

# Utilização de Agentes Móveis para Recuperação de Informações no Sistema MATHNET

Helena Nunes e Sofiane Labidi

Departamento de Engenharia de Eletricidade

Universidade Federal do Maranhão

Campus do Bacanga

65080-040 São Luis – MA

{helena,labidi}@dee.ufma.br

## Palavras Chaves

Aprendizagem cooperativa, Aprendizagem e instrução auxiliada por computador, Agentes móveis, Recuperação de informações, Ambiente MATHNET.

## Abstract

*This article is a part of our work in the MATHNET[8] project, whose aim is the design and implementation of an environment for computer supported cooperative learning. We present here a system for information retrieval based on the mobile agent concept. We show how it could be used in the learning process increasing the functionalities of the proposed environment.*

## 1 Resumo

Este artigo apresenta parte de nosso trabalho dentro do projeto MATHNET [8] cujo objetivo é a concepção e implementação de um ambiente de ensino/aprendizagem que utiliza o paradigma da aprendizagem cooperativa suportada por computador. Apresentamos especificamente um sistema para recuperação de informações baseado no conceito de agentes móveis e mostramos seu uso como auxílio no processo de ensino/aprendizagem incrementando as funcionalidades do ambiente proposto.

## 2 Introdução

O computador vem sendo utilizado na educação desde a década de 70, quando surgiram os primeiros CAIs (*Computer Aided Instruction*) e STIs (*Sistemas Tutoriais Inteligentes*), todos

baseados no modelo de ensino tradicional. No final da década de 80, o aspecto da cooperação no processo de aprendizagem passou a ser bastante considerado, e com isso, houve a necessidade dos STIs suportarem tal conceito. O paradigma da aprendizagem cooperativa suportada por computador (CSCL<sup>1</sup>), que estende o conceito dos STIs clássicos para suportar o conceito de cooperação, vem sendo amplamente empregado desde a década de 90. Exemplos de sistemas que seguem essa abordagem são encontrados em [1] [2] [3] e [7].

O projeto MATHNET<sup>2</sup> tem como objetivo desenvolver um sistema que explore efetivamente o paradigma CSCL. O sistema é baseado em agentes e provê um ambiente computadorizado cooperativo de ensino/aprendizagem à distância. O sistema dispõe dos seguintes recursos: chat, vídeo conferência, assistente de resolução de problemas e banco de dúvidas [15] – um repositório compartilhado por todos. Utilizando tais mecanismos, os alunos podem realizar atividades passadas pelo professor, resolver problemas, solicitar explicação para uma dúvida etc. (cf. **Figura 1**).

Para explorar o potencial da Internet, algumas funcionalidades do MATHNET podem ser incrementadas se implementadas através da utilização do paradigma de agentes, particularmente agentes móveis [9], para atividades que envolvam a recuperação de informações pela rede.

### **3 MATHNET**

O MATHNET propõe um ambiente no qual os estudantes, divididos em vários grupos, cooperam e aprendem a partir da interação com seu próprio grupo, com o sistema, com o professor, e com outros grupos, utilizando recursos multimídia e a tecnologia de Intranet [11]. Unindo as idéias que fundamentam os STIs aos recursos de uma rede, o MATHNET define um ambiente tutorial inteligente cooperativo (cf. **Figura 1**).

Um grupo, ou *área cooperativa* [7], é formado por dois ou três estudantes interagindo com o sistema através de um terminal. Os estudantes cooperam utilizando o computador como um tutor ativo e como uma ferramenta que serve para intermediar as interações com o professor e com os outros grupos de estudantes. Essas interações são realizadas através dos recursos de comunicação da Intranet. Um professor interagindo com o sistema através de um terminal é considerado uma *área cooperativa* específica.

Os conceitos principais são os de agente tutor, um grupo formado por agentes tutores (TAG<sup>3</sup>), agente pedagógico, modelo de domínio e modelo do aluno cooperativo [13]. Inicialmente, existe a necessidade da definição de três interfaces: as interfaces do aluno, do professor e do especialista. As áreas cooperativas são assistidas pelo TAG, que é responsável pela aplicação do conteúdo ao aluno, avaliação e acompanhamento das atividades. Cada agente representa um especialista e é detentor de um certo conhecimento.

Para aumentar as funcionalidades do ambiente MATHNET, os alunos também podem contar com um agente de software móvel que serve para auxiliá-los na busca de explicação para uma determinada dúvida. Na seção seguinte apresentaremos os conceitos elementares de um sistema baseado em agentes móveis, um modelo conceitual genérico e suas primitivas básicas.

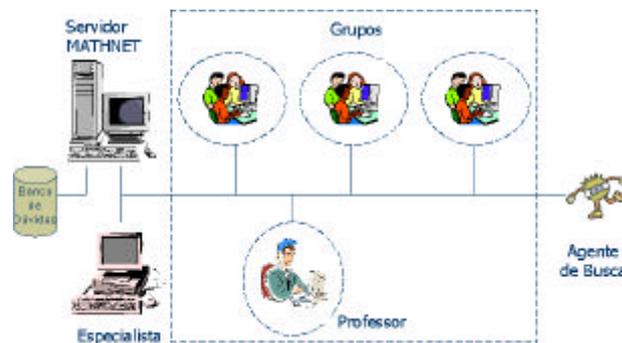


Figura 1. Organização do ambiente MATHNET

## 4 Agentes Móveis

Ultimamente muito se tem falado sobre a tecnologia de agentes móveis, e muitas aplicações para tais agentes podem ser encontradas na literatura (por exemplo em [4] [5] e [6]). Algumas das principais razões para o emprego de tais agentes podem ser encontradas em [5]. A questão da segurança é um dos pontos de maior crítica, porém muitos trabalhos que tratam desse aspecto têm sido propostos (como em [10], por exemplo).

Em Nwana e Ndumu [14], encontramos o ponto de vista de que os agentes móveis apresentam-se como uma solução clara para um problema ainda não tão bem definido. Nada que não possa ser resolvido utilizando abordagens tais como RPC (*Remote Procedure Call*) ou modelos de objetos distribuídos como RMI [19] (*Remote Method Invocation*) ou CORBA [18] (*Common Object Request Broker Architecture*). Nas seções seguintes

mostraremos como pretendemos tirar proveito da tecnologia de agentes móveis para recuperação de informações no MATHNET.

Uma proposta de padronização para os sistemas baseados em agentes móveis está sendo submetida à OMG (*Object Management Group*) desde de 1997 [16]. A base conceitual dos sistemas que seguem tal paradigma é constituída por dois conceitos elementares: agente móvel e o ambiente de execução do agente. Um agente móvel é um agente de software cuja execução não se limita ao ambiente onde foi iniciada sua execução e que possui cinco atributos básicos: estado, implementação, interface, identificador e autoridade [16]. Quando os agentes se movem, levam consigo seus atributos. O ambiente de execução compreende o contexto no qual o agente pode operar. Um modelo conceitual genérico é apresentado na **Figura 2**.

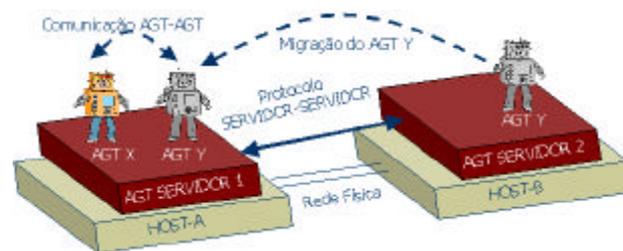


Figura 2. Modelo conceitual de um sistema baseado em agentes móveis

Os hosts A e B são máquinas conectadas através de uma rede. Para ser parte do sistema, cada host deve possuir um processo servidor executando, que constitui um ambiente de execução protegido. A camada do servidor é responsável por fornecer um ambiente de execução e também por disponibilizar primitivas aos programadores. A comunicação entre os diversos servidores em uma rede é propiciada pelo uso de um protocolo específico. Em um mesmo host, pode haver mais de um servidor de agentes rodando ao mesmo tempo e, nesses servidores, podemos ter ainda vários agentes executando simultaneamente. Um servidor de agentes pode ser especializado para prover serviços específicos.

O comportamento do sistema (cf. **Figura 2**) é dado pelo seguinte: no Host-A, o servidor de agentes AGT SERVIDOR 1 hospeda o agente AGT X. No Host-B, mais especificamente sob o servidor de agentes AGT SERVIDOR 2, um outro agente – AGT Y – está executando e decide migrar para o Host-A, seja porque ele precisa se comunicar com o AGT X ou porque ele requer alguns recursos que estão disponíveis somente no Host-A. Nisto, o Host-B efetua uma requisição de migração ao Host-A, e após a requisição ter sido aceita, o Host-B contata

o Host-A e transmite o AGT Y pela rede. O AGT SERVIDOR 1 aceita o agente que está sendo transmitido e prepara o ambiente para que o mesmo possa reiniciar sua execução. Dessa forma, o AGT Y continua sua execução no Host-A. O AGT Y pode, então, interagir localmente com o AGT X e/ou com o AGT SERVIDOR 1. De forma geral, os agentes podem migrar entre os diversos servidores de agentes e executar nos diversos ambientes providos por eles. Os sistemas baseados em agentes móveis devem prover primitivas básicas que suportem criação, destruição e transferência de agentes.

## **5 Problema da busca no MATHNET**

O MATHNET tem um banco de dúvidas que os alunos poderão consultar sempre que houver perguntas a serem respondidas, mas há outra fonte de informações que não pode ser esquecida: a Internet. Para usuários de um sistema como o MATHNET, por exemplo, não é interessante que a busca por informações esteja limitada ao escopo do sistema, desperdiçando o imenso potencial da Internet. Um dos problemas, no entanto, é o da descoberta da informação [14]. Como na Internet as informações não se encontram prontas, organizadas e não seguem a nenhum padrão específico, torna-se difícil a descoberta e a recuperação do que realmente é relevante. Agentes de informação e agentes de informação móveis têm sido propostos para amenizar esse problema buscando, filtrando e organizando informações da melhor maneira possível antes que as mesmas sejam entregues ao usuário. Se o usuário puder delegar tarefas, o sistema torna-se bem mais interessante. A questão da mobilidade não deve ser vista como uma necessidade, mas como algo que possa trazer melhorias quanto à interação dos objetos remotos, por exemplo. A mobilidade dos agentes traz à tona outros problemas: comunicação, ontologia e integração com outros sistemas [14]. No desenvolvimento do MATHNET, está sendo empregado o Processo Unificado [12], que tem como uma de suas principais características, ser dirigido por Casos de Uso. Na seção seguinte explanaremos um Caso de Uso do MATHNET em que um serviço de recuperação de informações pode ser empregado para aumentar uma funcionalidade do sistema.

## **6 Encontrando explicação para uma dúvida no MATHNET**

Seja o seguinte cenário básico de utilização do Caso de Uso Encontrar Explicação: ao resolver um problema ou realizar alguma atividade, o aluno pode submeter uma dúvida ao sistema e solicitar que o mesmo encontre uma explicação/justificativa. Isso é feito através de uma interface de dúvidas acionada a partir da interface do aluno (cf. **Figura 3**). Através da interface de dúvidas, o aluno poderá enviar uma mensagem diretamente para o professor,

consultar o banco de dúvidas ou acionar um agente de busca móvel. As duas primeiras opções dão ao aluno maior flexibilidade na utilização do sistema. Acionar o agente móvel subentende que os resultados não precisam ser imediatos. O aluno delega tarefas ao agente e continua suas atividades normalmente. Assim que o agente tiver algum resultado, o aluno será notificado. O seu funcionamento não depende do terminal do aluno estar on-line. A característica de mobilidade permite que o agente “viva” na rede. O agente possui um certo grau de autonomia para tomar suas próprias decisões, tais como migrar pelos nós da rede, ir para a Internet, gerar mensagens para o professor, consultar o banco de dúvidas, etc. O agente utiliza informações do modelo do aluno para tomar tais decisões e apresentar resultados “personalizados”.

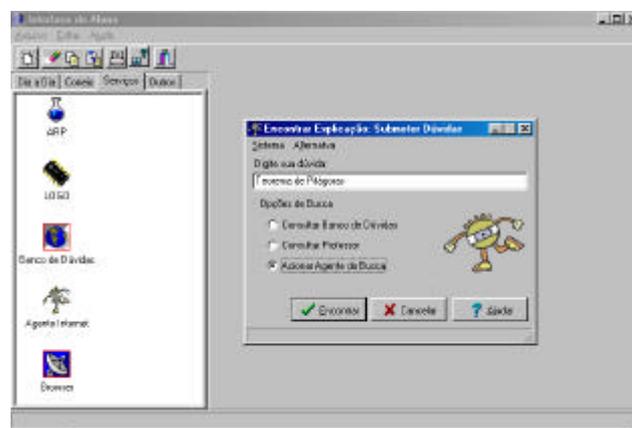


Figura 3. Interface do aluno e interface de dúvidas

O estágio de implementação encontra-se na construção da interface do aluno, a interface de resolução de problemas e a interface de dúvidas. Estamos utilizando a API Aglets da IBM [5] para implementar mobilidade. O agente será implementado em Java principalmente por razões de portabilidade e compatibilidade com a plataforma de execução escolhida para suportar mobilidade e pela facilidade de integração com outros sistemas. Soluções para os problemas de comunicação entre os agentes e de ontologia ainda estão sendo estudadas, e uma tendência é o uso de KQML [17].

## 7 Conclusão

Por querer estender a busca de informações para além das fronteiras do sistema, aproveitando o potencial da Internet, é interessante que algumas das funcionalidades do MATHNET possam suportar, além dos atributos básicos que definem um agente de software, também mobilidade. A questão da mobilidade na busca de informações no MATHNET pode

ser resolvida através do uso de uma plataforma como Aglets da IBM, por exemplo. Esse paradigma mostra-se bastante adequado para a resolução desse problema, por permitir melhores interações entre os objetos remotos. Como desafios, temos que nos preocupar, ainda que este não seja o foco principal do nosso trabalho, com os problemas da descoberta de informação, da comunicação, da ontologia e da integração com sistemas legados [14], entre outros, que poderão surgir no decorrer do desenvolvimento. A oportunidade de utilização desse paradigma também nos fornecerá subsídios para a descoberta de novos padrões de utilização e aplicações.

## 8 Referências

- [1] Boy, G.A. *Software Agents for Cooperative Learning*. In *Software Agents*. Edited Bradshaw J.M. AAAI Press/MIT Press. 1997.
- [2] Palthepe, S., Greer, J., and McCalla, G. *Learning by Teaching*. The Proceedings of the International Conference on the Learning Sciences, AACE. 1991.
- [3] Leroux, P. Vivet, M. and Brézillon, P. *Cooperation between Humans and a Pedagogical Assistant in a Learning Environment*. Second International Conference on the Design of Cooperative Systems (COOP'96). Juan-les-Pins, France, June 1996.
- [4] <http://www.tryllian.com>. A Tryllian Whitepaper. *Mobile Agents: The Work Force of the New Millennium*. November, 1999.
- [5] Lange, D.B. and Oshima, M. *Programming and Deploying Java Mobile Agents with Aglets*. Addison-Wesley, 1998.
- [6] Maes, P., Guttman, R. H. and Moukas, A. G., *Agents that Buy and Sell: Transforming Commerce as we Know It*. In *Communications of the ACM*. March, 1999.
- [7] Labidi, S. and Ferreira, J. S. *Technology-Assisted Instruction Applied to Cooperative Learning: the SHIECC Project*. In *Proceedings of the IEEE International Conference Frontiers in Education (FIE'98)*. Tempe, Arizona. November 4-7, 1998.
- [8] Labidi, S., Silva, J.C., Coutinho, L.R. and Costa, E.B. *MathNet: An Agent-Based Tutoring System for Supporting Cooperative and Distant Learning*. To appear in the *Proceedings of the International Conference on Computers and Advanced Technology in Education (CATE'2000)*. Cancun, Mexico. May 24-27, 2000.

- [9] White, J. *Mobile Agents*, In *Software Agents*, Bradshaw, J. Ed., MIT Press, 1997.
- [10] Karnik, N. *Security in Mobile Agent Systems*, PhD dissertation. Computer Science and Engineering, University of Minnesota, 1998.
- [11] Zimmerman, S. and Evans, T. *Building an Intranet with Windows NT 4*. O'Reilly Edt. 1996.
- [12] Booch, G., Rumbaugh, J. Jacobson, I. – *The Unified Software Development Process*, The complete guide of the Unified Process from the original designers, Addison-Wesley, 1999.
- [13] Coutinho, L. R. *A Modelagem do Aprendiz em Ambientes de Aprendizagem por Computador baseados em Atividades de Resolução de Problemas*. Dissertação de Mestrado, COPIN, UFBP – Campus II, Campina Grande, PB, março de 1999.
- [14] Nwana, H. Ndumu, D. *A Perspective on Software Agents Research*. The Knowledge Engineering Review, January 1999.
- [15] Castro, C.H.P. *Implementação de uma versão cliente/servidor de um fórum de dúvidas para o ambiente SHIECC de ensino cooperativo computadorizado*. Monografia de Conclusão de Curso, DEINF, UFMA, Fevereiro 1999.
- [16] The Object Management Group, “*The Mobile Agent System Interoperability Facility*”, OMG TC Document orbos/98-03-09, The Object Management Group, Framingham, MA, 1998. [www.omg.org](http://www.omg.org). The MASIF specification from OMG.
- [17] Finin, T. and Labrou, Y. “*KQML as an Agent Communication Language*”, In *Software Agents*. Bradshaw, J.M. (ed.), MIT Press, Cambridge, MA, pp. 291-316, 1997.
- [18] <http://www.corba.org>
- [19] <http://java.sun.com/products/jdk/rmi>

---

<sup>1</sup> Computer Supported Cooperative Learning

<sup>2</sup> O projeto MATHNET é parte do programa PROTEM CNPQ, mantido pelo Governo Brasileiro. (Proc. Inst. N° 680060/99-5)

<sup>3</sup> Tutorial Agents Group