CORDENONSI, A. Z., BERNARDI, Giliane, BASTOS, Fábio da Purificação de, ARAÚJO, Fabrício Viero, MÜLLER, Felipe Martins. Objeto de Aprendizagem para o Ensino de Heurísticas e Metaheurísticas em Cursos Superiores de Pesquisa Operacional In: XX Encuentro Nacional de Docentes en Investigación Operativa e XVIII Escuela de Perfeccionamiento en Investigación Operativa, 2007, Mar del Plata - Argentina. **Anais do XX Encuentro Nacional de Docentes en Investigación Operativa e XVIII Escuela de Perfeccionamiento en Investigación Operativa**. Mar del Plata: Univeridad Nacional de Mar del Plata, 2007.

OBJETO DE APRENDIZAGEM PARA O ENSINO DE HEURÍSTICAS E META-HEURÍSTICAS EM CURSOS SUPERIORES DE PESQUISA OPERACIONAL

ANDRE ZANKI CORDENONSI[1] (andrezc@inf.ufsm.br), **FELIPE MARTINS MÜLLER**[2] (felipe@inf.ufsm.br),
FÁBIO DA PURIFICAÇÃO DE BASTOS[3] (fbastos@ce.ufsm.br),
FABRÍCIO VIERO ARAÚJO[4] (vieros@terra.com.br), **GILIANE BERNARDI**[5] (giliane@unifra.br)

RESUMO

O ensino de heurísticas e meta-heurísticas, no âmbito da educação superior, encontra seu nicho nos cursos de graduação das áreas de computação, matemática e engenharias, principalmente. Em relação à metodologia pedagógica utilizada nas aulas, a explanação sobre as características dos algoritmos heurísticos é uma prática comum entre a comunidade docente. No entanto, percebe-se uma dificuldade crescente nos alunos em visualizar as nuances da lógica dos métodos propostos, pois a aplicação dos mesmos em exemplos pequenos se utilizando dos instrumentos tradicionais de ensino (quadro-negro e giz) é inócua para algoritmos mais complexos. O presente artigo apresenta o LOBO(*Learning Object Based on Optimization*), um objeto de aprendizagem que pode ser utilizado como auxiliar na condução de disciplinas que trabalhem com algoritmos heurísticos e meta-heurísticos para o Problema do Caixeiro Viajante, desenvolvido através de uma proposta pedagógica que insira o educando no contexto de sua própria aprendizagem.

PALAVRAS-CHAVE: Problema do Caixeiro Viajante, Educação Superior, Pesquisa Operacional

ABSTRACT

The heuristics and meta-heuristics teaching, in high education, its usually find out in graduation courses of the computation areas, mathematics and engineering's. However, the pedagogical methodology used in the lessons are the traditional approach based on the one way communication between the professor and the students. The students usually present an increasing difficulty to understand the logics of the considered methods; therefore the

---

[1] Professor do Departamento de Documentação, da Universidade Federal de Santa Maria(UFSM), Doutorando em Informática na Educação na Universidade Federal do Rio Grande do Sul(UFRGS)

[2] Professor do Departamento de Eletrônica e Computação, da Universidade Federal de Santa Maria(UFSM), Doutor em Engenharia Elétrica pela Universidade Estadual de Campinas(UNICAMP)

[3] Professor do Departamento de Educação, da Universidade Federal de Santa Maria(UFSM), Doutor em Educação pela Universidade de São Paulo(USP)

[4] Mestrando no Programa de Pós-Graduação em Engenharia da Produção da Universidade Federal de Santa Maria(UFSM)

[5] Professora do Centro Universitário Franciscano(UNIFRA), Doutoranda em Informática na Educação na Universidade Federal do Rio Grande do Sul(UFRGS)

application of the same ones in small examples using educational traditional instruments (blackboard and chalk) is very poor for more complex algorithms. This paper presents the LOBO (Learning Object Based on Optimization), a learning object that can be used to assist in the conduction of heuristics and meta-heuristics lessons using the Traveling Salesman Problem. The object was developed through a pedagogical proposal including the students in the context of its proper learning inserts.

KEYWORDS: Traveling Salesman Problem, Higher Education, Operational Research

1. INTRODUÇÃO

A educação contemporânea se preocupa cada vez mais nos processos de ensino-aprendizagem que passem da simples *reprodução* do conhecimento para, efetivamente, sua *produção*. Desta forma, o conhecimento científico ou filosófico que deve ser levado ao educando é, essencialmente, um recorte da realidade como ela se apresenta agora, podendo ser alterada constantemente durante o desenvolver do ser humano. Posto isso, é claro afirmar que os desafios dos professores também são aumentados segundo (Popper, 1983), pois ele precisa trabalhar com os alunos formas de estarem preparados para a mudança das suas bases conceituais, que podem ocorrer continuamente a partir de sua partida dos bancos escolares ou até mesmo se metamorfear em saberes completamente diversos. O aprender a aprender adquire novos significados e se torna premente na sociedade atual.

A Informática na Educação tem representado uma revolução na educação tradicional, nas políticas de educação públicas e nos formatos e metodologias propostas para a interação educando – educador. No entanto, a mera adição de suportes tecnológicos à sala de aula não contribui de forma transformadora, constituindo-se, em muitos casos, de um mero acessório ou de objetos de propaganda institucionais. O investimento tecnológico só se justifica caso o objetivo final da implementação do sistema seja plenamente satisfatório, ou seja, a efetiva aprendizagem do educando. Além disso, evidencia-se que o processo de ensino precisa mudar para um processo educativo, uma vez que, segundo Moran, “ensino e educação são conceitos diferentes. No ensino organiza-se uma série de atividades didáticas para ajudar os alunos a compreender áreas específicas do conhecimento (ciências, história, matemática). Na educação o foco, além de ensinar é ajudar a integrar ensino à vida, conhecimento e ética, reflexão e ação, a ter uma visão da totalidade”. (Moran, 2000).

No que concerne ao ensino das áreas de engenharia, matemática e ciência da computação, se destaca a área de heurísticas e meta-heurística com aplicação em otimização. A otimização tem como objetivo a resolução da alocação de recursos, tipicamente limitados, com o intuito de alcançar determinados objetivos. Considerando que existe um conjunto discreto de soluções possíveis, a resolução de um problema de otimização combinatória inclui o processo de geração, avaliação e comparação de soluções, num

determinado limite de tempo. A otimização é um tópico central nas áreas da ciência da computação, inteligência artificial e pesquisa operacional.

Uma heurística pode ser definida com um algoritmo que encontra uma solução factível, não necessariamente a melhor solução, para um determinado problema com uma determinada função objetivo, num tempo computacional razoável (Diaz, 1996). Estas mesmas heurísticas também podem ser facilmente adaptáveis a outros tipos de problemas envolvendo as áreas de Inteligência Artificial, Redes de Computadores e Arquitetura de Computadores. No entanto, somente a aplicação de heurísticas pode não resolver todas as classes de problemas. As meta-heurísticas, que se caracterizam por guiarem outras heurísticas, têm sido particularmente interessantes na resolução de problemas complexos. Em relação as meta-heurísticas, pode-se citar como principais métodos: *simulated annealing,* busca tabu, algoritmos genéticos, algoritmos meméticos e busca local genética.

O objetivo deste artigo é apresentar o objeto de aprendizagem LOBO (*Learning Object Based on Optimization*) que auxilie o educador nas disciplinas de heurísticas e meta-heurísticas, fornecendo uma ferramenta de fácil manipulação e útil no que concerne à explanação dos diversos algoritmos apresentados aos alunos, através do conceito de ação simulada, ou seja, favorecendo o aprendizado pela ação, recorrendo a situações simuladas no computador e explorando os mais diversos problemas.

2. ENSINO DE HEURÍSTICAS E META-HEURÍSTICAS: UM PROBLEMA ATROZ

No que concerne ao ensino das áreas de engenharia, matemática e ciência da computação, se destaca a área de heurísticas e meta-heurísticas com aplicações em otimização. A otimização tem como objetivo a resolução da alocação de recursos, tipicamente limitados, com o intuito de alcançar determinados objetivos. Considerando que existe um conjunto discreto de soluções possíveis, a resolução de um problema de otimização combinatória inclui o processo de geração, avaliação e comparação de soluções, em um determinado limite de tempo. Conforme (Corne *et al*, 1999), a otimização é um tópico central nas áreas da ciência da computação, inteligência artificial e pesquisa operacional.

Apesar de extremamente relevante no que concerne à aplicabilidade, o ensino de heurísticas e meta-heurísticas é, pedagógica e tecnologicamente, normalmente apresentado de forma desinteressante e dependente da fórmula *giz + quadro-negro*. Este tipo de pedagogia foi definida como *práticas bancárias* por (Freire, 1981 *apud* DE BASTOS *et al*, 2005), que afirma ser a escola tradicional o ato de depositar, os alunos são depósitos e os professores, os depositantes. Para o autor, o professor como mero *informador* ou *transmissor de conteúdo* não estabelece uma prática de ensino adequada à nova sociedade do conhecimento onde, muito mais que *aprender um determinado conteúdo*, o educando deve *aprender a aprender*.

No caso específico do ensino de heurísticas e meta-heurísticas, este usualmente apresenta um embasamento matemático forte e necessita um grande número de passos para que o processo faça sentido. Este processo burocrático, mas necessário, torna-se desestimulante se realizado através de um processo didático tradicional. Os diversos algoritmos heurísticos e meta-heurísticos propostos na literatura apresentam, invariavelmente, soluções muito boas para problemas de pequena dimensão. Como situação exemplo, consideremos o Problema do Caixeiro Viajante(PCV). O PCV é, provavelmente, o mais conhecido e estudado dos problemas de otimização, sendo definido como encontrar uma rota para um determinado veículo/pessoa que necessita visitar *n* diferentes cidades, iniciando e encerrando sua viagem na primeira cidade. Normalmente, deseja-se otimizar a rota de tal forma a minimizar o custo da mesma, que se confunde usualmente com a distância percorrida.

Quando se fala no ensino de heurísticas e meta-heurísticas a atenção se volta, principalmente, para a modelagem, solução e análise de problemas decisórios, sendo que um estudo de caso completo corresponde à realização de experimentos numéricos com modelos lógico-matemáticos. Estes experimentos envolvem geralmente grande volume de cálculos repetitivos, fazendo-se necessário o uso intensivo do computador. Também se torna necessário o emprego de um conjunto de fórmulas e técnicas matemáticas que, se não forem ilustradas de forma aplicada, corre-se o risco de que o alcance destas não seja compreendido pelos alunos (Dávalos, 2002).

Apesar da complexidade bem conhecida do mesmo, sua formulação, mesmo matemática, é simples e intuitiva, o que o torna um problema ideal para estudantes de graduação que o reconhecem imediatamente como uma questão a ser resolvida, apesar de muitos não enxergarem, em um primeiro momento, a explosão combinacional de soluções possíveis e sua conseqüente influência no tempo computacional gasto para resolver os problemas na otimalidade. Mesmo os computadores mais rápidos não conseguem realizar todos os cálculos necessários para a geração de todas as rotas possíveis (como exemplo, um problema relativamente simples de 25 cidades gera cerca de 6.2×10^{23} rotas possíveis e alguns milhões de anos de cálculos).

Desta forma, se torna impraticável a especificação de todas as rotas possíveis, o que garantiria o encontro da *melhor* rota possível. No entanto, mesmo com a utilização de heurísticas e meta-heurísticas o número de rotas calculadas para problemas maiores (15 cidades ou mais) se torna muito grande para ser realizado sem a utilização de um suporte adequado, no caso o computador.

Por outro lado, a utilização de exemplos pequenos (5 a 7 cidades) não explora suficientemente o problema para que seja possível uma correta demonstração da eficácia e da eficiência dos algoritmos demonstrados em aula. Usualmente, os algoritmos tendem a ter um bom desempenho no início, piorando ao final da construção da rota. Neste caso específico, se o exemplo utilizado for pequeno, tal característica, essencial e fortemente trabalhado nos

algoritmos construtivos, se torna estéril nas aulas planejadas, pois os diferentes algoritmos, desde os mais simples até os mais complexos, tendem a ter desempenhos parecidos em problemas pequenos.

Tal fato não ocorre em problemas maiores. Com o crescimento do número de cidades, a complexidade do problema aumenta e a eficiência dos algoritmos mais complexos se torna visível. Desta forma, é pedagogicamente necessário que os alunos visualizem o funcionamento destes algoritmos em problemas maiores e não somente em situações modelo, onde os resultados podem ser mascarados. No entanto, a simples resolução de problemas maiores não resolve o problema. O aluno precisa *visualizar* o que está acontecendo.

Existe um vácuo tecnológico e pedagógico que precisa ser preenchido para que o ensino das técnicas de heurísticas e meta-heurísticas se torne mais acessível para o corpo discente. Estas técnicas usualmente são discutidas dentro dos cursos de engenharia, administração e informática. No entanto, a simples exposição de cada técnica não prove ao aluno a aprendizagem significativa necessária para a real compreensão do fenômeno que está sendo investigado. Para que uma técnica seja eficientemente explorada e compreendida pelo aluno, ele precisa verificar como seu funcionamento altera as soluções encontradas no decorrer do processo, assim como os parâmetros alteram as mesmas. No entanto, as questões envolvendo a implementação das diferentes técnicas inviabilizam que todas as mesmas possam ser desenvolvidas por completo pelos alunos. Uma forma interessante proveria que o aluno testasse as técnicas e compreendesse seu funcionamento para que, posteriormente, utilizasse este aprendizado no desenvolvimento de suas próprias técnicas e funções heurísticas. Dentre as técnicas que podem ser utilizadas para o incremento da real compreensão dos alunos, a animação pode ser considerada a mais viável e eficaz. A animação por computador pode ser definida como uma seqüência temporal de mudanças visuais em uma determinada cena. Adicionando a mudança de posição dos objetos com translações ou rotações, a animação baseada em computador pode mostrar variações de tempo no tamanho do objeto, cor, transparência ou mesmo textura. Quando utilizada na educação, a animação pode ter duas abordagens. A primeira, denominada passiva, ocorre quando o usuário é um mero espectador da animação. Esta metodologia é comparada a experiência de assistir um filme ou uma seqüência pré-definida de eventos. Ela pode ser útil para a complementação dos aspectos vistos nas aulas presenciais, abordando somente o conhecimento pré-estabelecido pelo desenvolvedor da animação. Usualmente, sistemas que trabalham com este tipo de abordagem têm como característica principal a alta qualidade gráfica.

No entanto, a abordagem passiva possui uma limitação clara no que concerne a experimentação de novos conjuntos de dados, pois os parâmetros da animação são estabelecidos pelo desenvolvedor e não podem ser alterados pelos educados. Com a eliminação desta restrição e a incorporação de facilidades no manuseio das diretivas da animação, é possível para o aluno realizar suas próprias experiências, direcionando o seu conhecimento de

acordo com o seu ritmo de aprendizagem. Esta abordagem é denominada ativa. Sistemas que trabalham com esta metodologia geram animações com uma qualidade gráfica mais baixa, com o intuito de não sobrecarregar a máquina. Como exemplos, é possível citar os trabalhos de (Martins *et al*, 2003) e (Zachary, 2004). Do mesmo modo que a experimentação durante a operação das animações enriquece o aprendizado mais do que a mera observação passiva delas, é de se esperar que, com um sistema onde a própria implementação das animações gráficas é facilitada a ponto de poder ser realizada pelo estudante, a absorção do funcionamento das heurísticas seja ainda mais intensa.

Sistemas que trabalham com o conceito de animações que podem ser modificadas através de parâmetros e/ou escolhas do utilizador são conhecidos como simuladores. Para (Belhot *et al*, 2001) a simulação se apresenta como uma emulação de uma situação real, usualmente construída a partir de um modelo, que pode corresponder a uma representação simplificada da realidade. O principal conceito na simulação está no processo de experimentação do modelo, determinando, deste modo, o comportamento do sistema às mudanças em sua estrutura. O autor também exemplifica que o *modelo* em si pode se tornar o objeto de estudo, o que pode fornecer detalhes sobre o funcionamento e o conhecimento sobre o sistema. Na concepção de um sistema de simulação para o ensino de heurísticas e meta-heurísticas esta característica é fundamental, pois o modelo dos algoritmos é o tópico central onde se concentra o foco dos educadores e educandos. Em relação ao desenvolvimento do simulador, o sistema foi projetado com o intuito de formar um novo *objeto de aprendizagem* que pode ser utilizado livremente pelos professores e alunos. Este conceito é discutido no próximo capítulo.

3. OBJETOS DE APRENDIZAGEM

A utilização da tecnologia de objetos está se tornando cada vez mais difundida entre os envolvidos da área de informática. A Orientação a Objetos, surgida na década de 60, extrapolou os limites apenas da programação para invadir as áreas de análise e projeto de sistemas, modelagem de banco de dados, entre outras. A modelagem e implementação de sistemas baseados em objetos torna o desenvolvimento de sistemas o mais próximo da realidade observada. Para (Santanché, 2002) várias são as vantagens do uso de objetos na construção de *softwares,* entre elas: (a) objetos podem ser construídos de forma independente, podendo ser reaproveitados futuramente, evitando que novos projetos partam “do zero”; (b) podem ser combinados com outros objetos não limitando seu uso a um programa específico; (c) são facilmente configuráveis e podem ser personalizados de acordo com determinada necessidade; (d) a estrutura de construção dos objetos apresenta uma eficiente divisão entre sua lógica interna e sua interface de comunicação com o meio exterior, tornando mais fácil sua utilização.

Adotando esta filosofia surgem os chamados “Objetos de Aprendizagem” (*Learning Objects*), sendo utilizados de uma maneira até mesmo mais ampla do que a estabelecida pelo paradigma da orientação a

objetos. Para (IP *et al*, 2001), "Objetos de Aprendizagem incluem mecanismos de *software* relacionados ao tratamento dispensado para o conteúdo, apoio as interfaces de usuário para a interatividade e capacidade de acesso aos mesmos". O IEEE *Learning Technology Standards Committee (LTSC)* define Objetos de Aprendizagem como: "...qualquer entidade, digital ou não digital, que pode ser utilizada, reutilizada ou referenciada durante o aprendizado apoiado pela tecnologia. O aprendizado apoiado pela tecnologia inclui sistemas de treinamento baseados em computador, sistemas de aprendizado à distância, sistemas inteligentes de instrução, entre outros." (IEEE, 2002). Já (TAROUCO, FABRE e TAMUSIANAS, 2003) define um objeto de aprendizagem como a idéia de recursos suplementares ao processo de aprendizagem, que possam ser reutilizados para apoiar a aprendizagem.

Muitos autores têm usado a metáfora dos blocos de construção do jogo LEGO para descrever Objetos de Aprendizagem. A metáfora LEGO expressa a noção de "pequenas peças de instrução (blocos LEGO) que podem ser montadas (agrupadas) em uma grande estrutura instrucional." Para (DOWNES, 2000), a utilização de Objetos de Aprendizagem percorre um caminho onde os *softwares* educacionais eram desenvolvidos de forma customizada para cada situação necessária, passando para uma produção em massa (acesso a todos) e, finalmente, chegando na customização em massa dos componentes produzidos e distribuídos.

A popularização dos objetos de aprendizagem (NASCIMENTO, 2005; TAROUCO, FABRE e TAMUSIANAS, 2003; MALLOY e HANLEY, 2001; POLSANI, 2003) trouxe como preocupação investigativa formas de padronizar a especificação, construção e identificação dos objetos. Como em todas as áreas da computação, a adoção de modelos e padrões para o desenvolvimento de *software* se torna importante para a rápida prototipação e implementação dos mesmos. Aliada as modernas técnicas de desenvolvimento de *software*, é necessário também permitir a interoperabilidade dos objetos nos mais diversos tipos de AVEAs – Ambientes Virtuais de Ensino e Aprendizagem. Ambientes desenvolvidos com padrões fechados ou que não possam *importar* diferentes tipos de objetos traçam, seguramente, seus caminhos para a obsolescência. O próprio esforço de desenvolver novos e atrativos ambientes virtuais podem tornar os próprios sistemas inadequados em tempo inferior aos objetos de aprendizagem – em tempo, um *objeto de aprendizagem* para a simulação de movimentos retilíneos uniformes, matéria básica da física dos sólidos, pode não sofrer alterações por diversos anos.

Desta forma, é necessário estabelecer formas de transferir objetos de aprendizagem de um AVEA para o outro sem perda de sua funcionalidade, até mesmo porque não há escolhas *certas* em termos de um ambiente virtual de ensino/aprendizagem, e sim, escolhas mais ou menos adequadas ao próprio *lócus* de trabalho dos docentes e alunos, estratégias pedagógicas, financeiras e outros aspectos de caráter puramente tecnológico, como parque computacional (*hardware* e *software*) instalado na escola/instituição.

Dos padrões reconhecidos internacionalmente, apresentamos dois deles:

- Dublin Core: o *Dublin Core Metadada Initiative* se caracteriza por um fórum aberto com o intuito de desenvolver padrões de metadados que mantenham as características da interoperabilidade e do acesso eletrônico, suportando uma grande variedade de propósitos e modelos de negócios. Apesar de não ter sido criado especificamente para fins educacionais, o DC pode ser utilizado em objetos de aprendizagem e elementos bibliográficos tradicionais, tais como artigos e livros, o que representa uma maior representatividade em termos de generalidade. O padrão possui 18 elementos (Souza *et al*, 2000): título, autor, palavras-chave, categoria, descrição, publicador (entidade responsável por tornar o recurso disponível), colaborador, data (de disponibilização), tipo (texto, som, imagem, *software* ou outros), formato (identificação de *software* e *hardware* necessários à exibição/operação), acesso (ISBN ou URL), identificador do recurso (único), fonte (citação das fontes, quando necessário), idioma, relação (com outros recursos), cobertura (características espaciais ou temporais), direito autoral e contato;
- IEEE 1484.12.1 – 2002: a norma supra citada foi desenvolvida pelo *Learning Technology Standards Committee* do IEEE, que estabelecia uma especificação com atributos para descrever especificamente objetos educacionais (Fabre *et al*, 2003). Ela é utilizada, atualmente, em diversos *empacotadores* de objetos educacionais, como o ADL SCORM(SCORM, 2004) e o IMS Learning Package(IMS, 2005), servindo como base para a distribuição de objetos pela Internet. Esta norma, conhecida como *Learning Objects Metadata* - LOM especifica atributos agrupados em nove categorias para descrição de um objeto educacional(IEEE, 2002): (a) geral, contendo os atributos identificador, catálogo, entrada, título, idioma, descrição, palavras-chave, cobertura, estrutura e nível de agregação; (b) ciclo de vida, contendo os atributos versão, estado, tipo de contribuição, entidades que contribuíram e data; (c) meta-metadata, contendo os atributos identificador, catálogo, entrada, tipo de contribuição, entidades que contribuíram, data, esquema dos metadados e idioma; (d) requerimentos técnicos, que agrupa os requisitos formato, tamanho, localização, tipo de tecnologia, nome da tecnologia, versão mínima, versão máxima, duração, habilidades técnicas, instalação e requerimentos de sistemas computacionais pré-instalados; (e) educacional, contendo os atributos tipo de interatividade, recursos de aprendizagem, nível de interatividade, semântica, usuário final esperado, ambiente de utilização, faixa etária, descrição, contexto, dificuldade, tempo esperado e idioma; (f) direitos, que agrupa os atributos custo, direito autoral e condições de uso; (g) relações, contendo os atributos acerca do tipo de relação, recurso, identificador, catálogo, entrada e descrição; (h) anotações, que agrupa os atributos entidade, data e descrição; e (i) classificação, com os

atributos propósito, taxonomia, fonte, identificador, entrada, descrição e palavras-chave.

Analisando os padrões *Dublin Core* e *LOM*, é interessante observar que o último se apresenta de forma muito mais completa (e complexa) do que o primeiro. Apesar das dificuldades em catalogar um objeto seguindo o padrão LOM de forma completa (IMS, 2006), ele apresenta alguns pontos positivos que se destacam: sua utilização em empacotadores de objetos, como citado acima, principalmente seguindo o padrão *Instructional Management Systems (IMS) - IMS Learning Metadata*; a possibilidade de se utilizar um sub-conjunto dos grupos e atributos acima. Como definido em (IEEE, 2002), um conjunto de metadados que engloba alguns dos grupos e atributos da IEEE 1484.12.1 também é considerado compatível com o padrão estabelecido. Desta forma, os projetos de desenvolvimento de objetos educacionais podem estabelecer seus próprios sub-conjuntos de atributos e manter a compatibilidade entre os sistemas.

Considerando as asserções acima, o objeto de aprendizagem LOBO, que será apresentado na próxima seção, mantém junto ao seu sistema um conjunto de metadados baseados na IEEE 1484.12.1 para consulta e catalogo em bibliotecas virtuais de objetos educacionais.

4. OBJETO DE APRENDIZAGEM LOBO: ALGORITMOS PARA O PROBLEMA DO CAIXEIRO VIAJANTE

A escolha do problema básico que será utilizado como suporte as aulas de heurísticas e meta-heurísticas é crucial para o bom desenvolvimento da disciplina. A escolha do problema a ser resolvido pelos alunos deve ser bastante criteriosa. Se, por um lado, é importante que o mesmo seja o mais realista possível, para que os educandos se sintam naturalmente atraídos pela situação que, possivelmente lhes é familiar, o professor também deve ter em mente que o problema deve, necessariamente, ter soluções viáveis.

Como a área de heurísticas e meta-heurísticas é bastante ampla e a gama de problemas que pode ser resolvido é extremamente densa e rica, a primeira tarefa do professor é o estabelecimento de uma meta ou problema objetivo, que vai ser discutido durante toda a disciplina. Mesmo que, aparentemente, a exploração de um único problema possa condensar sobremaneira o séqüito de problemas possíveis dentro da área de pesquisa operacional, a intenção de uma disciplina introdutória de heurísticas e meta-heurísticas deve prover ao educando uma forma de repensar suas próprias idéias acerca de conceitos básicos como *melhor escolha* ou *melhor rota* em um dado problema. O aprendizado de pesquisa operacional não pode estar centrado no conhecimento de um ou outro algoritmo, e sim nos obstáculos e preceitos básicos que permeiam a maioria das situações, independente do problema que está sendo atacado. Desta forma, é possível que um educando consiga generalizar os conhecimentos construídos para atacar os mais diferentes problemas.

Considerando as asserções apresentadas acima, a decisão do grupo de pesquisa foi de utilizar um único problema para as aulas de heurísticas e meta-heurísticas, sendo que o escolhido foi o Problema do Caixeiro Viajante(PCV). O PCV é, provavelmente, o mais conhecido e estudado dos problemas de otimização. Sua fácil aplicação nos mais diversos campos do saber o levaram a um patamar diferenciado dos demais, pois inúmeros pesquisadores, das mais diferentes áreas, convergiram seus esforços para a resolução adequada do mesmo.

Durante a realização das aulas e até mesmo fora destas, o professor pode, a qualquer momento, utilizar o objeto de aprendizagem desenvolvido para propor questões que envolvam a comparação de resultados ou implementação de novos algoritmos (o código fonte, distribuído como *software livre*, é incluído dentro do próprio objeto, permitindo que os estudantes interajam ele de forma muita mais rica que as soluções no estilo *caixa preta* – *software* fechado, de código proprietário e sem possibilidade de modificações). A figura 1 apresenta a interface principal do objeto que será descrito a seguir.

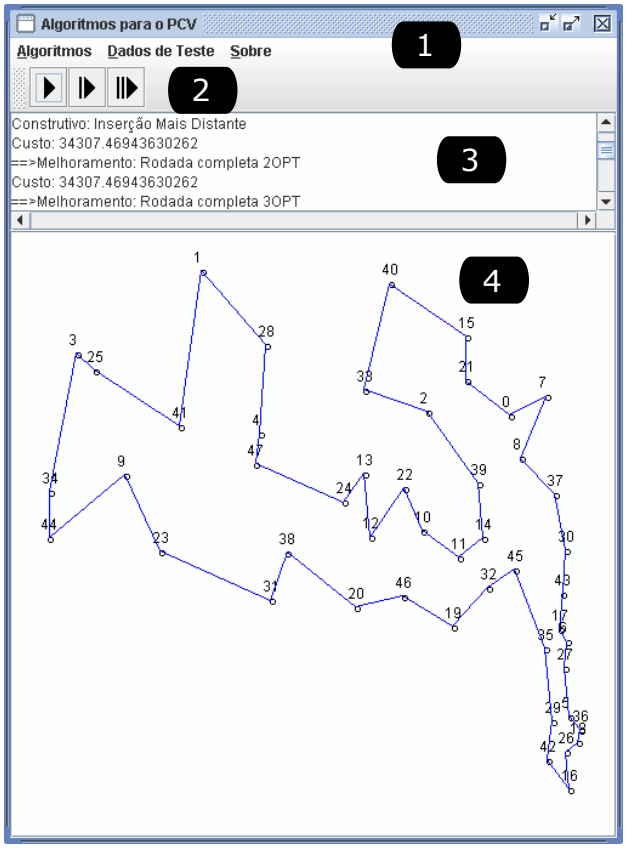

FIGURA 1 – INTERFACE PRINCIPAL DO OBJETO DE APRENDIZAGEM

A seção um apresenta as funcionalidades do menu do objeto. Este menu é subdivido em *Algoritmos*, *Dados de Teste* e *Sobre*. No primeiro, o usuário pode escolher o algoritmo que deseja simular. As opções

correntemente implementadas incluem três algoritmos construtivos (totalmente aleatório, vizinho mais próximo e inserção mais distante), dois de melhoramento (heurística 2-OPT e 3-OPT) e duas meta-heurísticas (*simulated annealing* e GRASP). É importante salientar que a mera escolha do algoritmo não implica na execução propriamente dita do mesmo. No menu *Dados de Teste*, o usuário pode escolher um entre cinco possibilidades pré-cadastradas com problemas clássicos da literatura (TSPLIB), além de poder informar seus próprios valores através da opção *Inserir Dados*, onde ele deve informar o número de cidades do problema, a cidade de origem e as distâncias euclidianas de cada cidade que serão computadas para a criação da matriz de distâncias (matriz de custos). Finalmente, o menu *Sobre* apresenta uma janela com as principais informações sobre o objeto e a ajuda do sistema.

A seção dois apresenta os controles de simulação que são utilizados para a efetiva utilização do objeto de aprendizagem. Inicialmente, o usuário deverá escolher o tipo de algoritmo que será simulado. Ele também deverá escolher os dados de teste para a simulação, através do menu apropriado. Com base nestas escolhas, o aluno tem três escolhas:

- simular: o primeiro botão efetua uma rodada completa do algoritmo selecionado. No caso dos algoritmos construtivos, ele executa a simulação até encontrar uma rota completa. Para os algoritmos de melhoramento, a opção escolhida será simulada utilizando uma heurística gulosa, ou seja, ela permanecerá ativa enquanto existirem melhoramentos a serem efetuados. É importante salientar que, ao simular um algoritmo de melhoramento 2-OPT e 3-OPT, antes do início do mesmo o usuário deverá responder duas questões:
  - tamanho da vizinhança: número de vizinhos que o algoritmo de melhoramento deverá pesquisar durante sua execução;
  - critério de parada de uma rodada: neste caso o usuário deverá escolher se o algoritmo deverá encontrar a *melhor* rota dentro de uma específica rodada ou a *primeira* rota que contém um custo menor que a solução corrente.

  Durante a execução de uma *simulação*, a interface principal do objeto de aprendizagem (seção 4 da figura 1) permanece estática, sem demonstrar graficamente o andamento da simulação. Ao final da mesma, o objeto apresenta um relatório final na forma de uma janela, onde é possível obter as informações sobre o problema, algoritmos utilizados, número de rodadas (se for o caso), custo da rota obtida, especificação da rota e a matriz de distâncias (figura 2). Após a apresentação do resultado, o usuário pode escolher novamente suas opções de configuração e efetuar uma nova simulação ou sair do sistema;

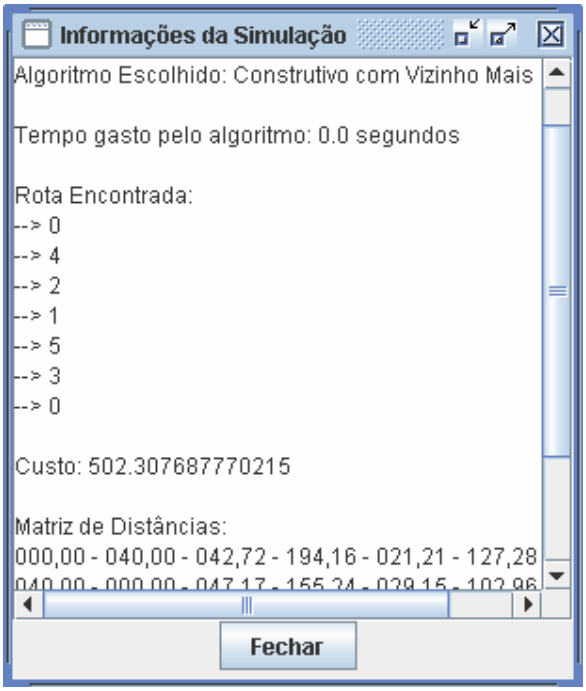

FIGURA 2 – RELÁTORIO FINAL DA SIMULAÇÃO

- passo de simulação: o segundo botão efetua um passo completo de simulação (disponível somente para os algoritmos construtivos e de melhoramento). Esta opção está disponível para que o aluno visualize graficamente, passo a passo, o funcionamento do algoritmo escolhido. Desta forma, a complexidade inerente das heurísticas desenvolvidas em sala de aula podem ser melhor compreendidas. Um importante aspecto deste objeto de aprendizagem está em dar liberdade ao aluno de, inicialmente, escolher o problema ou inserir seus próprios dados e, após a configuração destes parâmetros, simular passo a passo o algoritmo, visualizando, deste modo, o funcionamento da heurística com suas próprias configurações. Para o algoritmo construtivo do vizinho mais próximo, o objeto apresenta a escolha de cada cidade cuja distância até a cidade atual é a menor possível. Desta forma, ao escolher a opção *passo de simulação*, o algoritmo escolhe a cidade mais próxima da origem e constrói a rota até lá. Após, o usuário necessita clicar novamente no passo de simulação para efetuar uma nova rodada do algoritmo e escolher uma nova cidade. A implementação dos passos de simulação através do pressionar do botão do objeto tem o objetivo de fornecer um maior controle do usuário final. Implementada desta forma, o aluno pode escolher o seu próprio ritmo de simulação, andando mais rápido quando quer ou simplesmente parando para observar um passo mais detalhadamente. Para os algoritmos de melhoramento, se optou em realizar uma simulação mais sofisticada que englobasse as nuances da implementação. Salienta-se aqui que o objeto realiza uma rodada completa de 2-OPT ou 3-OPT (verifica, uma única vez, todas as possibilidades de troca), e depois apresenta o resultado ao aluno

através de uma simulação de três etapas. A primeira etapa apresenta o problema antes da aplicação do algoritmo de melhoramento utilizando arcos de cor azul para representar a configuração atual do sistema. A segunda etapa é apresentada ao usuário dois segundos após. Nesta etapa, o usuário tem a possibilidade de observar quais são os arcos que foram escolhidos para serem removidos da rota (vermelho). Finalmente, após um novo intervalo de dois segundos, as linhas grossas são removidas e os novos arcos são inseridos, formando a nova rota. Esta simulação é apresentada na figura 3, passos I a IV. Independente do algoritmo escolhido, ele continuará enquanto o usuário pressionar o botão *passo de simulação* até que não seja mais possível executar um passo (para os algoritmos construtivos este critério representa a construção de uma rota completa; para os algoritmos de melhoramento se não é possível melhorar a rota) ;

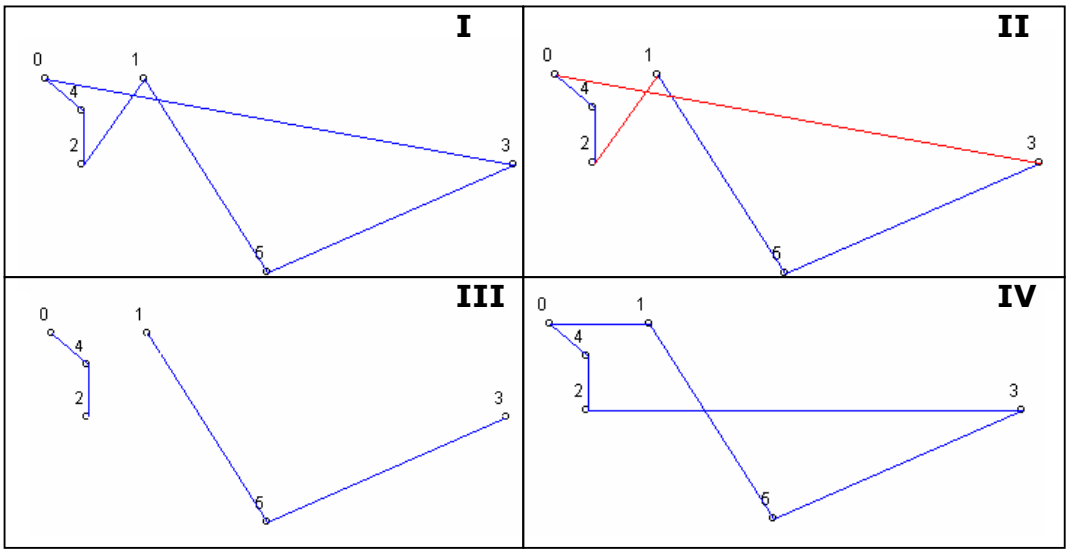

FIGURA 3 – APLICAÇÃO DA SIMULAÇÃO DO ALGORITMO 2-OPT

- passo interno de simulação: o terceiro botão representa a simulação de um passo mais interno da simulação, mostrando detalhes que estão escondidos quando da utilização da ação apresentada anteriormente (disponível somente para os algoritmos construtivos e de melhoramento). Para o algoritmo construtivo vizinho mais próximo, esta ação representa mostrar ao usuário os cálculos para cada nova inserção da rota. Desta forma, o aluno poderá perceber que é escolhido sempre o menor caminho entre a cidade atual e todas as demais pertencentes ao problema e que ainda não foram inseridas na rota. De forma semelhante, para o algoritmo de inserção mais distante, o objeto de aprendizagem apresentará, para cada nova inserção de uma cidade a sub-rota atual, os cálculos que correspondem a cada possibilidade de inserção, demonstrando qual é a mais distante. Em relação aos algoritmos de melhoramento, o passo interno de simulação apresenta todas as possibilidades de troca de arcos, mostrando os cálculos feitos e

apresentando a troca escolhida (se ela existir). Certamente, o passo interno de simulação é uma ação muito mais rica no que concerne aos dados exibidos para o usuário mas, de outra forma, ela também acresce complexidade à interpretação dos mesmos. A utilização com parcimônia de cada uma das ações disponíveis no objeto de aprendizagem é parte importante do processo de desenvolvimento das aulas.

A seção três do objeto de aprendizagem apresenta um resumo do histórico das simulações realizadas. Nele o usuário pode observar as principais informações a ações realizadas no ambiente, de tal forma a obter graus de comparação entre diversos algoritmos sem a necessidade de manter diversas janelas abertas ou recorrer a outros operacionalizadores tais como blocos de anotação. As ações armazenadas no histórico compreendem a escolha e execução de algoritmos (com seus respectivos valores finais) e a troca do problema objetivo.

Finalmente, na seção quatro se apresenta a simulação em si, onde as cidades são representadas graficamente como círculos e os arcos como linhas que unem os respectivos círculos. Um número que representa a cidade é apresentado ao lado da mesma, como forma de identificá-lo. A medida em que a simulação ocorre, os arcos são inseridos e removidos de acordo com os algoritmos escolhidos e o desenho do grafo é atualizado.

Adicionalmente, as meta-heurísticas apresentam outra interface, onde o andamento da simulação pode ser observado. Tal perspectiva foi utilizada pois estes algoritmos usualmente consomem muito tempo e podem ser necessárias milhares de testes para alcançar um resultado satisfatório. Desta forma, se optou por ilustrar o andamento da simulação através de gráficos (figura 4), ao contrário da utilização de passos internos de simulação.

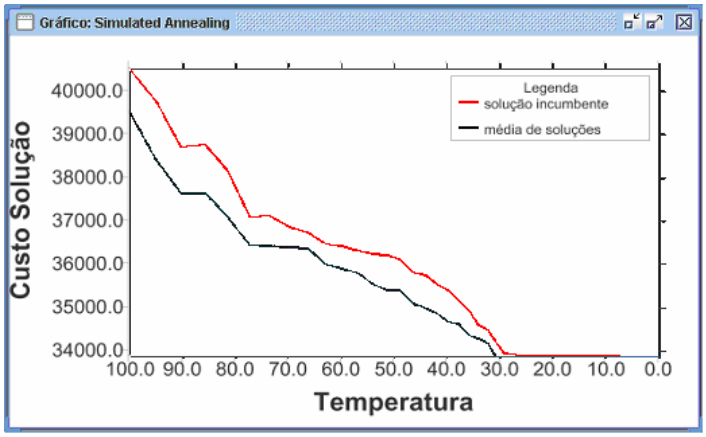

FIGURA 4 – GRÁFICO DA META-HEURÍSTICA SIMULATED ANNEALING

5. IMPLEMENTAÇÃO DE AULAS COM O OBJETO DE APRENDIZAGEM PROPOSTO

Como relatado anteriormente, uma dificuldade pertinente ao ensino de heurísticas e meta-heurísticas está na dissociação entre o caráter pedagógico da realização de pequenos exemplos desenvolvidos exaustivamente em aula com os resultados computacionais obtidos do mesmo. Se, ao mesmo tempo se torna uma escolha inicialmente promissora a realização completa e exaustiva de pequenos exemplos utilizando giz e quadro-negro, também se tornou bastante óbvio que esta abordagem pode conter armadilhas que dificultam o processo de constante motivação que os educandos necessitam para que as aulas se tornem produtivas.

Ao se utilizar pequenos exemplos em sala de aula, é muito provável que algoritmos pouco eficientes consigam atingir a otimalidade. Mesmo idéias construtivas bastante simples, como o algoritmo do vizinho mais próximo, podem conseguir excelentes resultados com problemas simples. Desta forma, se torna complicado explicar ao aluno que são necessárias técnicas e modelagens mais complexas para resolver o problema se os exemplos para os quais são apresentados são resolvidos rapidamente.

Este ponto se torna ainda mais complicado ao se trabalhar com algoritmos de melhoramento, pois se os algoritmos construtivos simples já encontram bons resultados, provavelmente haverá pouco o que fazer nas rotas na tentativa de otimizá-las. Novamente, o fracasso aparente destas técnicas, ao contrário de fornecer subsídios que prendam a atenção do aluno ou que os motive a tentar algo melhor ou diferente, acaba por agir como um desestimulador, pois *porque estudar, implementar e testar códigos tão difíceis se as soluções mais simples apresentam bons resultados?*

Esta pergunta, ouvida tantas vezes nas aulas de heurísticas e meta-heurísticas, é o grande motivador da implementação do objeto de aprendizagem apresentado neste artigo. Com a implementação visual dos algoritmos, é possível descrever suas particularidades em um nível mais alto, deixando os detalhes da implementação para discussões posteriores. Usualmente, se percebe que o grande problema na implementação destas metodologias está no entendimento destas e não na habilidade algorítmica e tecnológica dos estudantes, pois a área de heurísticas e meta-heurísticas necessita que os educandos se apropriem das nuances e diferenças entre os diversos algoritmos apresentados na literatura e tais propriedades só estão disponíveis ao se aplicar diversas vezes o mesmo algoritmo em diversos problemas. A realização de comparações de resultados e de técnicas está no cerne da compreensão da área.

Desta forma, o objeto de aprendizagem desenvolvido apresenta como sua principal contribuição a possibilidade de realizar diversos experimentos, compará-los e, essencialmente, *visualizá-los* de forma que facilite a compreensão. Ao apresentar as rotas sendo modificadas graficamente, os educandos podem perceber o comportamento dos algoritmos em consonância as expectativas apresentadas pelo professor(es) da disciplina, que vê, desta forma, seus prognósticos sobre a dificuldade em resolver *bem* o problema

confirmados através de um exemplo prático, o que difere da abordagem tradicional dos pequenos problemas com giz e quadro negro.

Como forma de explicitar a abordagem desenvolvida é apresentado a seguir a estrutura de uma aula, onde é abordado o desenvolvimento de algoritmos construtivos para o Problema do Caixeiro Viajante.

Algoritmos Construtivos para o PCV – Aula 1
- Ponto 1. Definição de Algoritmos Construtivos
- Ponto 2. Apresentação do Algoritmo de Vizinho mais Próximo
- Ponto 3. Apresentação de um exemplo para ser resolvido pelos alunos (poucas cidades)
- Ponto 4. Discussão sobre o resultado do Vizinho mais Próximo no problema escolhido
- Ponto 5. Simulação do problema apresentado no objeto de aprendizagem
- Ponto 6. Apresentação de outros problemas no objeto de aprendizagem
- Ponto 7. Discussão sobre o resultado do Vizinho mais Próximo nos problemas apresentados

Nesta primeira aula de algoritmos construtivos, os alunos são convidados a realizarem um teste de mesa com o algoritmo do vizinho mais próximo em um problema simples (poucas cidades), como ocorre normalmente em uma aula tradicional de heurísticas e meta-heurísticas. A diferença é que, após o debate inicial onde os resultados encontrados pelos alunos são debatidos e suas idéias sobre a qualidade do algoritmo são discorridas, o professor pode rebater a idéia, mesma que esta não tenha sido pronunciada em voz alta, de que um algoritmo simples como o Vizinho Mais Próximo pode resolver qualquer tipo de problema. Utilizando o objeto de aprendizagem com o problema tt48 da TSPLIB, fica fácil perceber que este algoritmo pode apresentar problemas de performance em problemas maiores. No ponto 6, ao rodar o problema citado anteriormente passo a passo, é fácil perceber – devido ao apelo visual – que nos passo finais do algoritmo, os arcos inseridos são muito grandes, pois há poucas cidades disponíveis para serem inseridas na rota final. A figura 5 apresenta três dos últimos passos do algoritmo em questão, para o problema tt48, onde esta situação é bastante nítida. As inserções dos arcos 44-1 e 1-7 representam o problema do algoritmo de forma bastante esclarecedora.

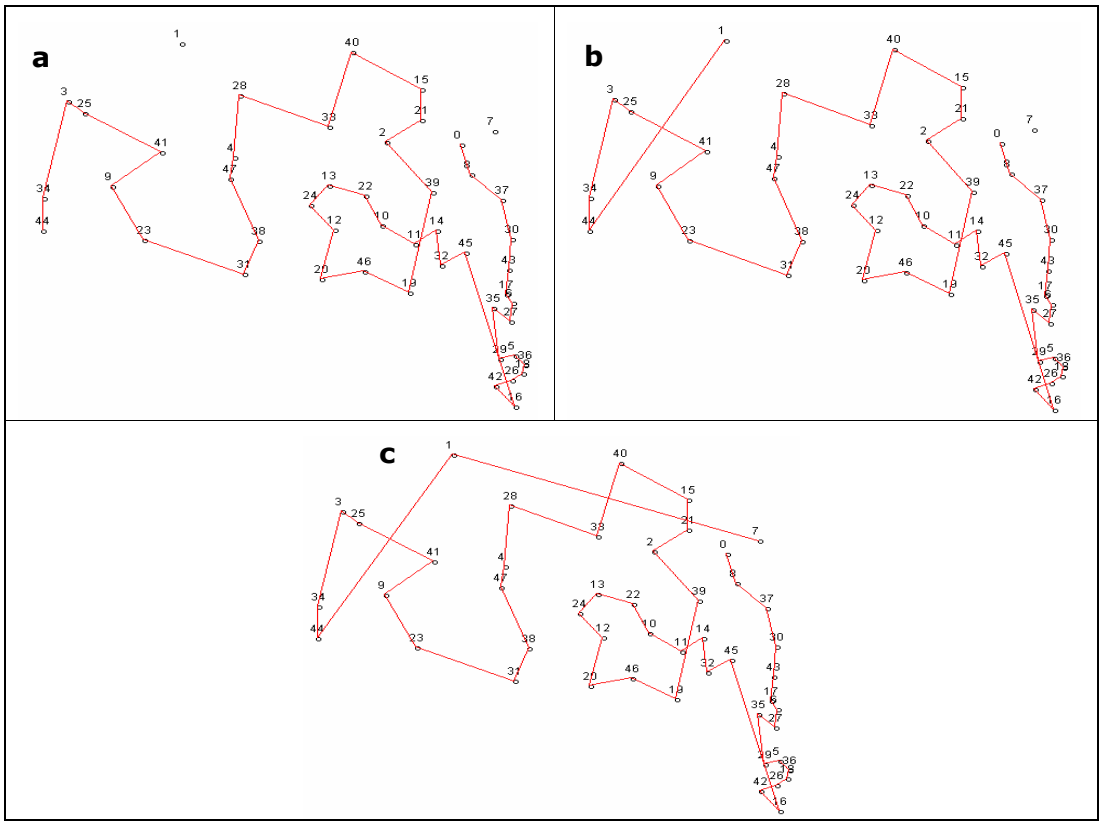

FIGURA 5 – CONSTRUÇÃO PASSO A PASSO DO ALGORITMO VIZINHO MAIS PRÓXIMO

5. CONCLUSÕES E PERSPECTIVAS FUTURAS

O ensino-aprendizagem de heurísticas e meta-heurísticas é, essencialmente, problematizador, o que torna perfeito o casamento entre os conteúdos e a educação dialógica-problematizadora. No entanto, não é a simples exposição de problemas que garante o efetivo aprendizado do educando nem a eficácia desta metodologia. Dentre as experiências realizadas, se nota, primeiramente, o grau de amadurecimento dos educandos envolvidos nesta metodologia. A realização de diversas experiências e o estudo de um problema único, onde todos estão envolvidos, desenvolveu uma massa crítica de conhecimento que se desaguava nas aulas presenciais. Comparativamente as experiências de educação tradicional – baseado no quadro e giz – os educandos se tornaram muito mais participativos, inquisitivos e propositivos.

Mesmo que nem todas as idéias apresentadas fossem coerentes ou produtivas, a simples menção das mesmas em aula comprova que os educandos estavam pensando no problema proposto, ao contrário de simplesmente estarem recebendo um conhecimento pronto para ser deglutido.

No entanto, é muito importante observar que alguns alunos não estão prontos para esta mudança, por medo ou acomodação, afinal, é mais fácil estudar alguns dias antes para uma dada prova formal do que desenvolver um trabalho mais complexo durante um semestre inteiro. Nestes casos, cabe ao professor dirigir o processo, explicando as vantagens da construção do conhecimento pelo fazer e pensar do que simplesmente pelo decorar. Não faltam exemplos na sociedade moderna, dita sociedade da informação, que corroboram a tese do que aprender a aprender é muito mais importante do que saber alguma técnica ou metodologia de cor. Por outro lado, as discussões em sala de aula podem se tornar abrangentes demais e, da mesma forma, cabe ao professor, dirigente do processo – apesar de não ser mais o único detentor do conhecimento – trazer novamente os alunos para o foco da disciplina. Desta forma, há espaço para discussão e debates de idéias novas e, potencialmente, profícuas, mas também há a necessidade de se consolidar um determinado conteúdo que faz parte de uma disciplina de graduação. Através do uso do objeto de aprendizagem com uma metodologia adequada, é possível melhorar o entendimento dos métodos matemáticos envolvidos na área de pesquisa operacional sem perder a profundidade necessária.

Um aspecto importante, tanto do objeto quanto da metodologia, é que o educador, ou grupo de educadores, continua sendo essencial para o desenvolvimento das aulas. Apesar de não ser o centralizador do conhecimento, pois ele já o partilha desde o início, deixando que os próprios alunos possam construí-los e desenvolvê-los a partir de suas idéias individuais e coletivas, o professor deve dirigir as aulas, conduzindo os alunos ao conteúdo que está sendo desenvolvido. Desta forma, há espaço para discussão e debates de idéias novas e, potencialmente, profícuas, mas também há a necessidade de se consolidar um determinado conteúdo que faz parte de uma disciplina de graduação. Através desta metodologia, é possível melhorar o entendimento dos métodos matemáticos envolvidos na área de pesquisa operacional sem perder a profundidade necessária.

Como trabalhos futuros pode se destacar a inserção de novos algoritmos aos já implementados no objeto de aprendizagem e a formalização de um plano de ensino completo contendo todas as aulas e construído utilizando como base a metodologia didática proposta e o objeto de aprendizagem desenvolvido.

6. BIBLIOGRAFIA

BELHOT, Renato V., FIGUEIREO, Reginaldo S., MALAVÉ, César O. “O Uso da Simulação no Ensino de Engenharia”, Congresso Brasileiro de Ensino de Engenharia, XXIX, Cobenge, 2001.

CORNE, D., DORIGO, M., GLOVER, F. “New Ideas in Optimization”, McGraw Hill, Berkshire, 1999.

DÁVALOS, R.V. “Uma Abordagem do Ensino de Pesquisa Operacional Baseada no Uso de Recursos Computacionais”, In: International Conference

On Engineering And Technology Education. 6. Proceedings... Santos. Brasil, 1999.

DE BASTOS, Fábio da Purificação, ALBERTI, Taís Fim, MAZZARDO, Mara Denize. “Ambientes Virtuais de Ensino-Aprendizagem: Os Desafios dos Novos Espaços de Ensinar e Aprender e suas Implicações no Contexto Escolar”, Revista Novas Tecnologias na Educação. V Ciclo de Palestras Novas Tecnologias na Educação, v. 3. n.1. mai. 2005.

DIAZ, A. “Optimización Heurística Y Redes Neuronables”, Editorial Paraninfo, Madrid, 1996.

DOWNES, S. “Learning Objects”, 2000**.** Disponível por www em: http://isit.arts.ubc.ca/ cdf/L_Objects.pdf

IEEE - Learning Technology Standards Committee (LTSC). “Draft Standard for Learning Object Metadata”, 2002. Disponível por www em http://ltsc.ieee.org/wg12/files/LOM_1484_12_1_v1_Final_Draft.pdf

IMS. “IMS Learner Information Package Specification”, 2005. Disponível por www em http://www.imsglobal.org/

IMS. “IMS Meta-data Best Practice Guide for IEEE 1484.12.1-2002 Standard for Learning Object Metadata”, 2006. Disponível por www em http://www.imsglobal.org/metadata/mdv1p3/imsmd_bestv1p3.html

IP, Albert; MORRISON, Iain. “Learning Objects in Different Pedagogical Paradigms”, Proceedings of the 18th Annual Conference of the Australian Society for Computers in Learning in Tertiary Education. Melbourne, 2001. Disponível por www em: http://www.medfac.unimelb.edu.au/ ascilite2001/pdf/papers/ ipa.pdf

MALLOY, Thomas E., HANLEY, Gerard, E. “MERLOT: A faculty-focused Web site of educational resources”, Behavior Research Methods, Instruments, & Computers. v. 33. n.2 p. 274-276. 2001.

MARTINS, A.J. et al. "Simulações On-Line no Ensino de Física e Química”, Revista Brasileira de Informática na Educação. Numero 7. Sbc. P.111-117. v. 11, n. 2. Julho/Dezembro, 2003.

MORAN, J.M. “Ensino e Aprendizagem Inovadores com Tecnologias Audiovisiuais e Telemáticas”, In: Novas Tecnologias Pedagógicas. Moran, Masetto, Behrens (Eds). Campinas, São Paulo, 2000.

NASCIMENTO, Anna Christina de Azevedo. “Construindo Comunidades de Elaboradores de Objetos de Aprendizagem através de Conteúdo, Tutoria e Interação dos Pares”, Secretaria de Educação a Distância, Ministério da Educação, dezembro 2005. Disponível por www em: http://rived.proinfo.mec.gov.br/artigos/Anna20063.pdf

POLSANI, Pithamber R. “Use and Abuse of Reusable Learning Objects”, In: Journal of Digital Information, v. 3, n. 4. article n 164, 2003.

POPPER, K. “Verdade, racionalidade e a expansão do conhecimento científico”, In: Popper, K, Conjecturas e refutações, Coleção Pensamento Científico, Ed. Universidade de Brasília, 1983.

SANTANCHÈ, André. “Anima: Representando e Integrando Objetos Educacionais na *Web*”, Dissertação de Mestrado**.** Salvador: UNIFACS/Programa de Pós-Graduação em Redes de Computadores, 2002.

SCORM, "SCORM 2004 3rd Edition Documentation Suite", 2006. Disponível por www em http://www.adlnet.gov/downloads/files/311.cfm

SOUZA, Marcia Izabel Fugisawa; VENDRUSCOLO, Laurimar Gonçalves; MELO, Geane Cristina. “Metadados para a descrição de recursos de informação eletrônica: utilização do padrão Dublin Core”, Revista Ciência da Informação. v.29 n.1. Brasília jan./abr. 2000.

TAROUCO, Liane Margarida Rockenbach; FABRE, Marie-Christine Julie Mascarenhas; TAMUSIUNAS, Fabrício Raupp. “Reusabilidade de objetos educacionais”, RENOTE - Revista Novas Tecnologias na Educação. Porto Alegre: Centro Interdisicplinar de Novas Tecnologias na Educação (UFRGS), v. 1, n. 1, 2003.

ZACHARY, J.L. “Introduction to Scientific Programming Computational Problem Solving Using: Maple And C”, University Of Utah. Animations Available from http://www.cs.utah.edu/~zachary/isp/applets/cannon/cannon.html, 1999.