

Definição de sequências de estudo com base no Ant System e em informações presentes em objetos de aprendizagem

Lucas Moreno de Araujo¹, Fabiano Fagundes¹

¹ Curso de Sistemas de Informação – Centro Universitário Luterano de Palmas (CEULP/ULBRA).
Teotônio Segurado 1501 SUL – 77054-970 – Palmas – TO – Brasil.
{lucasmaraujo,thilfa}@gmail.com

Abstract: *This paper presents an algorithm to provide the definition of study sequences from a set of learning objects described by metadata, considering the student's characteristics that is using the resources. The main technologies involved in this context are: LOM - Learning Object Metadata, providing the metadata structure for describing the domain and the Ant System, Artificial Intelligence algorithm based on the behavior of ants to define paths.*

Resumo: Este artigo apresenta um algoritmo para propiciar a definição de sequências de estudo a partir de um conjunto de objetos de aprendizagem descritos por metadados, levando-se em consideração as características do usuário que está utilizando tais recursos. As principais tecnologias envolvidas nesse contexto são: LOM – Learning Object Metadata, fornecendo a estrutura de metadados para a descrição do domínio e o Ant System, algoritmo da Inteligência Artificial baseado no comportamento das formigas para a definição de caminhos.

1. Introdução

Ao realizar seus estudos, os acadêmicos, atualmente, baseiam-se em conteúdos disponíveis na web, em repositórios de conteúdos didáticos disponibilizados por universidades, colégios, instituições educacionais, dentre outros. Em cada uma destas fontes podem existir conteúdos que tenham nível básico, sendo destinados a uma visão inicial acerca do assunto ou ainda nível avançado, contendo detalhes rebuscados e de difícil entendimento. Assim, depara-se com a seguinte dificuldade: como fazer para selecionar as informações, de forma que seja possível obter uma visão inicial acerca



Secretaria de Educação
Profissional e Tecnológica



Ministério
da Educação





do assunto para posteriormente aprofundar nos conceitos e construir o conhecimento desejável de uma forma coerente? Ou seja, como fazer para determinar uma sequência de estudos que se adequa às necessidades e características do aluno?

Neste artigo, é apresentada a proposta do AntStudy, um algoritmo para a definição de sequências de estudo baseado no *Ant System* e que segue a ideia desenvolvida por Brito *et al* (2002) de trabalhar esta técnica em sistemas não-determinísticos, grandemente subjetivos, como é o processo ensino--aprendizagem. O artigo está estruturado da seguinte forma: inicialmente serão apresentados alguns conceitos sobre LOM e *Ant System*, utilizados no desenvolvimento do AntStudy. Posteriormente serão mostradas as tecnologias e os detalhes de funcionamento do algoritmo desenvolvido. Por fim, são apresentadas as referências bibliográficas utilizadas como fontes de pesquisa.

2. LOM

LOM - *Learning Object Metadata* - consiste em uma estrutura de metadados referentes a um objeto de aprendizagem, que tem a finalidade de descrever características importantes sobre ele (IEEE, 2002, p. 5). O LOM tem o objetivo de facilitar a busca e aquisição dos objetos de aprendizagem, tanto por parte de instrutores de ensino, como também por sistemas programados para tal fim. É importante frisar que esse modelo não informa como os metadados devem ser consumidos por sistemas automatizados que trabalham com os objetos de aprendizado.

Ao todo, os metadados são agrupados em nove categorias principais de acordo com seu valor semântico:

- Geral: nesta categoria, encontram-se dados acerca do objeto de aprendizagem como um todo. Refletem características como a identificação e a composição do objeto;
- Ciclo de vida: os metadados da seção Ciclo de Vida descrevem a história e o estado do objeto de aprendizagem, assim como quem influenciou em seu conteúdo durante o desenvolvimento. Refletem uma das características dos objetos de aprendizagem, que é a possibilidade de atualização do conteúdo;





- Meta-metadados: são dados referentes às características do esquema de metadados que está sendo utilizado para descrever um objeto de aprendizagem. Esses dados não descrevem o objeto de aprendizagem em si;
- Técnico: esse grupo de metadados descreve as características inerentes ao objeto de aprendizagem, assim como os requerimentos necessários de hardware e software para acessar seu conteúdo;
- Educacional: o conjunto de metadados existentes nessa categoria reflete características de cunho pedagógico do objeto de aprendizagem, cujo objetivo é proporcionar um aprendizado de qualidade para quem estiver utilizando o recurso;
- Direito: esse grupo descreve os direitos de propriedade intelectual sobre um objeto de aprendizagem;
- Relacionamento: permite que sejam estabelecidos relacionamentos entre os objetos de aprendizagem;
- Anotações: armazenam comentários acerca de um objeto de aprendizagem. O metadado armazena o autor do comentário no atributo Entidade, a Data e o comentário, propriamente dito, no atributo Descrição;
- Classificação: apesar de alguns dos metadados expostos anteriormente proverem um formato de classificação dos objetos de aprendizagem (quanto à complexidade, nível de interação, quantidade de relacionamentos, entre outros), é possível propor novos formatos de classificação que atendam a particularidades de um determinado contexto.

A especificação traz ainda detalhes importantes como nome, significado, tamanho (de dados) e exemplos de utilização para cada um dos elementos presentes na hierarquia do LOM. Este trabalho busca propor a utilização de um conjunto destes metadados com a tecnologia Ant System, descrita a seguir, para auxiliar na definição de uma sequência de estudos.





3. Ant System

Atualmente, sabe-se que existem diversas abordagens para a resolução de problemas complexos na área de Inteligência Artificial. Entre elas, encontra-se a *Swarm Intelligence* (inteligência de enxames), cujos princípios são baseados no comportamento dos insetos e outros animais que vivem em sociedade (DORIGO, BIRATTARI, STÜTZLE, 2006, p. 28). Dentre esses animais, é possível encontrar abelhas, pássaros, peixes, formigas, etc.

O algoritmo proposto neste trabalho é baseado no *Ant System* (AS), que é uma técnica presente em um ramo de estudos da *Swarm Intelligence* denominado *Ant Colony Optimization* (otimização por colônias de formigas), cujos estudos são inspirados no comportamento das formigas. Um dos principais aspectos biológicos que inspiraram os estudos de computação voltados ao comportamento das formigas é o sistema de comunicação que alguns insetos utilizam que é mediado por uma substância denominada feromônio (DORIGO, BIRATTARI, STÜTZLE, 2006, p. 29).

Ferreira e Zarbin (1998, p. 3) afirmam que “feromônios são substâncias químicas secretadas por um indivíduo (nesse caso, um inseto) que permitem a comunicação com outro indivíduo da mesma espécie”. Enquanto caminham, as formigas deixam trilhas de feromônio que podem ser detectadas pelas outras formigas. Quanto mais feromônio em um determinado local, mais atrativo ele se torna, portanto, há uma maior probabilidade de que outras formigas passem pelo mesmo local. É importante lembrar ainda que o feromônio evapora com o passar do tempo (DORIGO e STÜTZLE, 2004, p. 12).

Dorigo, Maniezzo e Corloni (1996, p. 29-30) afirmam que o *Ant System* é versátil, permitindo que seja aplicado a variações de um mesmo problema com pequenas alterações para satisfazer seus requisitos. Além disso, ele possui robustez, tornando viável sua aplicação em várias áreas de análise combinatória.

O algoritmo apresenta alguns pontos fundamentais durante o seu processamento (DORIGO e STÜTZLE, 2004). O primeiro passo consiste em inicializar alguns parâmetros, como o grafo a ser percorrido e a quantidade de agentes que irão participar do processo. Posteriormente, deve-se criar um laço de repetição, que irá realizar um processamento durante várias iterações até se atingir uma solução aceitável.



A segunda parte do AS corresponde à atualização do feromônio nas trilhas, que é realizada em duas etapas:

1. Assim como o feromônio natural, o algoritmo considera uma taxa de evaporação, possibilitando que soluções ruins tenham cada vez menos probabilidade de serem percorridas por formigas artificiais durante a execução.
2. Após a evaporação, as taxas de feromônio presentes na trilha criada por um agente são modificadas. Dependendo do contexto, pode-se adaptar esse valor de acordo à trilha percorrida por cada formiga para representar particularidades do contexto. Por exemplo: em um problema para encontrar o menor caminho, pode-se adaptar esse valor de acordo ao comprimento de cada trilha, possibilitando que as trilhas menores tornem-se mais atrativas às formigas.

4. Metodologia

Este trabalho foi desenvolvido como uma pesquisa prática voltada para a busca de solução para o problema de determinação de uma sequência de estudos. Para tal fim foi realizado um levantamento bibliográfico sobre os itens que contribuiriam para o desenvolvimento da proposta de solução aqui apresentada na forma de implementação de um sistema que se utilize da técnica Ant System em conjunto com os conceitos de objetos de aprendizagem conforme propostos pela IEEE.

Assim, para o desenvolvimento do trabalho proposto foram relacionados, inicialmente os atributos do LOM aos de uma representação de um usuário acadêmico.

Posteriormente, foi criado um *Application Profile* (AP) a partir do LOM proposto pela IEEE. Os atributos escolhidos para a constituição deste AP foram baseados principalmente nos estudos realizados na fase anterior, visto que eles podem influenciar o comportamento do algoritmo. Para a constituição do AP, foi realizado um levantamento das características do LOM que poderiam ser relacionadas com os atributos de um usuário estudante e, assim, das nove categorias estudadas, três mostraram-se apropriados para o objetivo deste trabalho:



Secretaria de Educação
Profissional e Tecnológica



Ministério
da Educação





- geral: para o aproveitamento do identificador, palavras-chave e linguagens do objeto de aprendizagem;
- educacional: por possuir informações relacionáveis a um perfil, como dificuldade e nível de interatividade;
- relacionamentos: para a representação das relações iniciais entre os objetos de aprendizagem, que forneceria uma base para a construção inicial da estrutura de dados do algoritmo.

Dentro de cada um desses grupos do LOM, foram escolhidos alguns atributos que poderiam ser representados de forma numérica, para posterior utilização na realização de um cálculo de similaridade entre o usuário e o objeto de aprendizagem. Tal cálculo é explicado em detalhes posteriormente.

Depois de criado o AP, foi estruturada uma representação do usuário contendo os atributos necessários para o processamento do algoritmo. Os atributos do usuário foram escolhidos levando-se em consideração o AP criado na etapa anterior. Cada atributo do usuário teve uma característica equivalente a um atributo do objeto de aprendizagem. Essa relação de equivalência possibilitou a realização do cálculo de similaridade entre o objeto e o usuário.

Após a representação do usuário, foi iniciada a fase de desenvolvimento do algoritmo para a definição de sequências de estudo, baseado no *Ant System*. As próximas seções apresentam detalhes das classes implementadas no algoritmo AntStudy.

5. Classes do algoritmo

O AntStudy é composto por quatro classes principais: Grafo, ObjetoAprendizagem, Usuario e AntSystem.

5.1. Representação dos Objetos de Aprendizagem no Grafo

A classe Grafo é responsável por realizar desde a montagem da estrutura do grafo de objetos de aprendizagem até a visualização dos componentes na interface em tempo de execução. O primeiro



passo para a construção do grafo consistiu na criação de uma estrutura semelhante a uma matriz de adjacências que apresenta, na posição $[i][j]$:

- o valor 0, indicando que não há um relacionamento entre os objetos de aprendizagem identificados respectivamente por i e j ;
- o valor 1, indicando que o elemento j tem como pré-requisito o objeto identificado por i . Nesse caso, os objetos apresentam um relacionamento explícito;
- ou o valor 2, indicando que os elementos i e j são pré-requisito para um terceiro objeto de aprendizagem. Esse relacionamento (chamado implícito) tornou-se necessário para que não fosse possível criar uma sequência de estudos desrespeitando os pré-requisitos determinados no domínio, composto pelos objetos de aprendizagem.

Na Figura 1 é possível visualizar um exemplo de matriz de adjacência e o respectivo grafo que poderia ser gerado a partir dela.

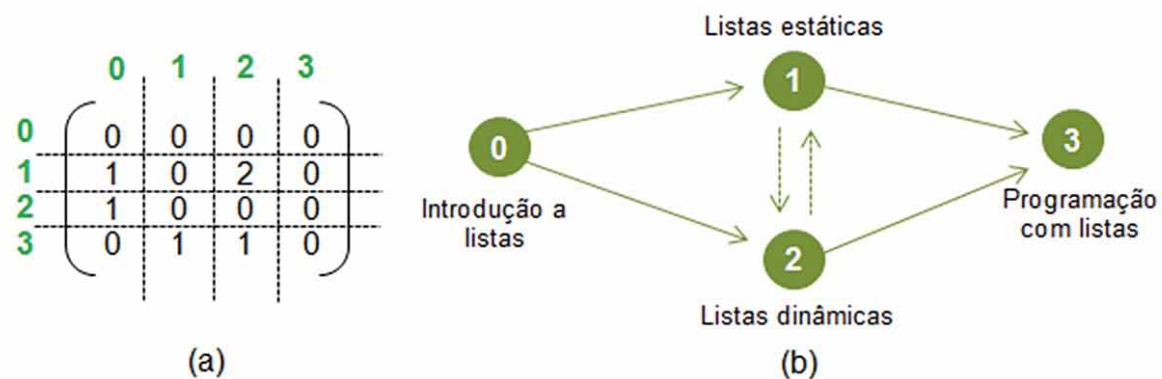


Figura 1: Pseudocódigo do algoritmo Ant System





A matriz de adjacência apresentada em (a) pode ser traduzida para o grafo apresentado em (b). A lista de objetos utilizada nesse exemplo apresenta os materiais na seguinte ordem: 0 - Introdução a listas, 1 - Listas estáticas, 2 - Listas dinâmicas e 3 - Programação com listas. O número 1 na posição [1][0] (considerando [linha][coluna]), indica que o material presente na posição 1 da lista de objetos (Listas estáticas) tem como pré-requisito o material 0 (Introdução a listas). Os números 1 presentes na mesma linha, como ocorre em [3][1] e [3][2], indicam que os materiais presentes na posição 1 e 2 (Listas estáticas e dinâmicas) são pré-requisitos para o material presente na posição 3 (Programação com listas). Sendo assim, eles possuem um relacionamento implícito (representado na matriz pelo número 2). Nesse caso, os relacionamentos explícitos são representados por setas contínuas e os implícitos por setas tracejadas.

Após a definição desses valores na matriz, é preciso criar os vértices e as arestas. Assim, é instanciado um vértice para cada objeto de aprendizagem presente na lista recebida como parâmetro no construtor da classe **Grafo**. No primeiro momento, os vértices são somente instanciados, sem relacionamentos entre si. Na sequência, são criadas as arestas a partir da execução de uma varredura na matriz de adjacências que estabelece os relacionamentos entre os vértices. Na criação das arestas, é importante ressaltar que relacionamentos explícitos são representados por uma única aresta; relacionamentos implícitos são representados por duas setas tracejadas.

Após criar os vértices e arestas no grafo, é preciso adicionar um vértice para representar o usuário. Para isso, faz-se uma varredura na matriz para verificar quais objetos de aprendizagem não possuem pré-requisitos e é estabelecida uma aresta entre o usuário e esses objetos. Por fim, é gerado um quadro mostrando o grafo com os vértices, arestas, quantidade de feromônio. Na Figura 2 é apresentado um exemplo de visualização gerada pelo AntStudy.



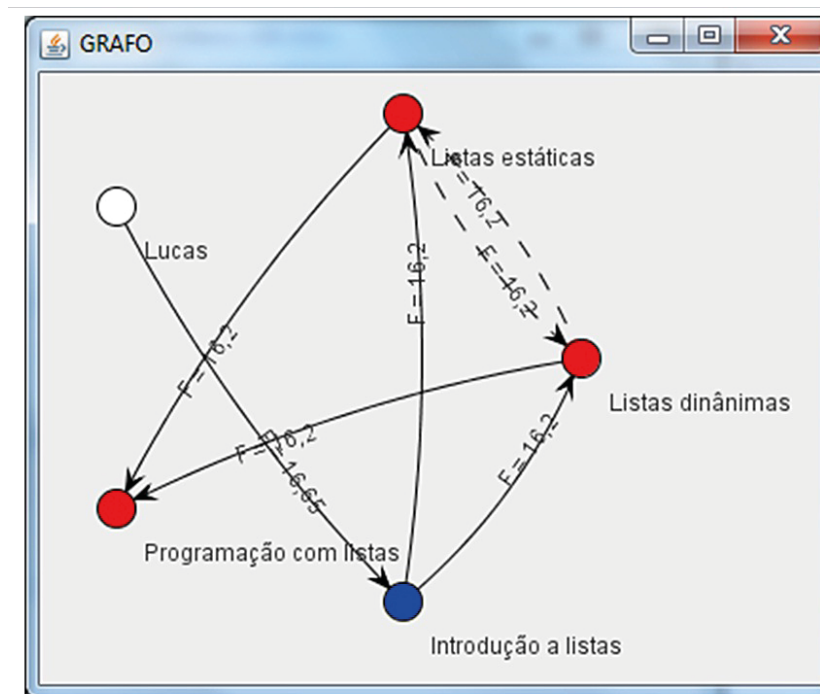


Figura 2: Exemplo de visualização

No quadro apresentado na Figura 2 existem vértices que representam o usuário; um objeto de aprendizagem não visitado; um objeto de aprendizagem percorrido. Em relação às arestas, há setas contínuas para os relacionamentos explícitos e setas tracejadas para os relacionamentos implícitos. Sobre cada seta é possível visualizar a quantidade de feromônio existente no elo.

A classe **Vertice** é utilizada juntamente com a classe **Aresta** para a constituição dos elementos no grafo. De forma geral, um vértice pode representar tanto um objeto de aprendizagem como também um usuário. Uma instância dessa classe possui como atributos: **Id**, para identificar o vértice, **Nome**,



que no caso de um material recebe o título e no caso de um usuário recebe um nome próprio, e o atributo **Visitado**, que representa o estado do vértice (*true* ou *false*). Esse último atributo é importante principalmente na etapa inicial da definição da sequência de estudos, para que um usuário não passe pelo mesmo objeto mais de uma vez.

A classe **ObjetoAprendizagem** representa um objeto de aprendizagem com seus respectivos atributos, que foram baseados na especificação do LOM: Identificador, Título, Palavras_chave, Linguagens, TipoRecursoAprendizado, NívelInteratividade, Dificuldade, Relacionamentos. Das nove categorias definidas no LOM, foram contemplados atributos de três: Geral, Educação e Relacionamento. Os demais atributos não foram incluídos no algoritmo por não influenciarem no contexto deste trabalho.

Para representar os relacionamentos entre dois objetos de aprendizagem, foi utilizada a classe auxiliar **Relacionamento**. Ela possui a função de englobar um relacionamento existente entre dois materiais; sendo assim, um objeto de aprendizagem possui uma lista de instâncias de Relacionamento, em que é possível indicar o **Tipo** e a **Referencia** a um objeto específico.

5.2 Representação do usuário

A classe **Usuario** promove a representação de um usuário, da qual são extraídas informações que influenciam na execução do algoritmo AntStudy. Deve-se ressaltar que no grafo, há uma instância da classe Vertice que representa o usuário. Essa instância faz uma referência à instância da classe Usuario que armazena os atributos do mesmo. Os atributos escolhidos para a representação do usuário estão diretamente relacionados aos atributos dos objetos de aprendizagem: Nome, Conceitos, Linguagens, TiposMateriais, NívelInteratividade, Dificuldade.

O principal método presente nesta classe é o **calcularHeuristica()**. Esse valor heurístico é utilizado para definir a adequação de um objeto de aprendizagem a um determinado usuário. Para a realização do cálculo, foi necessário criar duas estruturas para armazenar de forma numérica as características definidas nas instâncias do objeto de aprendizagem e na representação do usuário. Essas estruturas foram definidas a partir de dois *HashMaps*, contendo chave e valor. Na Figura 3 é possível visualizar a composição da estrutura que é utilizada para caracterizar um material.



Secretaria de Educação
Profissional e Tecnológica



Ministério
da Educação



Estrutura geral

Palavras-chave	Linguagens	Tipo de recurso de aprendizado	Nível de interatividade	Dificuldade
----------------	------------	--------------------------------	-------------------------	-------------

Exemplo

Java	Estruturas	ptBR	Apresentação de slides	Nível de interatividade	Dificuldade
1	1	1	1	0.6	0.8

Figura 3: Hashmap que caracteriza um objeto de aprendizagem

Na estrutura apresentada na Figura 3, as primeiras posições contemplam características booleanas. Sendo assim, no início dessa estrutura encontram-se todas as palavras-chave, as linguagens e o tipo de recurso de aprendizado do material como chaves do *HashMap*, sendo que essas chaves referenciam o valor 1, para indicar a presença desses elementos. Para o nível de interatividade e dificuldade, os valores podem variar na escala de 1 a 5, sendo que após a indicação do valor, ocorre um processo de normalização (os valores são divididos por 5, que é o valor máximo desses atributos, para que fiquem representados em uma escala de 0 a 1).

Para representar o usuário é utilizada uma estrutura semelhante à do material, sendo que as palavras-chave correspondem aos conceitos que o usuário domina e as linguagens são os idiomas que ele consegue entender. É importante ressaltar que no *HashMap* do usuário também devem estar presentes as palavras-chave, linguagens e tipo de recurso de aprendizado do material em questão, para que o cálculo da adequação esteja correto. Supondo que um material está escrito em inglês e português, e o usuário domina somente o português, o *HashMap* deve ter uma entrada para português, referenciando o valor 1 e uma entrada para inglês, referenciando o valor 0, para indicar que ele não possui conhecimento nesse idioma.



Após a criação dos dois *HashMaps* é realizado o cálculo do co-seno entre os dois vetores de características. Nesse ponto poderia ser utilizada outra forma de determinar a similaridade entre os objetos. O co-seno foi escolhido somente para a realização de testes empíricos. Esse cálculo é realizado a partir da equação apresentada na Figura 4.

$$sim = \frac{\sum_{i=1}^n (x_{ki} * y_{ki})}{\sqrt{\sum_{i=1}^n (x_{ki})^2} * \sqrt{\sum_{i=1}^n (y_{ki})^2}}$$

Figura 4: Cálculo do co-seno (SILVA, Edeilson et. al., 2011, p. 93, adaptada)

O cálculo da similaridade a partir do co-seno retorna um valor entre 0 e 1, sendo que quanto mais próximo de 1, mais similares são os dois vetores, portanto, mais indicado um objeto de aprendizagem a um determinado perfil de usuário. No cálculo apresentado na Figura 4, x_{ki} e y_{ki} correspondem aos valores presentes no *HashMap* do usuário e objeto de aprendizagem, respectivamente, em que i representa a chave (o cálculo é efetuado levando-se em consideração somente as chaves comuns das duas estruturas).

5.3 Definição da sequência de estudos

A classe principal do projeto é denominada **AntSystem**, visto que nela se realiza a definição da sequência de estudos para um determinado usuário. O algoritmo presente nessa classe contempla as três fases propostas para um algoritmo genérico baseado na técnica do *Ant System*: construção do grafo, busca de uma solução e posterior atualização das taxas de feromônio. Para a primeira parte do processamento, foi utilizado um método chamado **recuperarListaObjetos()**, para instanciar os objetos de aprendizagem com seus respectivos atributos e colocá-los em uma lista.



Secretaria de Educação
Profissional e Tecnológica



Ministério
da Educação





Posteriormente, deve-se instanciar o usuário para o qual será definida a sequência de estudos. A construção do grafo é feita a partir da invocação do método `construir()`, presente na classe `Grafo` (apresentada anteriormente). Com o usuário instanciado e o grafo construído, devem-se inicializar as taxas de feromônio igualmente em todas as arestas do grafo.

Depois de inicializar as taxas de feromônio, o algoritmo executa um laço de repetição para a realização de um percurso completo pelo grafo, contemplando todos os objetos de aprendizagem. O percurso se inicia a partir do vértice que representa o usuário. Nesse algoritmo, a figura da formiga encontra-se implícita na execução do algoritmo ao sair do nó corrente e percorrer o grafo pelos objetos de aprendizagem para cada usuário. A cada objeto visitado tem-se uma solução e, portanto, a atualização das taxas de feromônio. O percurso é finalizado quando todos os objetos são visitados.

O primeiro processo presente nesse laço de repetição é a recuperação dos vizinhos do nó corrente. Depois disso ocorre a primeira filtragem: não se pode incluir na sequência de estudos um objeto de aprendizagem que foi visitado. Os objetos que passarem pela etapa de filtragem seguem para a próxima etapa que indicará qual objeto será colocado na sequência naquele momento. Essa indicação é baseada em uma probabilidade, que é influenciada tanto pela quantidade de feromônio presente na aresta, como também por um valor heurístico, que indica o quanto um objeto de aprendizagem é adequado a um usuário. Nesse trabalho, a probabilidade foi calculada com base na sentença matemática proposta por Dorigo e Stützle (2004, p. 70) para o problema do Caixeiro Viajante, que é apresentada na Figura 5.

$$p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\gamma_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\gamma_{il}]^\beta}, \text{ se } j \in N_i^k$$

Figura 5: Cálculo da probabilidade de acesso a um objeto de aprendizagem





Na Figura 5, τ_{ij} representa um valor heurístico que caracteriza a aresta entre o vértice i e o vértice j , γ_{ij} remete-se à quantidade de feromônio presente nesse elo e os expoentes α e β determinam o peso que cada um desses dois valores vai ter na definição da probabilidade. A letra k representa uma formiga (nesse caso um usuário) e N_i^k é o conjunto de vértices que k pode acessar a partir do vértice i .

O método `escolherVertice()` indicará qual objeto será colocado na sequência. Antes de atribuir à variável `NoCorrente` o novo nó que pertence à sequência de estudos, o objeto de aprendizagem que está sendo visualizado é adicionado a uma pilha, denominada `CaminhoPercorrido`. Essa estrutura de dados auxilia no processo de continuação da sequência de estudos quando um usuário está visualizando um objeto de aprendizagem que não tem mais sucessores. Nesse caso, o algoritmo executa um *backtracking*, passando pelos vértices que foram visitados para verificar os sucessores que ainda não foram visitados.

Após a escolha do vértice, inicia-se a etapa de atualização das taxas de feromônio no grafo:

1. primeiramente, é aplicada uma taxa de evaporação de 0,01 a todas as arestas do grafo, para possibilitar que os caminhos pouco percorridos se tornem depreciados pelo algoritmo com o passar do tempo. Essa taxa é equivalente a 1% da quantidade de feromônio atual. Na proposta de Dorigo e Stützle (2004, p. 71), essa taxa receberia um valor de 0,5. No AntStudy, não foi utilizado o valor original, pois em testes empíricos, identificou-se que as taxas de feromônio decresciam de forma acelerada a cada execução.
2. posteriormente, a aresta percorrida recebe uma taxa de incremento de feromônio, calculada a partir da Figura 6.

$$\Delta\gamma_{ij} = \mu * \tau_{ij}$$

Figura 6: Incremento do feromônio no AntStudy





Na equação apresentada, a taxa de variação de feromônio ($\Delta\gamma_{ij}$) aplicada como incremento em uma aresta é resultado da multiplicação entre a nota fornecida pelo usuário ao objeto de aprendizagem (μ , que varia entre 1 e 10) e pelo valor heurístico calculado entre o usuário e o material.

Todos esses processos de filtragem, cálculo de probabilidades, escolha do vértice e atualização do feromônio repetem-se até que todos os nós do grafo sejam visitados; finalizando, assim, o algoritmo. Tem-se então, uma sequência de estudos definida, com base nas características do usuário e dos objetos de aprendizagem presentes no contexto. Deve-se ressaltar que, a cada escolha de um material para a sequência de estudos, ocorre um incremento da quantidade de feromônio presente na aresta que foi percorrida.

6. Considerações e contribuições

No AntStudy, o ambiente da natureza foi representado a partir de um conjunto de objetos de aprendizagem, descritos a partir de características propostas em um modelo de referência denominado LOM. Esse modelo proporcionou a cobertura de uma grande quantidade de características e a possibilidade de se fazer um *Application Profile* para utilizar somente os atributos necessários em cada contexto.

Como principal contribuição deste trabalho tem-se a verificação da possibilidade de aplicar AS a problemas que não são da área de otimização, conforme as aplicações convencionais da técnica. Isso foi possível a partir do entendimento dos processos propostos no algoritmo e posterior modelagem do contexto conforme as etapas do AS.

Como sugestão de trabalhos futuros, poderia ser desenvolvido um modelo de estudante mais elaborado, englobando um conjunto maior de características acerca do usuário. Além disso, propõe-se a criação de um conjunto de estereótipos, de forma manual ou automática (a partir da clusterização), para a definição de sequências de estudo para um grupo de pessoas com características semelhantes. Tal modelo poderia ser acompanhado de um modelo de diagnóstico, responsável por atualizar a localização dos usuários nos grupos, conforme eles adquirem mais conhecimentos e para agrupar os novos usuários que passarem a utilizar o sistema.



Outra proposta de trabalho futuro seria considerar um conjunto maior de características dos objetos de aprendizagem na definição das sequências de estudo. Para tal, deveriam ser acrescentadas ainda novas informações para representar os usuários e implementadas modificações no cálculo da heurística que define a similaridade entre estudantes e materiais.

Além disso, pode-se modelar e implementar um banco de dados para armazenar os objetos de aprendizagem, usuários e informações do grafo (como feromônios e arestas percorridas), para possibilitar a utilização do algoritmo em um ambiente real. Sendo assim, os usuários poderiam acessar um sistema que utiliza o AntStudy para o sequenciamento de estudos em momentos distintos, sem que as informações do percurso realizado sejam perdidas.

7. Referências bibliográficas

Brito, Parcilene Fernandes; Fagundes, Fabiano; Alves, João Bosco da Mota. "Uma Estrutura para Definição de Seqüências de Estudos Baseada na Técnica Ant System". In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO – SBIE, 13, 2002, São Leopoldo. Anais... São Leopoldo: UNISINOS, 2002. p. 247 – 253.

IEEE. Learning Technology Standarts Committee (LTSC). "Draft Standart for Learning Object Metadata (IEEE - 1484.12.1)". 2002. 44 p.

Dorigo, Marco; Birattari, Mauro; Stützle, Thomas. "Ant colony optimization". *IEEE Computational Intelligence Magazine*, v. 1, n. 4, p. 28-39, 2006.

Dorigo, Marco; Stützle, Thomas. "Ant colony optimization". Massachusetts: The MIT Press, 2004. 305 p.

Dorigo, Marco; Maniezzo Vittorio; Colorni Alberto. "Ant System: Optimization by a colony of cooperating agents". *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, v. 26, p. 29-41, fev. 1996.



Secretaria de Educação
Profissional e Tecnológica



Ministério
da Educação

