forma que uma política eu um MDP, estabelecendo-se um valor para cada ação em cada estado em que é executada. A diferença é que em um i-plan consideramos apenas uma seqüência no espaço de estados, e na avaliação de uma política consideramos as ações em todos os estados. A princípio o agente BDI terá a mesma abordagem para atravessar o espaço de estados. Selecionará uma intenção, identificará um i-plan para alcançar sua intenção e executará seu i-plan até perceber que seu i-plan não irá alcançar sua intenção ou que sua intenção não pode ser alcançada (ou não é a melhor intenção possível). Neste momento o agente irá gerar um novo i-plan ou escolher uma nova intenção e gerar um i-plan para alcançá-lo e o processo irá se repetir.

Compara-se este processo com um agente MDP no mesmo espaço de estados, no qual o agente sempre sabe, através de sua política, qual a melhor ação a tomar. A desvantagem é que calcular uma política ótima é mais custoso do que criar um plano simples. O preço pago pelo agente BDI é que este tem que computar o que fazer online enquanto o calculo de política do agente MDP pode ser feito offline e ainda seu plano

pode ser sub-ótimo pois pode haver desvio em suas ações.

Apesar de suas diferenças, podemos estabelecer diversas equivalências entre os dois considerando como eles trabalham no mesmo espaço de estados. Dada uma política, que tem uma ação para cada estado podemos derivar um ou mais i-plans, determinando uma trajetória através do espaço de estados. Portanto uma política incorpora um conjunto de i-plans. Por outro lado, consideraremos um conjunto de i-plans cada um com uma trajetória através de um conjunto de estados. Este conjunto de i-plans identificam que ação tomar em todos estados contidos na trajetória, e adicionarmos nenhuma ação aos estados que não estiverem na trajetória, teremos então uma política (não muito boa). Entretanto, esta última equivalência não será explorada neste trabalho, pois foge ao nosso interesse imediato.

4. Políticas para I-Plans

Assumiremos que temos uma política π que é a solução para um MDP completamente especificado. Também assumimos que π é ótima, pois os resultados obtidos dependem disto. Entretanto de qualquer π é possível extrair valores de utilidade para os estados que irão induzir π, e estes poderão ser usados para estabelecer i-plans.

Nesta seção consideraremos que π foi obtido através da convergência de um algoritmo específico que dá um valor a cada par estado/ação. Agentes BDI também podem mapear estados e ações em valores. Estes valores são computados atribuindo-se um valor a cada i-plan. Seja ψ um i-plan de tamanho p, e ψ_i a iésima ação envolvida em ψ. Uma maneira de atribuir valor a ψ é

$$V(\psi) = \sum_{i=1}^{p} \frac{R(s_{i-1}^{\psi}, \psi_i)}{i}$$

onde $V(\psi)$ é o valor do i-plan ψ, s_0^{ψ} é o estado inicial de ψ, s_{i+1}^{ψ} é o estado para qual o agente espera chegar depois de executar a ação ψ_i, e $R(s_{i-1}^{\psi}, \psi_i)$ é a recompensa por executar a ação ψ_i no estado s_{i-1}^{ψ}. Portanto o valor atribuído para um i-plan é a soma das recompensas que serão alcançadas se todas as ações tiverem o efeito desejado; a divisão da recompensa por i denota o desconto pelo tempo, pois recompensas ganhas mais cedo são mais valiosas do que aquelas ganhas no futuro.

É importante notar que esta é apenas uma entre várias maneiras de se atribuir valores a um i-plan. De maneira geral, apenas precisamos que um i-plan ψ tenha cada ação adicionada ao plano valores de custo não negativos, e que estes valores dependam das recompensas dos estados que o agente planeja visitar. Por exemplo, em um ambiente simples a função de recompensa poderia ser definida por:

$$R(s_{i-1}^{\psi}, \psi_i) = \begin{cases} 1 & \text{se } \psi \text{ leva diretamente ao objetivo} \\ 0 & \text{caso contrario} \end{cases}$$

e $R(s, a) = 0$ para cada outro par estado/ação.

A seguir apresenta-se a definição formal do conceito de i-plan, bem como o que significa um i-plan obedecer a uma dada política.

Definição 4.1 *Uma seqüencia de ações $\psi_0, \psi_1, \ldots, \psi_p$, é chamada um i-plan se o ψ_i's $(0 \leq i \leq p)$ forem selecionadas com o objetivo de executá-los um por vez em ordem para alcançar um dado objetivo.*

Definição 4.2 *Um i-plan ψ de tamanho p obedece a uma política π se, e somente se, $\forall i, 1 \leq i \leq p, \pi(s^{\psi}_{i-1}) = \psi_i$, onde s^{ψ}_i é o estado para qual o agente está planejando chegar após executar a ação ψ_{i-1}, e s^{ψ}_0 é seu estado inicial*

A definição 4.2 simplesmente especifica que um i-plan obedece a uma política se, e somente se, as ações prescritas pelo i-plan são as mesmas prescritas pela política através dos estados intermediários do i-plan. Lembre-se que assumimos que os i-plans são lineares, que nenhuma consideração é feita devido a resultados inesperados de suas ações.

A definição de obediência deve ser seguida pela definição de conformância:

Definição 4.3 *Uma política π está de acordo a um i-plan ψ de tamanho p se, e somente se, $\forall i, 1 \leq i \leq p, \pi(s^{\psi}_{i-1}) = \psi_i$, onde s^{ψ}_i é o estado que resulta da execução da ação ψ_{i-1} no estado s^{ψ}_{i-1}, e s^{ψ}_0 é o estado no qual a primeira ação é executada.*

Desde que um i-plan seja indexado pela intenção que irá alcançar quando for executada, podemos estender a noção de obediência e conformidade para intenções. Uma política π está de acordo com uma intenção i se para todos os i-plans $\psi^{i,s}$, π conforma para $\psi^{i,s}$, e uma intenção i obedece uma política π se todos os i-plans $\psi^{i,s}$ obedecem à π

Com estes conceitos podemos propor a seguinte afirmação:

Hipótese 4.4 *Dado um agente BDI e um agente MDP com uma política ótima π, se o agente BDI está no estado s_i, então o i-plan ψ com o maior valor de utilidade será aquele que ψ obedece a π, começando em s_i*

Em geral a afirmação somente se sustentará se estados com a mesma recompensa são considerados os mesmos nas duas abordagens, BDI e MDP. De outra forma, mesmo que as utilidades sejam equivalentes, as ações podem não ser exatamente as mesmas pois a ordem em que os estados com a mesma recompensa são considerados, podem afetar a seleção de ações. A prova desta afirmação pode ser feita para cenários progressivamente mais complexos.

Os resultados iniciais encontrados em [Simari and Parsons 2006] para relacionar políticas com intenções e i-plans são estabelecidos para casos determinísticos e totalmente acessíveis, como a seguir:

Proposição 4.5 *Sejam $< S, A, T, R, P, \Pi >$ um agente MDP e $\pi \in \Pi$ uma política que é ótima sob um critério de máxima utilidade esperada. Seja $< S, A, T, B, D, I, Del, M >$ um agente BDI. Considere que Del sempre seleciona a intenção com a maior utilidade e M seleciona o i-plan com a maior valor. Se o ambiente for totalmente observável e determinístico, então $\forall s \in S, \forall a \in A, |T(s, a)| = 1$, então $\forall i \in I, \forall s \in S$, é válido que $\psi^{i,s}$ obedece π.*

Prova. *O resultado deriva diretamente do fato que assumimos que Del é ótimo e que as ações são completamente determinísticas no ambiente. Como π é ótima sob um critério de máxima utilidade esperada, esta sempre irá selecionar ações que levam o*

agente para o melhor estado-objetivo da melhor forma possível. Como Del escolhe a intenção com a maior utilidade, e suas ações são determinísticas, irá selecionar a mesma intenção/objetivo que π leva seu agente a escolher. De maneira similar, como M escolhe o i-plan com as maiores recompensas, este irá escolher um i-plan que percorre a mesma trajetória através do espaço de estados (a partir de qualquer estado que o agente se encontre) que π. Portanto, se assumirmos que estados com recompensas iguais são considerados na mesma ordem por π e M, está claro que i-plans gerados por M irão obedecer π. □

Se as ações não forem determinísticas, a utilidade dos i-plans não estará claramente definida. Ao invés de uma simples soma das recompensas através do caminho do plano, a falha nas ações deve ser considerada. Portanto devemos assumir que os componentes de deliberação e o raciocínio meio-fim são ótimos sob um critério de máxima utilidade esperada ao contrário de serem capazes de escolher a intenção e o i-plan (respectivamente) com as maiores recompensas.

Proposição 4.6 *Sejam $< S, A, T, R, P, \Pi >$ um agente MDP e $\pi \in \Pi$ uma política que é ótima sob um critério de máxima utilidade esperada. Seja $< S, A, T, B, D, I, Del, M >$ um agente BDI, onde M e Del sejam ótimos sob um critério de máxima utilidade esperada. Se o ambiente for totalmente observável e não-determinístico, então $|T(s, a)| \geq 1$, então $\forall i \in I, \forall s \in S$, é válido que $\psi^{i,s}$ obedece π.*

Prova. *Pelo fato de assumirmos que Del está escolhendo intenções que são ótimas sob um critério de utilidade máxima esperada e M está construindo i-plans que são ótimos sob um critério de utilidade máxima esperada, o mesmo argumento da 4.5 nos diz que todo o i-plan irá obedecer π inclusive se suas ações não forem determinísticas.* □

Estes resultados [Simari and Parsons 2006] mostram porque a abordagem BDI tem dificuldades para gerar um comportamento ótimo. Nos modelos BDI clássicos a deliberação seleciona uma intenção e então a análise meio-fim constrói um plano para alcançar tal intenção. Para poder escolher um conjunto de ações ótimo (ver 4.6), o componente de deliberação deve poder escolher a intenção que é ótima sob um critério de utilidade máxima esperada antes da análise de meio-fim escolha um i-plan.

Analisaremos o caso em que o ambiente não é totalmente observável, em outras palavras, o agente não sabe em qual estado S ele está, e deve confiar em suas estimativas do estado atual do ambiente [Lovejoy 1991]. Modelos MDP (tecnicamente POMDP - Partially Observable Markov Decision Process sob tais condições) tratam este tipo de situação estendendo o conceito de estado. Ao invés de lidar com um estado que é $s \in S$, o espaço de estados descrevendo todos os estados no ambiente, um estado se torna uma probabilidade de distribuição sobre todos s. Se tentarmos enumerar todas as possíveis distribuições de s'_i, então podemos pensar em políticas e i-plans que se preocupam com um novo espaço de estados $S' = \bigcup_i s'_i$. Se chamarmos S' a contraparte parcialmente observável de S e considerarmos este o novo espaço de estados onde os agentes BDI e MDP irão operar, então ambos B e P irão identificar algum $s' \in S'$ como estado atual do agente. Podemos então estender a Proposição 4.6 como [Simari and Parsons 2006]:

Proposição 4.7 *Sejam $< S, A, T, R, P, \Pi >$ um agente MDP e $\pi \in \Pi$ uma política que é ótima sob um critério de máxima utilidade esperada. Seja $< S, A, T, B, D, I, Del, M >$ um agente BDI, onde M e Del sejam ótimos sob um critério de máxima utilidade esperada. Se o ambiente for parcialmente observável, com S' sendo a contraparte de S, e*

não determinística, então que $|T(s,a)| \geq 1$*, então* $\forall i \in I, \forall s' \in S'$*, é válido que* $\psi^{i,s'}$ *obedece* π.

Prova. *Este resultado [Simari and Parsons 2006] generaliza o resultado anterior, porque agora estamos considerando um ambiente parcialmente observável. Entretanto, devido ao fato de expandirmos* S *para* S' *em ambos MDP e BDI modelos, o resultado deriva diretamente da 4.6.* □

Esta seria uma formalização da relação entre políticas e i-plans, similar àquela discutida na Seção 2. Esta correspondência é válida sob restritivas suposições, em particular os requerimentos ótimos (política, M e Del), mas garante que os i-plans gerados refletirão uma política ótima. Se quisermos relaxar a necessidade de um i-plan corresponder a uma política ótima, poderemos criar i-plans com menos restrições.

Apesar destes resultados formais apresentados em [Simari and Parsons 2006] e em outros trabalhos (como, por exemplo, em [Paruchuri et al. 2006, Nair and Tambe 2005, Gupta et al. 2007]) nos indicarem que podemos extrair i-plans que obedecem a uma política, eles não nos dizem como fazê-lo.

Observa-se que podemos obter um i-plan a partir de qualquer política através de uma simples procura através do espaço de estados, seguindo uma política π até um máximo local ser atingido. Este estado é então selecionado como intenção. Para selecionarmos uma única intenção, assumimos que as ações do agente sempre têm o resultado mais provável; de outra forma, teríamos uma árvore como resultado e não um caminho simples.

Seguindo a política desta forma podemos obter quantos i-plans forem necessários: depois que alcançar o estado intenção, simplesmente continuamos seguindo a política a partir do estado alcançado após a última intenção.

O processo que descrevemos irá construir um conjunto de i-plans que obedecem uma política arbitrária. Tal política não é necessariamente ótima, portanto nada pode se garantir sobre o resultado, bem como sobre os i-plans estabelecidos, apenas que os membros do conjunto de i-plans obedecerão à política.

5. Algoritmos para a obtenção de planos que obedecem uma política ótima

Nesta seção, introduziremos o algoritmo **Algoritmo De_Política_para_Planos**, que gera um i-plan que obedece uma política ótima de um MDP, apresentado como um pseudocódigo de C (Fig. 3). Este algoritmo utiliza o algoritmo de iteração de valor (Fig. 1) e o de iteração de política (Fig. 2), extraídos de [Russell and Norvig 1995]. O algoritmo de Iteração de Valor retorna a função de utilidade, que para cada estado fornece a sua utilidade. O algoritmo de Iteração de Política determina, de acordo com a função de utilidade, a política ótima.

No algoritmo **Algoritmo De_Política_para_Planos**, `s` irá receber o estado inicial do MDP e `g` irá receber o estado intenção, a partir de uma busca no espaço de estados, pelo estado de máxima utilidade esperada em `U`. No laço, enquanto o estado atual não for o estado de intenção, seleciona-se a ação indicada pela política para o estado atual `s(a)`, adicionando-se esta ao plano. Após, atualizamos o estado atual a partir da ação indicada pela política. Assumimos que a ação tem seu resultado mais provável, para fins de obtenção do próximo estado.

função ITERAÇÃO-DE-VALOR(pdm, ϵ) retorna uma função de utilidade

entradas: pdm, um PDM com estados S, modelo de transição T, função de recompensa R, desconto γ , ϵ o erro máximo permitido na utilidade de qualquer estado

variáveis locais: U, U', vetores de utilidades para estados em S, inicialmente zero δ , a mudança máxima na utilidade de qualquer estado e uma iteração

repita
$U \leftarrow U'; \delta\gamma 0$
para cada estado s **em** S **faça**
    $U'[s] \leftarrow R[s] + \gamma max_a \sum_{s'} T(s, a, s')U[s]$
    se $|U'[s] - U[s]| > \delta$ **então** $|U'[s] - U[s]$
até $\delta < \epsilon(1 - \gamma)/\gamma$
retornar U

Figura 1. Algoritmo de Iteração de Valor

6. Um exemplo de aplicação

Nesta seção, introduziremos um exemplo simples para que o funcionamento do algoritmo De_Política_para_Planos possa ser visualizado de forma clara. O exemplo escolhido, extraído de [Russell and Norvig 1995], é completamente observável. Deixamos, para trabalhos futuros, a análise de ambientes parcialmente observáveis.

Suponha que o agente esteja no ambiente da Figura 4 composto por uma matriz de 3×4. Começando pelo estado $(1, 1)$, ele deve escolher uma ação por período de tempo. A interação com ambiente termina quando o agente atinge um dos estados finais, $(2, 4)$ ou $(3, 4)$ com recompensas -1 e $+1$ respectivamente. Nos outros estados a recompensa é de $-0, 04$. Em cada estado, as ações disponíveis são *Acima, Abaixo, Esquerda e Direita*. O ambiente é completamente observável, portanto o agente sempre sabe em que estado se encontra.

Apesar do ambiente ser completamente observável, suas ações não são determinísticas. O modelo usado para movimento do agente está ilustrado na Figura 5. Cada ação alcança seu objetivo com uma probabilidade de 80% ou $0, 8$, mas no restante do tempo, a ação move o agente em ângulos retos até a direção pretendida. Além disso, se o agente bater em uma parede, ele permanecerá no mesmo estado. Por exemplo, a partir do estado inicial $(1, 1)$, a ação *Acima* move o agente para $(2, 1)$ com probabilidade de $0, 8$ mas, com probabilidade de $0, 1$ ele se move para esquerda e vai para $(1, 2)$ e com probabilidade de $0, 1$ ele se move para direita se choca com a parede e permanece em $(1, 1)$

Para este exemplo, o algoritmo de Iteração_de_Valor forneceu as utilidades para os estados como mostrado na Figura 6. O algoritmo de Iteração_de_Política apresentou os resultados mostrados na Figura 7. Assim, o algoritmo De_Política_para_Planos identifica o estado de maior utilidade como intenção $(3, 4)$ construindo então o seguinte i-plan: `[Acima, Acima, Direita, Direita, Direita]`

função ITERAÇÃO-DE-POLÍTICA (pdm) **retorna** uma política

entradas: pdm, um PDM com estados S, modelo de transição T

variáveis locais: U, U' vetores de utilidades para estados em S, inicialmente zero π, um vetor de política indexado pelo estado, inicialmente aleatório

repita
$U \leftarrow$ AVALIAÇÃO-DE-POLÍTICA (π, U, pdm)
$inalterado? \leftarrow$ verdadeiro
para cada estado s **em** S **faça**
    se $max_a \sum_{i'} T(s,a,s')U[s'] > \sum_{i'} T(s,\pi[s,s'])U[s']$
    então $\pi[s] \leftarrow argmax_a \sum_{i'} T(s,a,s')U[s']$
    $inalterado? \leftarrow$ falso
Até inalterado?
retornar P

Figura 2. Algoritmo de Iteração de Política

```
policyToIplan(Policy pi, Util U, MDP m) {
    Iplan i;
    s = getCurrentState(m);
    g = getGoalState(s, U);
    while not s = g do {
        i = s(a);
        n = getNextState(s, Pi);
        s = n;
    }
    return i;
}
```

Figura 3. Algoritmo para obter um i-plan a partir de uma política ótima

7. Conclusão e considerações finais

Pode-se observar que alguns autores afirmam que o modelo BDI se estabeleceu, pelo menos em parte [Bratman et al. 1988], porque os modelos de decisão teóricos apresentavam-se computacionalmente intratáveis para serem usados na prática. Isto pode levar a uma percepção de que os dois modelos são de alguma forma colocados em oposição um ao outro, isto é, que um pesquisador adota o modelo BDI somente se ele acredita que modelos de decisão teóricos são errados ou impraticáveis, e que usar o modelo BDI leva de alguma forma a obter um desempenho menos que perfeito (porque, se constitui de uma abordagem heurística).

Anteriormente, foi mostrado [Simari and Parsons 2004] que numa tarefa em particular existem casos onde usar um modelo MDP resulta numa solução melhor, enquanto em outros casos uma abordagem BDI funciona melhor. Considerando que a tarefa cresce em tamanho além dos problemas que podem ser resolvidos de maneira ótima usando um

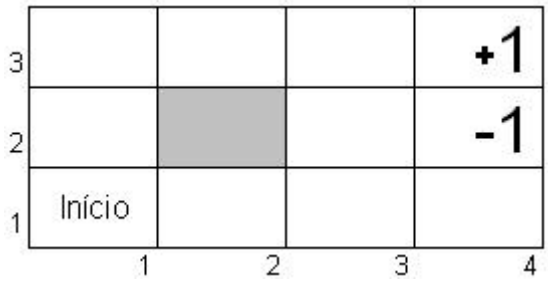

Figura 4. Um ambiente simples 3×4 que apresenta o agente com um problema de decisão seqüencial

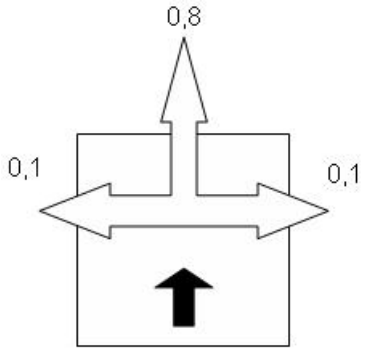

Figura 5. Ilustração do modelo de transição do ambiente

Figura 6. Utilidades dos estados

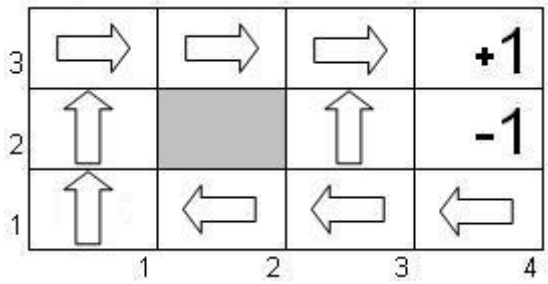

Figura 7. Política ótima obtida com o algoritmo

MDP, a abordagem BDI funciona melhor do que a melhor abordagem MDP. Isto sugere que alguns aspectos da abordagem BDI devem ser considerados com mais detalhe.

Os resultados apresentados em [Simari and Parsons 2006] (Seção 3) mostram que o modelo BDI não é inerentemente sub-ótimo. Se for possível, e os resultados em trabalhos anteriores mostram que isto pode acontecer em teoria, construir um conjunto de intenções que obedeçam a uma política ótima, então o modelo BDI nos dará uma performance ótima. Entretanto, isto pode não acontecer na prática, já que intenções não são construídas a partir de políticas ótimas, mas sim do conhecimento do domínio e, pela diferença entre os dois é que o modelo se torna sub-ótimo.

Neste artigo, analisamos a proposta de trabalhar com uma abordagem híbrida apresentada em [Simari and Parsons 2006], e introduzimos um algoritmo que obtém planos para agentes BDI a partir de uma política ótima, apresentando um exemplo ilustrativo de sua aplicação. Mapeamos políticas para i-plans dado que i-plans derivados de uma política ótima são aqueles adotados pelo agente BDI que seleciona o i-plan com a maior utilidade, e uma reconsideração de estratégia ótima; além disso, apresentamos uma versão computacional deste mapeamento, na forma de um algoritmo de busca baseado em seguir uma dada política através do espaço de estados.

Como trabalhos futuros, pretendemos avançar no estudo destes modelos híbridos, com o objetivo de desenvolver um algoritmo para obter i-plans a partir de políticas ótimas para modelos de Processos de Decisão Fracamente Acoplados [Parr 1998, Meuleau et al. 1998], para a aplicação no desenvolvimento de um sistema multiagente de um "chão de fábrica", com o problema de alocação de recursos, processos e produtos.

Agradecimentos

Este trabalho é parcialmente financiado pela Petrobrás (Projeto COPPETEC). Agradecemos aos revisores pelas sugestões recebidas.

Referências

Boutilier, C., Dean, T., and Hanks, S. (1999). Decision theoretic planning: Structural assumptions and computational leverage. *Journal Artificial Intelligence Res.*, 10.

Bratman, M. E., Israel, D. J., and Pollack, M. E. (1988). Plans and resource-bounded practical reasoning. *Computational Intelligence*, 4(4):349–355.

Gupta, T., Varakantham, P., Rauenbusch, T. W., and Tambe, M. (2007). Demonstration of teamwork in uncertain domains using hybrid bdi-pomdp systems. In *6th Intl. Conf. Autonomous Agents and Multi-Agent Systems, Demo Track*.

Lovejoy, W. S. (1991). A survey of algorithmic methods for partially observed Markov decision processes. *Annals of Operations Research*, 28(1–4):47–66.

Meuleau, N., Hauskrecht, M., Kim, K.-E., Peshkin, L., Kaelbling, L. P., Dean, T., and Boutilier, C. (1998). Solving very large weakly coupled markov decision processes. In *AAAI/IAAI*, pages 165–172.

Nair, R. and Tambe, M. (2005). Hybrid BDI-POMDP framework for multiagent teaming. *Journal of Artificial Intelligence Research*, 23:367–420.

Parr, R. (1998). Flexible decomposition algorithms for weakly coupled markov decision problems. In Cooper, G. F. and Moral, S., editors, *Proc. 14th Conf. Uncertainty in Artificial Intelligence, Madison*, pages 422–430. Morgan Kaufmann.

Paruchuri, P., Bowring, E., Nair, R., Pearce, J., Schurr, N., Tambe, M., and Varakantham, P. (2006). Multiagent teamwork: Hybrid approaches. In *Computer society of India Communications*. CSI. (Invited Talk, avaliable at http://teamcore.usc.edu/publications.htm).

Rao, A. S. and Georgeff, M. P. (1992). An abstract architecture for rational agents. In *KR*, pages 439–449.

Russell, S. J. and Norvig, P. (1995). Artificial inteligence : A modern approach.

Simari, G. I. and Parsons, S. (2004). On approximating the best decision for and autonomous agent. In *Proc. 6th Work. Game Theoretic Decision Agents*, pages 91–100.

Simari, G. I. and Parsons, S. (2006). On the relationship between MDPs and the BDI architecture. In Nakashima, H., Wellman, M. P., Weiss, G., and Stone, P., editors, *5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2006), Hakodate, Japan, May 8-12, 2006*, pages 1041–1048. ACM.