CORDENONSI, A. Z. ; MÜLLER, Felipe Martins ; BASTOS, Fábio da Purificação de ; ARAÚJO, Fabrício Viero . *Objetos de Aprendizagem para o Ensino de Heurísticas e Meta-Heurísticas: Uma Abordagem Baseada em um Ambiente de Aprendizagem e na Educação Dialógica Problematizadora.* In: XXXVIII Simpósio Brasileiro de Pesquisa Operacional, 2006, Goiânia - GO. Anais

# Objetos de Aprendizagem para o Ensino de Heurísticas e Meta-Heurísticas: Uma Abordagem Baseada em um Ambiente de Aprendizagem e na Educação Dialógica Problematizadora

Andre Zanki Cordenonsi [1]
Felipe Martins Muller [2]
Fábio da Purificação de Bastos [3]
Fabrício Viero Araújo [4]

[1] {andrezc@inf.ufsm.br} Universidade Federal de Santa Maria, Departamento de Documentação
[2] {felipe@inf.ufsm.br} Universidade Federal de Santa Maria, Departamento de Eletrônica e Computação
[3] {fbastos@ce.ufsm.br} Universidade Federal de Santa Maria, Departamento de Educação
[4] {vieros@terra.com.br} Universidade Federal de Santa Maria, Programa de Pós-Graduação em Engenharia da Produção

## RESUMO

O ensino de heurísticas e metaheurísticas na área de pesquisa operacional está centrado, basicamente, na definição de modelos matemáticos e algoritmos para o teste de teorias que são apresentadas em sala de aula. Este modelo tradicional de ensino pressupõe o professor como principal mentor do processo de transferência da teoria matemática para uma definição lógica de passos algorítmicos que os educandos devem entender e, posteriormente, implementar em alguma linguagem de programação. Neste artigo, propomos um objeto de aprendizagem que auxilie o professor no desenvolvimento de suas aulas, apresentando um sistema de simulação de interface simples e consistente.

**PALAVRAS-CHAVE:** ensino de heurísticas e metaheurísticas, objetos de aprendizagem, AMEM

## ABSTRACT

The heuristics and metaheristics teaching process is centered in a set of mathematical models definitions and algorithms, which are presented in classrooms. This traditional model of education estimates the professor as a main mentor of the transference process of the mathematical theory to define algorithmical logical steps. The studens must understand the algorithms and implement it in some programming language. In this paper, we present a learning object to assist the development of the classes by the teacher, using a simple and effective interface simulation system.

**KEYWORDS:** heuristics and metaheuristics teaching, learning objects, AMEM

## 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. Conforme (Corne e outros, 1999), 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.

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*. O ensino de heurísticas e meta-heurísticas, que usualmente apresenta um embasamento matemático forte e necessita um grande número de passos para que o processo faça sentido, torna-se desestimulante se realizado através de um processo didático tradicional. Também é importante salientar que a literatura é pobre no que concerne ao estudo didático de tais conteúdos, concentrando-se normalmente na definição e comparação de algoritmos e suas complexidades.

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). 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 e outros, 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.

Desta forma, o objetivo deste artigo é apresentar um objeto de aprendizagem que auxilie o educador nas disciplinas de heurísticas e meta-heurísticas, fornecendo uma ferramenta de fácil manipulação e útil no que concerne a explanação dos diversos algoritmos apresentados aos alunos.

## 2. Concepção Pedagógica

A Educação Dialógica-Problematizadora interpreta que a ação educacional pode ser construída metodologicamente, conforme ANGOTTI e DELIZOICOV(1990), através dos seguintes momentos: problematização inicial, organização do conhecimento e aplicação do conhecimento. A problematização inicial é apresentada como um desafio na forma de questões e/ou situações que

devem ser discutidas e debatidas pelos alunos. No próximo momento, na organização do conhecimento, o educador deve orientar a sistematização do mesmo, para a completa compreensão do tema e do problema apresentado. Este conhecimento deverá ser usado pelo aluno para analisar o problema inicial e generalizá-lo para outras situações que são explicadas pelo mesmo conhecimento, o que caracteriza o terceiro momento, da aplicação do conhecimento.

Durante a concepção do ambiente, a teoria educacional Dialógica-Problematizadora (DE BASTOS e MULLER, 1999) foi reorganizada pelo grupo de pesquisa e apresentada em três momentos: desafio inicial, melhor solução educacional no momento e desafio mais amplo. Inicialmente, o modelo é ativado no início da aula com um *desafio inicial*, que instiga os alunos a investigarem suas próprias visões de mundo através de um desafio concreto e/ou um problema a ser resolvido. Através de múltiplas interações entre aluno/educador e aluno/aluno, deve-se consolidar o conhecimento científico através da codificação/decodificação da *melhor solução educacional no momento*, sistematizada pelo educador. Desta forma, o educador pode confrontar as visões de mundo dos alunos, tencionando as visões científicas e cotidianas, rompendo-as e problematizando o *desafio mais amplo*, "que busca avaliar processualmente a universalidade, validade e limitação do conhecimento científico-tecnológico abordado na aula." (DE BASTOS e MULLER, 1999). Deste modo, o projeto AMEM procura consolidar a problematização dos conteúdos escolares, desafiando educadores e alunos a trabalhar na escola com a perspectiva da resolução de problemas científico-tecnológicos, sendo estruturado nos módulos desenvolvidos que são apresentados no próximo capítulo.

## 3. Ambiente de Aprendizagem: AMEM

O AMEM (Ambiente Multimídia para Educação Mediada por Computador) foi concebido e construído baseado solidamente em uma concepção metodológica clara e objetiva, onde o professor é levado a adotar uma prática pedagógica que retrate a teoria investigação-ação fundamentadora da definição do sistema. Desta forma, os docentes imbuídos da necessidade de estudar suas próprias práticas através deste marco teórico reconhecem o ambiente como uma ferramenta única e inovadora para suas próprias ações. É importante salientar que, ao contrário de outros ambientes educacionais tradicionais, o AMEM não é apenas uma coleção de ferramentas disponíveis. Através dos operacionalizadores pedagógicos que podem ser escolhidos (programação, atividade extraclasse, atividade de colaboração), o professor pode organizar sua prática docente através da metodologia pedagógica proposta.

Na nova versão do AMEM os usuários podem assumir três perfis diferentes: o de *professor,* o de *aluno*, centros do processo de ensino-aprendizagem e o de *administrador*, encarregado da manutenção do ambiente. Existem seis módulos principais: meus dados (onde o usuário tem acesso às informações de caráter pessoal); comunicação (ferramentas para comunicação síncrona e assíncrona); disciplinas (acesso às suas disciplinas e suas atividades escolares); biblioteca (depósito dos materiais escolares); ajuda (manual de uso das ferramentas do ambiente); e sair (finalização das atividades no ambiente). Desta forma, o ambiente possibilita a criação, participação e administração de cursos baseados na *internet* de forma operacional, sem exigir do docente um amplo conhecimento das tecnologias envolvidas para disponibilizar um curso interativo através da *Internet*.

Como operacionalizadores pedagógicos, foram desenvolvidos três subsistemas: *aulas*, *atividades* e *colaborações.* A aula representa a estrutura cronológica dos *pontos* que serão desenvolvidos pelo docente, tendo associada a ela uma bibliografia disponível. Para cada ponto, é possível associar *objetos de aprendizagem* previamente cadastrados no sistema. Já as *atividades* são apresentadas na forma de problemas a serem resolvidas, sendo possível também associar bibliografias ou objetos à atividade. Finalmente, a *colaboração* é usada pelo professor para possibilitar ao aluno participação ativa do processo de desenvolvimento da próxima aula

(disponibilização de bibliografia, pré-programação, recebimento de sugestões,etc.). A versão 2.0 do ambiente apresenta uma interface mais limpa e fácil de navegar. Uma visão geral do sistema pode ser visualizada na figura 1.

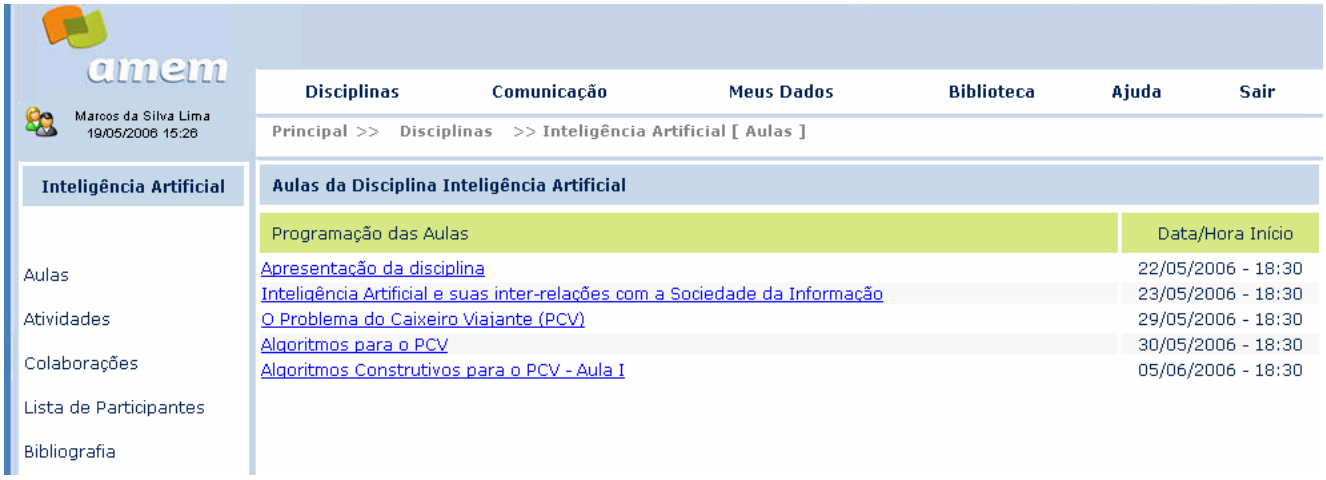


**Figura 1.** Visão geral do AMEM – 2.0

Utilizando o ambiente AMEM como suporte para as aulas, foi desenvolvido um objeto de aprendizagem que foi anexado ao sistema com o objetivo de auxiliar as atividades docentes do professor, fornecendo uma ferramenta de fácil utilização e que trouxesse real significado aos alunos no que concerne ao entendimento dos algoritmos expostos em aula. Este objeto e suas características são apresentados no próximo capítulo.

## 4. Objeto de Aprendizagem: 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 todos as 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. 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.

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 2 apresenta a interface principal do objeto que será descrito a seguir.

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) e dois de melhoramento (heurística 2-OPT e 3-OPT). 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. A figura três demonstra o fluxograma do controle do objeto, que será detalhado a seguir. Inicialmente, o usuário deverá escolher o tipo de algoritmo que será simulado (*construtivo* ou *melhoramento*). 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. Este parâmetro pode variar de um a N(todos os vizinhos);
    - 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 2) 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. Conforme é possível perceber pelo fluxo de controle (figura 3), após a apresentação do resultado, o usuário pode escolher novamente suas opções de configuração e efetuar uma nova simulação ou sair do sistema;

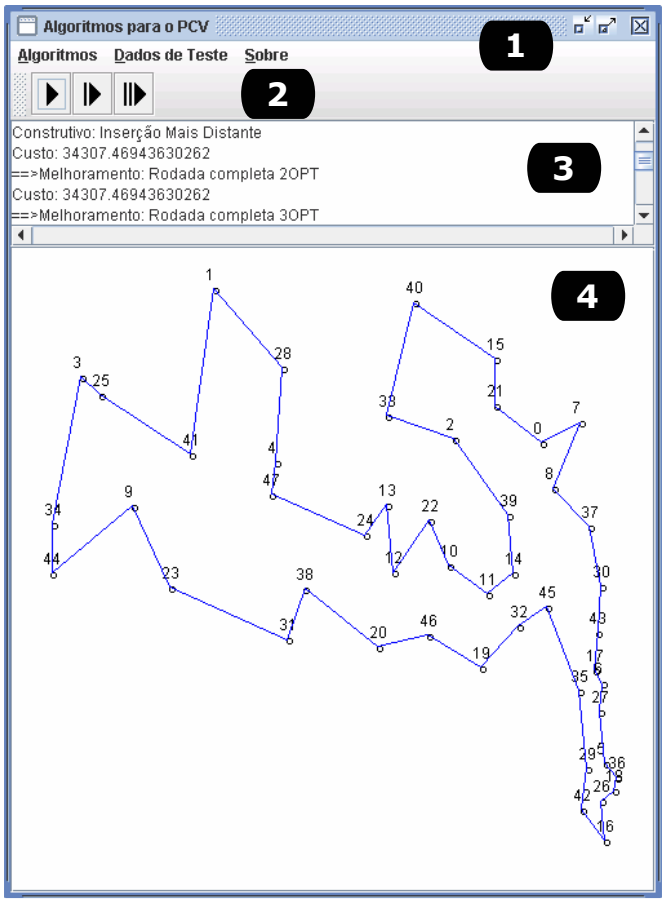


**Figura 2** – Interface principal do Objeto de Aprendizagem

- passo de simulação: o segundo botão efetua um passo completo de simulação. 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. Em relação aos demais algoritmos construtivos, o aleatório não apresenta a característica de *passo de simulação* pois não há objetivo em simular passo a passo uma construção puramente aleatória; e a inserção mais distante apresenta a rota inicial e as sub-rotas sendo construídas até que a rota final com todas as cidades seja encontrada. 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 3-OPT 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. 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). Neste caso, a janela de resultados é apresentada ao usuário que poderá novamente escolher, possivelmente, um novo algoritmo ou novos dados;

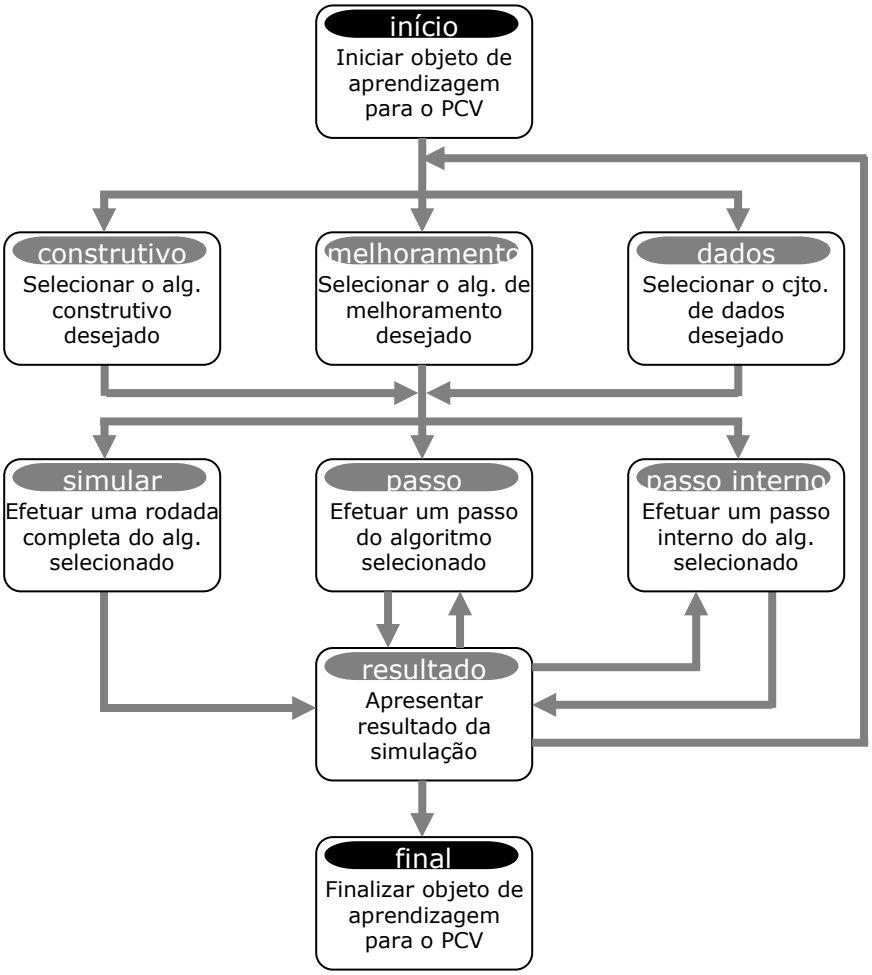


**Figura 3** – Fluxo de controle do objeto de aprendizagem

- 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. 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, como demonstrado no capítulo 6.

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. A figura 4 representa uma possível visualização do histórico de utilização do objeto de aprendizagem.

```
Histórico de resultados:
Dados: tt48.tsp [TSPLIB]
Troca de dados: pr76.tsp [TSPLIB]
Construtivo: Vizinho mais próximo
Custo: 153461.92250215414
==>Melhoramento: 20 Rodadas completas 2OPT
Custo: 114285.31985734106
==>Melhoramento: 02 Rodadas completas 3OPT
Custo: 111237.63037209141
==>Melhoramento: 06 Rodadas completas 2OPT
Custo: 110887.0876500589
==>Melhoramento: 01 Rodadas completas 3OPT
Custo: 110887.0876500589
```

**Figura 4** – Visualização do Histórico das Simulações

Finalmente, na seção três 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. Como última característica, é interessante notar que esta seção foi implementada como uma câmera sintética, onde os pontos e suas distâncias euclidianas são representadas em função do tamanho da janela. Caso o usuário aumente ou diminua a interface da aplicação, os pontos serão redimensionados para as novas proporções. Esta característica auxilia na visualização do objeto de aprendizagem. Problemas maiores podem ser vislumbrados em um espaço maior e problemas com um menor número de cidades podem ser representados em um espaço de interface bem menor.

## 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 em modelos maiores consigam atingir a otimalidade. Mesmo idéias construtivas bastante simples como o algoritmo do vizinho mais próximo pode 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 abordada 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 6 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.

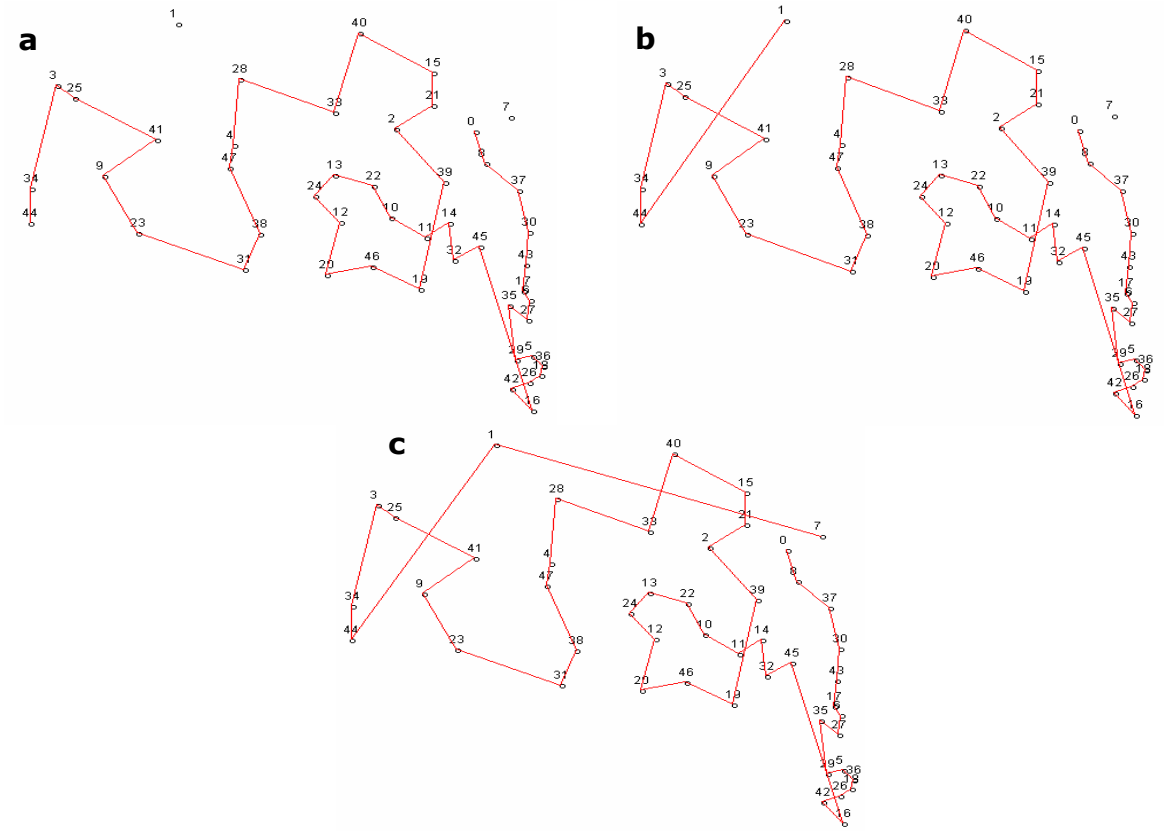


**Figura 6** – Construção passo a passo do Algoritmo Vizinho Mais Próximo

## 6. 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-problematizador. 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 desta metodologia, é possível melhorar o entendimento dos métodos matemáticos envolvidos na área de pesquisa operacional sem perder a profundidade necessária. Um aspecto importante, tanto do ambiente 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, construído como base a metodologia didática proposta e o objeto de aprendizagem desenvolvido.

## 7. Bibliografia

ANGOTTI, J. A. P. et al. **Metodologia do Ensino de Ciências**. São Paulo: Cortez, 1990.

DE BASTOS, F.P., MULLER, F.M. **Criando Desafios em Informática.** In: Atas Da Iv Escola De Verão Sobre Investigação-Ação Educacional. UFSM, Santa Maria, RS, 1999.CORNE, D., et al. **New Ideas in Optimization**. Mcgraw Hill, Berkshire, 1999.

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

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.

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.

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

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.