

Integrating the Organizational Model Moise^+ to a Cognitive Agent Architecture applied to Robocup Simulator 2D

Mateus Paiva Fogaça
Centro de Ciências Computacionais - C3
Universidade Federal do Rio Grande - FURG
Rio Grande – RS – Brazil
Email: mateus.p.fogaça@gmail.com

Eder Mateus Gonçalves
Centro de Ciências Computacionais - C3
Universidade Federal do Rio Grande - FURG
Rio Grande – RS – Brazil
Email: edergoncalves@furg.br

Resumo—This paper presents the initial results integrating the organizational model Moise^+ to a cognitive agent architecture once instantiated the UvA Trilearn agent applied to Robocup Simulator 2D. At this moment, the main goal is to equate the team performance using an explicit organization compared to the same team considering the social aspects specification in an implicit form. From this integration, it is expected a more simple way to input a social organization to the team development.

I. INTRODUÇÃO

Um sistema multiagente (SMA) é especificado e implementado a partir de três dimensões básicas: os agentes em si, os mecanismos de interação e comunicação entre os agentes e destes com o ambiente, e sua organização social. Destas dimensões, apenas os aspectos relacionados ao desenvolvimento interno dos agentes são naturalmente especificados de maneira explícita no projeto. No entanto, os aspectos relacionados a interação e comunicação dos agentes, bem como sua organização social, podem ser especificados de maneira implícita ao sistema ao qual farão parte.

Impor uma determinada organização a um grupo de agentes dar-se-á no sentido de estabelecer de forma explícita uma ou mais metas, sejam elas de contexto local aos agentes ou global ao sistema. Nesse caso, o papel da organização é permitir a um observador externo que se entenda para que propósito o sistema tende [1].

Em [2] é proposta uma arquitetura de agente que apresenta três níveis decisórios: nível reativo, nível instintivo e nível cognitivo. O nível reativo é composto por um conjunto de comportamentos de baixo nível que implementam as habilidades necessárias para que o agente atue no ambiente. O nível instintivo tem a função de identificar os estados do ambiente e definir qual o comportamento ativo no nível reativo. Além disso, estas informações de estado devem ser enviadas para o nível cognitivo que as utiliza, juntamente com a meta global do SMA, para determinar as metas individuais do agente. Estas metas individuais correspondem a estados desejados que devem ser alcançados pelos agentes. A escolha do melhor estado é feita a partir de um conjunto de funções preditivas que selecionam a melhor ação a ser tomada pelo agente. Esta seleção considera apenas o subconjunto de ações restringida pela modelo de organização.

A Robocup é uma entidade que visa estabelecer diretrizes para pesquisas nas diferentes áreas do conhecimento que viabilizam o desenvolvimento de artefatos robóticos em torno de problemas comuns. Um destes problemas padrão é a construção de equipes de futebol de robôs, reais e/ou virtuais. Entre as competições organizadas, tem-se aquelas voltadas para aspectos de software. No caso da simulação em 2D, tem-se uma excelente plataforma para desenvolvimentos envolvendo SMA's. A plataforma é denominada SoccerServer [3].

Este artigo apresenta a integração do modelo organizacional Moise^+ à arquitetura de agente cognitivo proposto em [2]. Esta arquitetura é implementada tendo como base o agente UvA Trilearn [4], especialmente em seus componentes reativos, e que possui uma estrutura base adequada aquela proposta em [2]. O FURGBol-Sim é resultado da integração destas abordagens. Neste trabalho, apresenta-se os resultados iniciais desse processo de integração, onde o objetivo é igualar a performance do FURGBol-Sim com organização social com do código base UvA Trilearn. Isto é feito extraindo a organização social implícita ao UvA Trilearn e aplicando ao FURGBol-Sim utilizando o Moise^+ .

Este artigo é estruturado da seguinte forma. A próxima seção descreve a arquitetura interna do agente utilizado no FURGBol-Sim, e a seção III descreve como dá-se o aproveitamento do código fonte do UvA Trilearn para esta implementação. A seção IV descreve como dá-se a extração da organização social implícita ao código do UvA Trilearn e sua especificação e inserção no código segundo o modelo Moise^+ . A seção V descreve os resultados do processo de integração e a seção VI apresenta as conclusões deste trabalho e as perspectivas de futuros trabalhos.

II. ARQUITETURA DE AGENTE AUTÔNOMO CONCORRENTE

Nesta seção busca-se apresentar as características da arquitetura interna dos agentes que compõem o FURGBol-Sim, independente do modelo de organização SMA adotado.

Um ambiente complexo, como o Soccer Server, exige soluções que agreguem aspectos de baixo nível, formado por comportamentos reativos que atendam restrições de tempo

real, bem como aspectos deliberativos que permitam tratar com questões de planejamento e atendimento de metas, por exemplo. Não obstante, o agente deve estar inserido em um contexto social que permita compartilhar seus planos e metas. Nesse sentido, propõe-se um modelo de agente cognitivo, denominado agente autônomo concorrente [2], que apresenta três níveis decisórios: reativo, instintivo e cognitivo como descrito na figura 1.

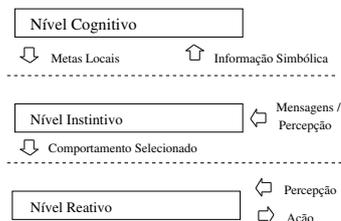


Figura 1. Arquitetura genérica de agente cognitivo

O *nível reativo* permite a interação do agente com o ambiente, tratando dos aspectos de percepção e ação. Deste modo, este nível é responsável pelo tratamento das restrições de tempo-real do agente. É formado por um conjunto de comportamentos reativos, que encapsulam as habilidades fundamentais do agente para atuar no ambiente. No caso do Soccer Server, citam-se como habilidades fundamentais, entre outras, passar, chutar, correr, marcar, interceptar a bola. Estes comportamentos podem ainda ser vistos como um nível de abstração hierárquico superior aos comandos básicos do ambiente.

O *nível instintivo* tem como funções básicas reconhecer o estado do jogo e executar a meta local do agente, que no âmbito do modelo *Moise+* corresponde a especificação de uma missão. A execução da meta local, ou missão, é dada pela seleção de uma seqüência de comportamentos reativos que levam o estado atual do jogo a um estado desejado. A meta local permanece válida enquanto determinados parâmetros associados ao estado atual do jogo são verificados. Quando o estado do jogo muda, seja para um estado desejado ou não, uma nova meta local deve ser selecionada.

O *nível cognitivo* tem a função de integrar o agente no SMA, coordenando objetivos coletivos com seqüências de ações individuais. Deste modo, o nível cognitivo transforma as metas globais em metas locais, por intermédio de uma instanciação da especificação funcional (FS) do modelo *Moise+*, segundo os papéis que possui e o grupo ao qual o agente pertence. Comparado aos níveis inferiores, o nível cognitivo possui restrições de tempo real mais brandas, o que permite a execução de tarefas mais complexas. A escolha da meta local dá-se por intermédio de um conjunto de funções preditivas do ambiente, que escolhem dentro do subconjunto de ações do ambiente, previamente restrita pelo modelo de organização, aquela que leva ao melhor estado futuro do ambiente, segundo as metas globais do SMA.

III. UVA TRILEARN

A arquitetura descrita na seção anterior é implementada a partir do código base do UvA Trilearn [4], que possui uma arquitetura de baixo nível, referente aos níveis hierárquicos mais próximos do ambiente, muito similar ao agente autônomo

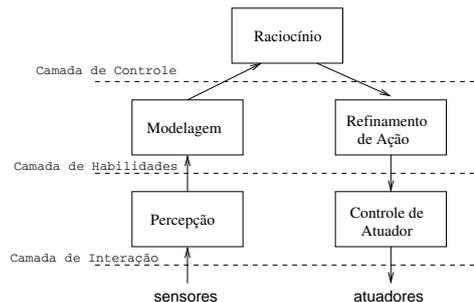


Figura 2. Arquitetura do Agente UvA Trilearn

concorrente. A arquitetura do UvA Trilearn é descrita pela figura 2.

A mais inferior é a *Camada de Interação*, que trata da interação do agente com o ambiente. A *Camada de Habilidades* usa a funcionalidade oferecida pela camada de interação para construir um modelo abstrato do mundo e para implementar os comportamentos dos agentes. A *Camada de Controle* contém os componentes deliberativos do agente. Sua função é escolher a melhor ação na camada de habilidades de acordo com o modelo de mundo e atual estratégia do agente.

O agente UvA Trilearn é implementado em C++ com suporte nativo em sistema operacional Linux, constituído por três threads: uma para percepção, uma para atuação, e outra para deliberação. A principal vantagem desta abordagem garantir um mínimo atraso em operações de entrada e saída com o servidor de simulação.

Para utilizar o UvA Trilearn como código fonte base, o objetivo é determinar uma estratégia de deliberação que garanta a escolha da melhor ação dado o modelo de mundo atual.

O mapeamento entre a arquitetura do agente autônomo concorrente e a implementação do UvA Trilearn é praticamente direta. A camada de interação mais o componente de refinamento de ação da camada de habilidades do UvA Trilearn correspondem ao nível reativo do agente autônomo concorrente. O componente de modelagem da camada de habilidades do UvA Trilearn implementa a função de identificação de estado do nível instintivo do agente autônomo concorrente. Finalmente, as funções restantes do nível instintivo mais o nível cognitivo do agente autônomo concorrente devem ser implementados na camada de controle do UvA Trilearn.

IV. MODELO DE ORGANIZAÇÃO MULTIAGENTE

O FURGBol-Sim é implementado segundo uma abordagem *top-down*, ou seja, da especificação social, que regula as relações entre os agentes que compõem a equipe, em direção as camadas mais baixas de implementação do agente, considerando que este possui uma arquitetura hierárquica que apresenta ambos aspectos, deliberativo e reativo. Deste modo, a descrição da equipe parte da especificação do SMA e de seu modelo de organização.

A idéia básica de um SMA é permitir que um grupo organizado de agentes cooperem na resolução de problemas que estão além das capacidades de resolução individual de cada

um deles. No entanto, esta definição contrapõem dois aspectos fundamentais da distribuição de problemas: buscar as metas do SMA e a autonomia dos agentes. A organização formal de um SMA permite identificar e ajustar o equilíbrio entre estes dois aspectos, por meio de um conjunto de restrições comportamentais adotada pelos agentes. Uma boa organização consiste em determinar um espaço de busca de ações menor que aquele determinado pelo ambiente, que corresponde a todos os mapeamentos entre percepções e ações, porém maior que aquele que leve a finalidade do SMA, de modo a respeitar a autonomia dos agentes [5].

Uma classificação para os modelos organizacionais divide-os em modelos baseados na dimensão *funcional*, *estrutural* e *deôntica*. Os modelos funcionais operam no sentido de alcançar as *metas globais* do sistema. Os modelos estruturais focam-se em conceitos como *papéis* e *grupos* para organizar os agentes. Já os modelos deônticos baseiam-se na definição de *normas* e *permissões* dentro do SMA.

Para o FURGBol-Sim, o modelo adotado é o *Moise+*. Este modelo organizacional foi selecionado uma vez que é o único que aborda as três dimensões, estrutural, funcional e deôntica. Segundo o modelo *Moise+*, as três dimensões organizacionais formam uma Especificação Organizacional (OS, do inglês *Organisational Specification*). Quando os agentes adotam uma determinada OS, eles formam um Entidade Organizacional (OE, do inglês *Organisational Entity*).

A Especificação Estrutural (SS, do inglês, *Structural Specification*) do modelo *Moise+* é construída em três níveis: os comportamentos que um agente deve possuir quando ele é responsável por um papel, que corresponde ao seu *nível individual*; as relações de comunicação, autoridade e de conhecimento entre os papéis, que corresponde ao *nível social*; e a agregação dos papéis em grupos, que corresponde ao *nível coletivo*. A Especificação Funcional (FS, do inglês, *Functional Specification*) declara como o SMA alcança as *metas globais*, que são de caráter coletivo, decompondo-as em *planos* e distribuindo-os aos agentes por meio de *missões*. A Especificação Deôntica (DS, do inglês, *Deontic Specification*) descreve as obrigações e permissões dos papéis para as missões.

Para a implementação do SMA do FURGBol-Sim é utilizada a organização apresentada na Figura 3. Adotou-se a mesma organização de um jogador padrão do UvA Trilearn, onde todos o jogadores do time assumem o papel *deMeer*. Este papel possui duas atribuições: ficar bem posicionado, ou seja, se for o mais perto da bola deverá persegui-la, caso contrário ir para sua posição estratégica dentro do campo; e caso esteja com a posse da bola, chutá-la em direção ao gol.

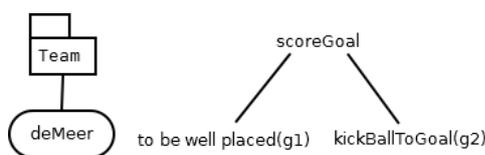


Figura 3. Organização do FURGBol-Sim

V. RESULTADOS

Para comparar as estratégias descritas de forma explícita e implícita, realizou-se 100 partidas das quais foram coletados os placares. A tabela I contém o somatório dos placares e a porcentagem de gols de cada equipe ao fim da simulação.

Placar	Estratégia Explícita	Implícita
Numérico	233	301
Porcentagem	43.6%	56.4%

Tabela I. RESULTADOS DA SIMULAÇÃO

A diferença de desempenho entre as equipes é atribuída aos componentes aleatórios do SoccerServer. Este fator faz com que as ações dos jogadores não sejam precisas. Portanto, acredita-se que as duas equipes possuem o mesmo desempenho.

VI. CONCLUSÕES

Este artigo descreveu a arquitetura e a implementação da equipe FURGBol-Sim, projetada para o Soccer Server 2D. Esta equipe integra aspectos do modelo de organização *Moise+* de SMA, uma arquitetura de agente cognitivo e o código fonte base do UvA Trilearn. Esta abordagem possui o mérito de restringir o espaço de busca de decisões do agente a um tamanho que garanta a realização dos objetivos coletivos do SMA, respeitando a autonomia dos agentes.

Nesta etapa do trabalho o objetivo é equipar o desempenho da equipe base original, com uma organização social implícita ao modelo, a uma equipe que integra os componentes de alto nível do agente cognitivo proposto em [2] junto com a explicitação da organização social original utilizando o modelo *Moise+*. Os resultados obtidos nos levam a acreditar que se obteve êxito no objetivo.

O resultado mais evidente deste processo é a modularização do processo de desenvolvimento em torno de conceitos sociais como meta globais, missões, papéis, etc. Deste modo, permite-se um desenvolvimento incremental da equipe de modo a atender um número cada vez maior de situações coletivas no ambiente. Não obstante, os próximos passos prevêem a extensão do número de situações atendidas pela equipe elevando a sua performance coletiva.

REFERÊNCIAS

- [1] V. Dignum and F. Dignum, "Modelling agent societies: Co-ordination frameworks and institutions," in *Proceedings of the 10th Portuguese Conference on Artificial Intelligence (EPIA'01)*. Berlin: Springer, 2001, pp. 191–204, INAI 2258.
- [2] A. L. da Costa and G. Bittencourt, "From a concurrent architecture to a concurrent autonomous agents architecture," in *International Joint Conference on Artificial Intelligence (IJCAI'99)*, 1999.
- [3] M. Chen, E. Foroughi, F. Heintz, Z. X. Huang, S. Kapetanakis, K. Kostiadis, I. N. Johan Kummeneje, O. Obst, P. Riley, Y. W. Timo Steffens, and X. Yin, *RoboCup Soccer Server: for Soccer Server Version 7.07 and later*, May 2001, www.robocup.org.
- [4] J. R. Kok, N. Vlassis, and F. Groen, "Uva trilearn 2003 team description," in *Proceedings CD RoboCup 2003*, D. Polani, B. Browning, A. Bonarini, and K. Yoshida, Eds., Padua, Italy, July 2003.
- [5] J. F. Hübner and J. S. ao Sichman, "Aplicação de organização de sistemas multiagentes em futebol de robôs," in *XI Escola de Informática do SBC*, vol. 1. Lages-SC: Angelo Augusto Frozza, 2003, pp. 119–147.