

UNIVERSIDADE NOVE DE JULHO - UNINOVE
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO

WILSON DA SILVA LOURENÇO

**OBJETO DE APRENDIZAGEM PARA O ENSINO DE ALGORITMOS
SOLUCIONADORES DE PROBLEMAS DE OTIMIZAÇÃO EM REDES**

SÃO PAULO

2015

WILSON DA SILVA LOURENÇO

**OBJETO DE APRENDIZAGEM PARA O ENSINO DE ALGORITMOS
SOLUCIONADORES DE PROBLEMAS DE OTIMIZAÇÃO EM REDES**

Dissertação de mestrado apresentada ao Programa de Pós Graduação em Engenharia de Produção da Universidade Nove de Julho – UNINOVE, como requisito parcial para obtenção do grau de Mestre em Engenharia de Produção.

Prof. Sidnei Alves de Araújo, Dr. - Orientador

SÃO PAULO

2015

**OBJETO DE APRENDIZAGEM PARA O ENSINO DE ALGORITMOS
SOLUCIONADORES DE PROBLEMAS DE OTIMIZAÇÃO EM REDES**

Por

WILSON DA SILVA LOURENÇO

Dissertação de mestrado apresentada ao Programa de Pós Graduação em Engenharia de Produção da Universidade Nove de Julho – UNINOVE, como requisito parcial para obtenção do grau de Mestre em Engenharia de Produção.

Presidente: Prof. Dr. Sidnei Alves Araújo - Orientador, UNINOVE

Membro interno: Prof. Dr. Marcos Antonio Gaspar

Membro interno: Prof. Dr. Ivanir Costa

Membro externo: Prof. Dr. Rodrigo Franco Gonçalves

SÃO PAULO

2015

Só existem dois dias no ano em que nada pode ser feito. Um se chama ontem e outro se chama amanhã, portanto, hoje é o dia certo para amar, acreditar, fazer e principalmente viver.”

(Dalai Lama)

AGRADECIMENTOS

Agradeço primeiramente a Deus, pois sem ele nada é possível e a UNINOVE por todo apoio financeiro. A minha mãe, D. Arlinda, por sempre me incentivar a nunca desistir daquilo que começamos. À minha esposa Priscilla Cunha, pela paciência, ajuda e compreensão durante os períodos de ausência. Ao meu orientador Prof. Dr. Sidnei Alves de Araújo pelo apoio, paciência e companheirismo empregado ao longo desta jornada. Ao Prof. Ms. Wonder Alexandre Luz Alves pela ajuda em questões técnicas na construção do objeto de aprendizagem e a Prof.(a). Ms. Ana Carolina Caldas Dias Jordan por ter me incentivado a iniciar este desafio.

RESUMO

Os problemas de otimização em redes (POR) são comuns a diversas áreas como engenharia, transportes e telecomunicações, e têm sido objetos de intensas pesquisas e estudos. Entre os POR clássicos estão os problemas de Caminho Mínimo (PCM), Fluxo Máximo (PFM) e Caixeiro Viajante (PCV), os quais normalmente são estudados em cursos de graduação e pós-graduação tais como Engenharia de Produção, Ciência da Computação, Sistemas de Informação e Logística, com a utilização de recursos como giz e lousa, o que dificulta o trabalho do professor, no sentido de mostrar o funcionamento dos algoritmos que solucionam esses problemas, mantendo a motivação dos alunos para a aprendizagem. Neste contexto, propõe-se nesta pesquisa, uma ferramenta computacional, caracterizada como um Objeto de Aprendizagem (OA) denominado TASNOP - *Teaching Algorithms for Solving Network Optimization Problems*, cuja finalidade é contribuir para compreensão dos alunos sobre conceitos de POR e, principalmente, sobre o funcionamento dos algoritmos A*, Busca Gulosa, e Dijkstra, usados para resolução do PCM, Ford-Fulkerson empregado na resolução de PFM e o algoritmo Vizinho mais Próximo para resolução do PCV. É importante ressaltar que o OA proposto pode ser acessado via *web* e, inclusive, ser acoplado em ambientes de ensino a distância (EaD). Experimentos realizados no ano de 2014 envolvendo 129 alunos do curso de Ciência da Computação, dos quais 51 resolveram um exercício com o uso do TASNOP e 78 sem o seu uso, permitiram verificar que os alunos que utilizaram o TASNOP obtiveram melhor desempenho na resolução do exercício proposto, corroborando a ideia de que o OA contribuiu para melhorar suas compreensões acerca dos algoritmos abordados nesta pesquisa. Em adição, os 51 alunos que usaram o TASNOP responderam a um questionário sobre o seu uso e, com base nessas respostas, ficou evidente o potencial do TASNOP como uma ferramenta de apoio ao ensino.

Palavras-chave: Problemas de Otimização em Redes. Caminho Mínimo. Fluxo Máximo. Caixeiro Viajante. Objeto de Aprendizagem.

ABSTRACT

The network optimization problems (NOP) are common to several areas such as engineering, transport and telecommunications, and have been objects of intense research and studies. Among the classical NOP are the problems of Shortest Path (SPP), Max Flow (MFP) and Traveling Salesman (TSP), which are usually studied in undergraduate and graduate courses such as Industrial Engineering, Computer Science, Information Systems and Logistics, with the use of resources such as chalk and blackboard that hinder the teacher's work, in the sense of showing the functioning of algorithms that solve these problems while maintaining students' motivation for learning. In this context, it is proposed in this research, a computational tool, characterized as a Learning Object (OA) and called TASNOP - Teaching Algorithms for Solving Network Optimization Problems, whose purpose is to contribute to students' understanding about concepts from NOP and, mainly, the functioning of algorithms A*, Greedy Search and Dijkstra used for resolution of SPP, Ford-Fulkerson employed in the resolution of MFP and the Nearest Neighbor to solve the TSP. It is important to highlight that the proposed OA can be accessed through web and also employed in distance learning environments (DLE). Experiments conducted in 2014 with 129 students of Computer Science, from which 51 performed an exercise using the TASNOP and 78 without this tool, confirm that students who used the TASNOP performed better in solving the proposed exercise, corroborating the idea that the OA helped to improve their understanding about the algorithms discussed in this research. In addition, the 51 students who employed the TASNOP answered a questionnaire about its use and, the answers indicated that the TASNOP shows a potential to be used as a learning support tool.

Keywords: Network Optimization Problems, Shortest Path, Max Flow, Traveling Salesman, Learning Object.

LISTA DE ILUSTRAÇÕES

Figura 1 – Exercício aplicado e questionário sobre o TASNOP	40
Figura 2 – Escala de Likert de 5 pontos	44
Figura 3 – Interface do TASNOP	47
Figura 4 – Tela do TASNOP com a solução do exercício empregado na sua avaliação	49
Figura 5 – Gráfico das respostas para o item “Contribuição para a aprendizagem dos algoritmos”	54
Figura 6 - Gráfico das respostas para o item “Usabilidade”	52
Figura 7 - Gráfico das respostas para o item “Potencial para ser um objeto de aprendizagem”	55
Figura 8 - Gráfico das respostas para o item “Confiabilidade dos resultados”	56
Figura 9 - Gráfico das respostas para o item “Interface”	53
Figura 10 - Sumarização dos resultados	57
Quadro 1 – Tipos de Objetos de Aprendizagem	24
Quadro 2 – Benefícios dos objetos de aprendizagem	24
Quadro 3 – Compilação dos POR abordados e seus algoritmos solucionadores	34
Quadro 4 – Conjunto de metadados do TASNOP	48

LISTA DE TABELAS

Tabela 1 – Usabilidade do TASNOP	52
Tabela 2 – Interface do TASNOP	53
Tabela 3 - Tabulação das respostas para o item Contribuição para a aprendizagem dos algoritmos	54
Tabela 4 – Potencial para ser um objeto de aprendizagem	55
Tabela 5 – Confiabilidade dos resultados apresentados pelo TASNOP	56

LISTA DE ABREVIATURAS E SIGLAS

A*	Algoritmo A estrela
API	Interface de programação de aplicações
AVA	Ambiente virtual de aprendizagem
BIOE	Banco Internacional de Objetos Educacionais
EaD	Educação a distância
LOM	<i>Learning Object Metadata</i>
OA	Objeto de aprendizagem
PFM	Problema do fluxo máximo
PCM	Problema de caminho mínimo
PCV	Problema do caixeiro viajante
POR	Problemas de Otimização em Redes
PPC	Potencial pedagógico computacional
TASNOP	<i>Teaching Algorithms for Solving Network Optimization Problems</i>
TI	Tecnologia da informação
TIC	Tecnologia da informação e comunicação

LISTA DE SÍMBOLOS

A	conjunto de arestas
α	conjunto de ações que permitem gerar estados sucessores
c	aresta com menor capacidade
g	função de custo de caminho
h	função heurística
G	grafo
\bar{O}	conjunto de estados finais
k	número de nós
n	nó
N	conjunto de nós
P	percurso de um grafo
Γ	conjunto de estados expandidos (visitados)
s	estado corrente
S	espaço de estados
s_0	estado inicial
Σ	conjunto de estados sucessores ainda não visitados
u	estado sucessor
U	conjunto de estados sucessores

SUMÁRIO

1. INTRODUÇÃO	14
1.1 PROBLEMA DE PESQUISA	15
1.2 HIPÓTESE	Erro! Indicador não definido.
1.3 OBJETIVO	16
1.4 JUSTIFICATIVA	16
1.5 ESTRUTURA DA DISSERTAÇÃO.....	18
2. FUNDAMENTAÇÃO TEÓRICA	19
2.1 AMBIENTES DE APRENDIZAGEM ON-LINE	19
2.1.1 Objeto de Aprendizagem (OA).....	21
2.2 PROBLEMAS DE OTIMIZAÇÃO EM REDES (POR).....	25
2.2.1 Representação de Problemas em Redes	25
2.2.2 Problema de Caminho Mínimo (PCM)	27
2.2.2.1 Algoritmo de Dijkstra	28
2.2.2.2 Algoritmo de Busca Gulosa (<i>Greedy Search</i>)	29
2.2.2.3 Algoritmo A*	30
2.2.3 Problema do Fluxo Máximo	31
2.2.3.1 Algoritmo Ford-Fulkerson.....	31
2.2.4 Problema do Caixeiro Viajante	32
2.2.4.1 Algoritmo do Vizinho Mais Próximo.....	33
2.3 JUNG (<i>Java Universal Network Graph</i>).....	35
2.4 TRABALHOS CORRELATOS.....	35
3. MATERIAIS E MÉTODOS	37
3.1 MÉTODO DE PESQUISA	37
3.2 MATERIAIS EMPREGADOS	39
3.3 METODOLOGIA PARA A CONDUÇÃO DOS EXPERIMENTOS	42

4. RESULTADOS.....	46
4.1 DETALHAMENTO DO OBJETO DE APRENDIZAGEM PROPOSTO	46
4.2 EXPERIMENTOS REALIZADOS PARA AVALIAÇÃO DO OA PROPOSTO.....	50
4.2.1 Usabilidade	51
4.2.2 Interface.....	52
4.2.3 Contribuição para aprendizagem dos algoritmos.....	53
4.2.4 Potencial para uso como um objeto de aprendizagem	54
4.2.5 Confiabilidade dos resultados.....	55
5. CONCLUSÕES	59
REFERÊNCIAS BIBLIOGRÁFICAS.....	60
GLOSSÁRIO.....	67
APÊNDICE A – TRABALHOS RESULTANTES DAS PESQUISAS REALIZADAS DURANTE O MESTRADO.....	68

1. INTRODUÇÃO

Quando se fala em “redes”, logo se pensa em redes físicas (redes telefônicas, elétricas, rodoviárias, ferroviárias, rede de esgotos, TV a cabo, informática, etc.), pois são as que melhor identificam esta estrutura. Em todos esses casos pretendemos enviar algo de um ponto para outro em uma rede específica e da forma mais eficiente possível, tanto para fornecer um bom serviço, como para utilizar um meio de transmissão (normalmente dispendiosa) de forma fácil e eficaz (AHUJA; MAGNANTI; ORLIN, 1993).

Em redes de computadores, por exemplo, toda informação transferida tem diversos caminhos possíveis e normalmente tem percorrer o menor deles. De acordo com Ballou (2003), na área de logística e transportes temos que entregar produtos com o melhor custo/benefício, e para minimizar esse custo e maximizar o benefício temos que minimizar as distâncias. Como se pode observar, tanto no caso das redes de computadores quanto no problema de transporte, o problema a ser resolvido pode ser caracterizado como o problema do caminho mínimo.

Conforme Sevastian, Rus e Dobrota (2012), a determinação do fluxo máximo é um problema de otimização em redes e tem sido estudado em diversas áreas por muitos anos. A grosso modo, o problema é calcular a quantidade máxima de "coisas" que podem ser movidas de um ponto a outro em uma rede, de acordo com as limitações na capacidade das arestas (meios que ligam os pontos). Estas "coisas" podem ser pacotes de dados que trafegam pela Internet, a água através de tubos ou alguns caminhões viajando nas rodovias. Assim, as limitações podem referir-se a largura de banda (para Internet), às dimensões do tubo (para os sistemas de distribuição de água) ou à velocidade média do tráfego (por estradas congestionadas).

O Problema do Caixeiro Viajante, por sua vez, se estende a muitos problemas de roteirização, sendo um bom exemplo a tarefa de um carteiro na entrega das cartas, a qual consiste, basicamente, em sair do posto do correio, percorrer todos os pontos de entrega uma única vez e, finalmente, retornar ao posto do correio. É claro que o ideal é realizar esta tarefa no menor tempo possível.

Conforme Welzl e Ali (2006), algoritmos para resolução de Problemas de Otimização em redes (POR) geralmente são representados por grafos, e amplamente estudados

em diversas áreas como, por exemplo, computação, transporte e logística, telecomunicações, entre outras.

Cabe ressaltar que tais algoritmos, assim como muitos outros conteúdos ensinados em cursos de graduação, são normalmente ensinados usando apenas recursos de giz e lousa, dificultando o trabalho do professor no sentido de mostrar repetidas vezes o funcionamento desses algoritmos e de manter a motivação dos alunos para a aprendizagem (MELO;BARROS;MORAES,1997). Com o advento de técnicas gráficas expositivas de ensino e do atual acesso aos recursos da Tecnologia da Informação e Comunicação (TIC), é importante que as instituições de ensino e os professores incorporem diferentes abordagens ao processo de aprendizagem (SILVA; MARTINS; NIGRO,2010).

De acordo com as colocações dos autores citados acima, fica clara a importância de ensinar algoritmos de resolução de POR para alunos de cursos de graduação e pós-graduação tais como Engenharia de Produção, Ciência da Computação, Sistemas de Informação e Logística, entre outros.

Nesse contexto, vislumbra-se que um Objeto de Aprendizagem (OA) pode ser empregado com um importante instrumento para auxiliar e estimular a compreensão dos algoritmos para resolução de POR considerados neste trabalho.

E importante salientar que mesmo utilizando um OA, não se deve dispensar a leitura de textos e outros materiais fornecidos pelo professor, pois a leitura é extremamente importante para o desenvolvimento do aluno (PAULA; PIVA; FREITAS, 2009).

1.1 PROBLEMA DE PESQUISA

Como desenvolver uma ferramenta computacional para auxiliar na aprendizagem de algoritmos para resolução de POR? Que recursos computacionais deve possuir esta ferramenta e como ela pode ser classificada, levando em conta as definições empregadas na literatura para caracterização dos ambientes online voltados para o ensino?

1.2 OBJETIVO

Essa pesquisa teve como objetivo principal o desenvolvimento de um objeto de aprendizagem (OA), denominado TASNOP – *Teaching Algorithms for Solving Network Optimization Problems*, para auxiliar na compreensão dos alunos sobre o funcionamento de algoritmos para resolução de POR, entre os quais Caminho Mínimo, Fluxo Máximo e Caixeiro Viajante. Esse objeto de aprendizagem consiste em uma ferramenta computacional dotada de uma *interface* gráfica, que permite ao aluno representar, resolver o problema investigado e desenvolver sua compreensão sobre cada um dos algoritmos disponíveis.

1.3 JUSTIFICATIVA

Com o advento das TICs (Tecnologia da Informação e Comunicação), os alunos estão cada vez mais acostumados a usar ambientes computacionais (*softwares*, *sites*, objetos de aprendizagem e ambientes virtuais de aprendizagem, entre outros) em seu dia a dia, e buscam isso quando precisam aprender novos assuntos (MCGREAL, 2004).

Conforme Werthein (2000) menciona, as transformações em direção à sociedade da informação, em estágio avançado nos países industrializados, constituem uma tendência dominante mesmo para economias menos industrializadas e definem um novo paradigma, o da tecnologia da informação (TI), que expressa a essência da presente transformação tecnológica em suas relações com a economia e a sociedade.

Com o acesso democratizado, global e total à informação e ao conhecimento por meio dos meios de comunicação e equipamentos eletrônicos, surge uma nova sociedade chamada de Sociedade da Informação (ALARCÃO, 2003).

Como citam Tigre e Noronha (2013), a computação em nuvem (*cloud computing*) simboliza a tendência de colocar toda a infraestrutura e informação disponível de forma digital na internet, incluindo *software* aplicativo, ferramentas de busca, redes de comunicação, provedores, centros de armazenamento e processamento de

dados. O conceito de nuvem é muito importante porque permite que a computação se transforme em uma utilidade pública, pois os bens da informação são não rivais e podem ser utilizados simultaneamente por ilimitados usuários. Sua principal vantagem é a possibilidade de utilizar os recursos de *hardware* e *software* disponíveis de forma mais eficiente, permitindo reduzir a capacidade ociosa em armazenamento e processamento de dados, por meio do compartilhamento de computadores e servidores interligados pela internet. A infraestrutura é acessada por terminais e dispositivos móveis que conectam a nuvem ao ser humano.

O mundo mudou a forma como a informação é produzida e o conhecimento é construído. As velocidades mudaram, o acesso mudou, a internet permitiu as trocas, o intercâmbio. Não existe mais o monopólio da informação, não é mais preciso guardar conteúdos, eles estão todos armazenados em bancos de dados, na *cloud*.

Com essa mudança de paradigma na forma como a sociedade consome informação, surgiu um conceito de “novos” alunos, que estão acostumados a interagir e explorar essas tecnologias. A geração atual está cada vez mais acostumada a usar esses conteúdos multimídia.

De acordo com Barbosa (2012), as TICs e as mídias interativas deram um novo sentido a Educação a Distância (EaD) por meio de *interfaces* de aprendizagem estruturadas para facilitar os contatos e a construção do conhecimento entre os participantes de EaD em Ambientes Virtuais de Aprendizagem (AVA), como uma proposta pedagógica, rica em trocas sociais. A mediação no EaD, estimulada por suportes tecnológicos, favorece o exercício cognitivo do aluno para adquirir novos conhecimentos com a orientação ou colaboração de outros – professores ou pares – que tenham mais domínio sobre os conteúdos.

O ensino baseado na educação que conta apenas com giz, lousa e saliva não é mais reconhecido como o único modelo de aprendizagem formal (MELO;BARROS;MORAES,1997). Esse ensino torna o aluno desmotivado, não contribuindo, assim, para facilitar seu aprendizado. O uso dos OAs atualmente disponíveis, além dos *sites* interativos, fóruns de discussões, redes sociais, postagens, salas de bate papo, entre outros, são algo que passa a fazer parte das metodologias de ensino atuais (LÁZARO; TEZANI, 2014).

Diante do exposto, esta pesquisa se justifica pela importância da exploração dos recursos das TICs para o ensino e também pelo fato de não haver na literatura uma ferramenta computacional que considerem o conjunto de problemas e algoritmos (A^* , Busca Gulosa, Dijkstra, Ford-Fulkerson e Vizinho mais Próximo) investigados neste trabalho. A pesquisa aqui apresentada trata de assuntos de vital importância para as áreas de Pesquisa Operacional e Educação em Engenharia as quais, segundo a ABEPRO, compõem as 10 áreas afins de um Curso de Pós Graduação em Engenharia de Produção, ao qual está vinculado o autor dessa pesquisa. Além disso, o processo de ensino desses algoritmos em cursos de graduação e pós-graduação normalmente não conta com recursos multimídia que auxiliem nos seus aprendizados.

1.4 ESTRUTURA DA DISSERTAÇÃO

Os demais capítulos presentes nesta dissertação estão estruturados do seguinte modo: no segundo capítulo há uma fundamentação teórica a respeito de AVAs e OAs, POR, representação de problemas de otimização em redes, problemas de caminho mínimo, fluxo máximo e caixeiro viajante, incluindo os respectivos algoritmos de resolução, a biblioteca JUNG, bem como a descrição de alguns trabalhos correlatos. No terceiro capítulo estão descritos a metodologia empregada bem como a maneira como foram conduzidos os experimentos. No quarto capítulo os resultados obtidos a partir da aplicação do OA desenvolvido em turmas de graduação são apresentados e discutidos. Por fim, no quinto capítulo, apresentam-se as conclusões e as sugestões para continuidade dessa pesquisa.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, primeiramente são descritos os conceitos relacionados aos ambientes de aprendizagem e os objetos de aprendizagem. Em seguida, são apresentados os Problemas de Otimização em Redes (POR) e os algoritmos que os solucionam – A* (A estrela), Busca Gulosa e Dijkstra (utilizados para a solução de problemas de caminho mínimo), Ford-Fulkerson (utilizado para a solução de problemas de fluxo máximo) e Vizinho mais Próximo (empregado na solução do problema do caixeiro viajante). Por fim, apresenta-se a biblioteca JUNG, utilizada no desenvolvimento do OA proposto.

2.1 AMBIENTES DE APRENDIZAGEM ON-LINE

Um ambiente de aprendizagem on-line usa dois ou mais formatos de conteúdo, emitindo mensagens do emissor para o receptor (ou seja, do professor para o aluno). A combinação de diferentes tipos de conteúdo cria o efeito de aprendizagem com um meio que é capaz de produzir resultados de aprendizagem benéficos específicos (MAYER; MORENO, 2002).

Um ambiente de aprendizagem on-line pode ser definido como um ambiente multimídia composto por um conjunto de simulações, modelagens e visualizações, o qual é projetado com intuito de oferecer recursos para contribuir para a aprendizagem coletiva e individual (ALMEIDA; VIEIRA; LUCIANO, 2001). De acordo com Silva (2006), os objetos de aprendizagem também são elementos que podem compor um ambiente de aprendizagem on-line ou ambiente virtual de aprendizagem. Esses ambientes são utilizados para permitir a criação de unidades de aprendizagem: cursos, disciplinas, lições, atividades. Professores criam materiais para esses ambientes, enquanto os gerenciadores de ambientes (normalmente pessoal da área técnica de informática) disponibilizam esses materiais e gerenciam o funcionamento do ambiente, através de criação de *login* e senhas, e do controle de acesso. Os alunos utilizando um ambiente multimídia têm acesso aos materiais e OA's (objetos de aprendizagem) disponibilizados, seja copiando/baixando arquivos

ou assistindo a vídeos lá disponíveis, desenvolvem atividades propostas, participam de fóruns e até mesmo fazem atividades avaliativas.

Em essência, multimídia é o uso de um número de diferentes formatos de conteúdo para transmitir informação. Texto, animação, vídeo, áudio e gráficos são exemplos de tipos comuns de formato de conteúdo utilizados (YI; HWANG, 2003).

A necessidade de educação mudou por causa de um aumento na demanda por uma força de trabalho altamente qualificada, que se espera que aprenda continuamente. Como resultado, a aprendizagem on-line está se tornando cada vez mais uma parte importante do ensino superior à distância (ALAVI; LEIDNER, 2001). A fim de apoiar a aprendizagem on-line, vários ambientes de aprendizagem têm sido desenvolvidos, como o *WebCT*, quadro negro, e *ClassFronter* (NGAI; POON; CHAN, 2007).

A introdução de ambientes de aprendizagem on-line no ensino superior apresenta novos desafios como a mudança nos papéis e expectativas de professores e alunos. No entanto, esta transição nem sempre acontece sem complicações (BENNETT; LOCKYER, 2004).

Para ser um verdadeiro mediador da aprendizagem, o ambiente de aprendizagem on-line tem que ser aceito por docentes e alunos. A aceitação é um pré-requisito das intenções de uso real dos sistemas de informação (KELLER, 2009).

Pesquisas recentes tentam traçar uma linha entre a pesquisa e a prática. A questão sobre se a aprendizagem on-line tem efeitos positivos e em que casos ela contribui, ainda não foi claramente respondida (SAADE; NEBEBE; MAK, 2011).

Muitos modelos cognitivos diferentes de aprendizagem têm sido sugeridos e estudados. Estes modelos foram baseados em psicologia social ou sociologia e, em seguida, formatados especificamente para utilização de estudo via ambientes de aprendizagem on-line (VENKATESH *et al.*, 2003).

É de suma importância perceber, como apresentado pelos autores acima citados, a mudança de paradigma das aulas. Os ambientes de aprendizagem on-line fazem parte, cada vez mais, do modelo educacional atual. Além disso, eles podem ser compostos por unidades de aprendizagem, denominadas objetos de aprendizagem, os quais são descritos a seguir.

2.1.1 Objeto de Aprendizagem (OA)

O *Institute of Electrical and Electronics Engineers*¹ (IEEE, 2005), que em 1996 criou um grupo para desenvolver e promover padrões tecnológicos instrucionais, o *Learning Technology Standards Committee*² (LTSC), define objeto de aprendizagem como qualquer entidade, digital ou não digital, que pode ser utilizada, reutilizada ou referenciada em aprendizagem, educação ou treinamento suportados por tecnologias.

Para Wiley (2002), os OAs permitem aos projetistas a construção de pequenos componentes instrucionais, os quais podem ser reutilizados em diferentes contextos de aprendizagem, incluindo os ambientes de aprendizagem on-line. Assim, trata-se de qualquer recurso (digital ou não) que pode ser reutilizado para apoiar a aprendizagem (WILEY, 2002).

Na literatura, podem-se encontrar diversas conceituações para objeto de aprendizagem. Para a Rede Interativa Virtual de Educação (RIVED), um dos principais repositórios de objetos de aprendizagem nacionais, encontra-se a seguinte definição:

“Um objeto de aprendizagem é qualquer recurso que possa ser reutilizado para dar suporte ao aprendizado. Sua principal ideia é "quebrar" o conteúdo educacional disciplinar em pequenos trechos que podem ser reutilizados em vários ambientes de aprendizagem. Qualquer material eletrônico que provém informações para a construção de conhecimento pode ser considerado um objeto de aprendizagem, seja essa informação em forma de uma imagem, uma página HTML, uma animação ou simulação” (RIVED, 2007).

A tecnologia é um agente de mudança, e as grandes inovações tecnológicas podem resultar em mudanças de paradigma. A rede de computadores conhecida como a internet é uma inovação que leva a mudanças radicais na forma como as pessoas se comunicam e fazem negócios. A internet está prestes a trazer uma mudança de paradigma na forma como as pessoas aprendem. Consequentemente, uma grande

¹ Instituto de Engenheiros Elétricos e Eletrônicos

² Comitê de Padrões de Tecnologia de Aprendizagem (IEEE)

mudança também pode vir nos materiais educativos desenvolvidos e entregues para aqueles que desejam aprender. Uma tecnologia instrucional chamada "objetos de aprendizagem" atualmente lidera outros meios de ensino para a próxima geração de design instrucional, devido ao seu potencial de reutilização, generatividade, adaptabilidade e escalabilidade (HODGINS, 2002; GIBBONS; NELSON, RICHARDS, 2000).

Mason, Pegler e Weller (2005) afirmam que é cada vez mais comum o uso de objetos de aprendizagem em cursos a distância, como uma aula ou ainda uma lição distinta dentro de uma aula. Cada objeto proporciona ao aluno uma experiência em particular, permitindo que o mesmo se envolva em uma oportunidade de aprendizado a cada sessão de estudo.

Para McGreal (2004) um objeto de aprendizagem pode ser baseado em um texto eletrônico, uma simulação, uma página *web*, uma imagem, um filme Quicktime³, um aplicativo Java⁴, entre outros.

Silva (2006) define que o objeto de aprendizagem é:

- Uma unidade de auto conteúdo de aprendizagem: cada OA pode ser tomado de forma independente;
- Reutilizável: pode ser utilizado em múltiplos contextos para múltiplos propósitos;
- Alterável: pode residir na internet e versões revisadas podem ser disponibilizadas para outros usuários;
- Agregável: pode ser agrupado em maiores coleções de conteúdos, incluindo estruturas tradicionais de cursos;
- Acompanhado de metadados: cada OA tem informações descritivas, permitindo que sejam facilmente encontrados por meio de uma pesquisa.

Ainda segundo Silva (2006), as principais características de um objeto de aprendizagem são:

³ Quicktime é um programa para vídeos da empresa Apple (www.apple.com)

⁴ Java é uma linguagem de programação desenvolvida pela empresa Sun (www.sun.com)

- Granularidade: o quão grande um objeto de aprendizagem pode ser para que não impeça sua reutilização. Se o objeto for muito grande, envolvendo muitos conceitos e com uma proposta hermeticamente fechada em relação ao que pretende ensinar, tornar-se-ia difícil ou mesmo impossível a sua reutilização;
- Metadados: descrições que acompanham os objetos são chamadas de metadados (dados acerca de dados). Um objeto de aprendizagem, portanto, para ser visualizado e/ou pesquisado por usuários, deve possuir um metadado que o especifique claramente;
- Reutilização: um objeto construído para um determinado contexto de aprendizagem pode ser reutilizado para construção de outro objeto de aprendizagem em contexto diferente;
- Revisável: permitir que as revisões sejam feitas sem afetar o restante do objeto de aprendizagem;
- Personalizável: para que educadores e aprendizes possam alterá-lo no atendimento de suas necessidades individuais;
- Aplicável: possa ser utilizado em diferentes projetos instrucionais;
- Independente: para que seja um segmento de instrução, uma pequena unidade para aprendizagem;
- Recombinável: para poder ser utilizado em lições ou cursos por meio de combinações de objetos de aprendizagem;
- Durável: para poder ser reutilizado várias vezes sem se tornar obsoleto;
- Interoperável: para ser utilizado independente do tipo de software ou sistema operacional que irá acessá-lo

De acordo com Sheperd (2000) objetos de aprendizagem são classificados em três tipos, baseado nos objetivos de sua utilização, como pode ser verificado no quadro 1. O autor elenca ainda os benefícios dos objetos de aprendizagem para alunos, administradores e desenvolvedores, os quais são apresentados no quadro 2.

Quadro 1 – Tipos de Objetos de Aprendizagem

Integrados	Informativos	Práticos
Mini tutoriais; Mini estudos de caso, Simulações, etc. com suporte da TI.	Visões gerais / sumários; Descrições / definições; Demonstrações / modelos; Exemplos de trabalhos; Casos / histórias; <i>Papers</i> e artigos.	Jogos de simulação, sem o uso da TI. Exercícios de prática e treinamento; Exercícios de revisão; Testes / avaliações.

Fonte: Adaptado de (SHEPERD, 2000)

De acordo com Pegden, Shannon e Sadowiski (1990), simulação é o processo de projetar um modelo computacional de um sistema real e conduzir experimentos com este modelo com o propósito de entender seu comportamento e/ou avaliar estratégias para sua operação.

Quadro 2 – Benefícios dos objetos de aprendizagem

Para os alunos	Para os administradores	Para os desenvolvedores
Personalização: cursos podem ser construídas para atender às necessidades individuais; A aprendizagem vem em partes menores; A aprendizagem fica disponível em uma	Os cursos podem ser personalizados para atender às necessidades de diferentes públicos; Os cursos podem ser construídos utilizando componentes de uma vasta gama de fontes; Os componentes podem ser	Os objetos podem ser criados ou modificados usando muitas ferramentas de autoria diferentes; Os mesmos objetos podem ser empregados em uma variedade de plataformas de hardware e software.

base <i>real-time</i> .	reutilizados para atender a uma gama de necessidades de aprendizagem.	
-------------------------	---	--

Fonte: Adaptado de (SHEPERD, 2000)

Os metadados de objetos de aprendizagem consistem em um ou mais arquivos que descrevem seus aspectos técnicos e educacionais. Conforme Wiley (2002), os metadados são formados por um conjunto de informações para descrever um recurso com estrutura padronizada, facilitando a recuperação e acesso aos objetos de aprendizagem, ou em suas palavras: dados sobre dados.

2.2 PROBLEMAS DE OTIMIZAÇÃO EM REDES (POR)

Em se tratando de problemas de otimização, é comum encontrar POR. Eles surgem naturalmente na análise e projeto de sistemas de grande porte, tais como comunicação e transporte. Eles também podem ser utilizados para modelar importantes classes de problemas combinatórios, como atribuição, caminho mínimo, fluxo máximo e o problema do caixeiro viajante. Quando se fala em redes, entende-se as mesmas como sendo um conjunto de nós ligados por arestas, tendo como exemplos mais comuns e conhecidos a internet, as redes sociais, o corpo humano, as redes telefônicas e as vias públicas, entre outros (MENDES, 2005). Nas subseções a seguir apresentam-se a representação dos problemas em redes, os problemas do caminho mínimo, fluxo máximo e caixeiro viajante.

2.2.1 Representação de Problemas em Redes

De acordo com Arenales *et al.* (2007), muitos problemas de otimização encontrados na prática podem ser melhor analisados por meio de uma estrutura especial denominada grafo ou rede. Assim, eles são denominados POR e estão presentes em diversas aplicações como, por exemplo: transporte ou fluxo de itens de um ponto a outro em uma rede com um determinado objetivo, transmissão de mensagens em

redes de comunicação de dados, envio de água em uma rede de distribuição, transporte de carga em uma rede viária, planejamento e programação da produção, programação de projetos, programação de máquinas e de pessoal, distribuição de bens, etc..

Esses POR podem ser tratados genericamente como Problema do Caminho Mínimo (PCM), Problema de Fluxo Máximo (PFM), Problema do Caixeiro Viajante (PCV) ou Problema do Carteiro Chinês, entre outros.

Os POR podem ser modelados por grafos, e as rotas podem ser modeladas pelos caminhos desses grafos. Além disso, também podem existir algumas restrições em relação às características dessas ligações, como por exemplo, custos associados ao uso específico das rotas.

Conforme definido por Bertsekas (1998), um grafo $G=(N, A)$ consiste em um conjunto de nós e um conjunto de arestas. O número de nós e arestas são denotados por N e A , respectivamente. Um grafo é referido como estando completo se contiver todas as arestas possíveis; isto é, se os nós estiverem conectados entre si. De acordo com a literatura há grafos orientados e os grafos não orientados, os quais são empregados de acordo com o problema e o cenário que o descreve.

De acordo com Campos (2013), os grafos são orientados quando suas arestas possuem uma orientação definida, ou seja, quando a aresta indica uma direção de fluxo ou sentido entre os dois nós que ela conecta. Naturalmente, quando esta noção de direção não existe, dizemos que o grafo é não orientado.

Um percurso P em um grafo é uma sequência de arestas, representada por uma sequência de nós (N_1, N_2, \dots, N_k) com $k \geq 2$ conectados por $k-1$ arestas. N_1 e N_k são chamados de partida (ou de origem) e nó final (ou destino) de P , respectivamente. Neste trabalho são considerados apenas os grafos não orientados, em decorrência da natureza dos problemas abordados. Assim, todas as referências a um grafo nessa pesquisa, implicitamente assumem que se trata de um grafo não orientado.

2.2.2 Problema de Caminho Mínimo (PCM)

No PCM o objetivo é encontrar o caminho mais curto entre dois nós de um grafo. Em outras palavras, dado um grafo que representa o problema investigado e um par de nós (origem e destino), o PCM consiste em determinar o percurso de menor custo que liga estes nós (CORMEN, 2001).

O PCM surge em um número surpreendentemente grande de contextos. Por exemplo, em uma rede de comunicação de dados, na qual os pacotes de dados trafegam a partir de suas origens até seus destinos; na transposição de um rio, onde o caminho mínimo pode levar a economia de milhões de dólares em desapropriações e tempo de projeto, em uma rede de transporte na qual um veículo ou pessoa deve fazer um percurso de um ponto a outro com o menor tempo de viagem, etc. (ATZINGEN *et al.*, 2007).

O PCM vem sendo estudado há mais de 40 anos em diversas áreas de conhecimento como Ciência da Computação, Matemática e Engenharia de Transportes. Devido à dificuldade computacional em tratar este problema, a maioria das pesquisas nesta área se concentra no desenvolvimento de algoritmos eficientes para resolver o PCM. O PCM surge em um número surpreendentemente grande de contextos. Por exemplo, em uma rede de comunicação de dados, existem roteadores (nós) e *links* de comunicação (arestas) em que os pacotes de transferência (grupos de bits) trafegam a partir de suas origens até seus destinos. O método mais comum para a seleção do percurso de deslocamento (ou via) de pacotes é baseada na formulação de um caminho mínimo. A cada elo de comunicação é atribuído um custo, que pode ser, inclusive, a distância entre os nós que representam o elo. Um caminho mínimo é determinado por meio de um algoritmo endereçando cada pacote ao longo do percurso entre o nó de origem e nó de destino. (ATZINGEN *et al.*, 2007).

Na última década, muitos OAs para ensino de conteúdos de cursos de graduação têm sido propostos. Contudo, existem poucos OAs de algoritmos para a resolução do PCM. Entre as aplicações encontradas para esta finalidade na literatura podemos citar as propostas por Barros *et al.* (2007), Berón, Henriques e Pereira (2006), Welzl e Ali (2006) e Lima e Araújo (2013). Destas, apenas a última considera os mesmos

algoritmos tratados nesta pesquisa. Contudo, ela não foi desenvolvida para ambiente *web*, o que dificulta o seu uso em decorrência da necessidade de instalação de pacotes adicionais (por exemplo, o *framework* .NET), necessários para sua execução e seu acoplamento a um ambiente de aprendizagem on-line.

Na literatura, existem muitos algoritmos para resolver o PCM. Os principais são: Dijkstra, Busca Gulosa (*Greedy Search*) e A* (A Estrela), os quais são aplicados no objeto de aprendizagem proposto neste trabalho, denominado TASNOP, e brevemente descritos nas subseções 2.2.2.1 a 2.2.2.3.

2.2.2.1 Algoritmo de Dijkstra

O algoritmo de Dijkstra foi concebido pelo cientista da computação holandês Edsger Dijkstra em 1956 e publicado em 1959, para solucionar o problema do caminho mínimo em grafos simples para determinar o menor caminho entre dois vértices. Atualmente o algoritmo de Dijkstra é empregado em vários contextos, desde o estudo de uma cadeia de produção até o clássico problema do carteiro que não pode passar duas vezes na mesma rua (BARROS; PAMBOUKIAN; ZAMBONI, 2007).

De acordo com Atzingen et. al. (2007), o algoritmo de Dijkstra permite determinar a solução ótima por meio da adição de vértices à árvore de caminho mínimo pelo processo de relaxamento de uma aresta; em outras palavras, consiste em verificar se há a possibilidade de melhorar o caminho obtido até o momento.

O pseudocódigo a seguir descreve o funcionamento do algoritmo de Dijkstra.

Dijkstra(α , s_0 , \bar{O})

```

1  $\Gamma \leftarrow \emptyset$ ;
2  $\Sigma \leftarrow \{s_0\}$ 
3 enquanto  $\Sigma \neq \emptyset$  faça
4      $s \leftarrow \text{removePrimeiro}(\Sigma)$ 
5     se  $s \in \bar{O}$  então devolva caminho(s)
6      $\Gamma \leftarrow \Gamma \cup \{s\}$ 
7      $U \leftarrow \text{gere\_sucessoresG}(s, \alpha)$ 
```

```

8      Para cada  $u \in U \setminus \Gamma$ 
9          se  $u \notin \Sigma$  então  $\Sigma \leftarrow \Sigma \cup u$ 
10         senão  $\Sigma \leftarrow \Sigma \cup u$  (se, e somente se,  $u$  é menor que sua réplica em  $\Sigma$ )
11 devolva fracasso

```

2.2.2.2 Algoritmo de Busca Gulosa (*Greedy Search*)

Estratégias de busca gulosa utilizam funções heurísticas para fazer as melhores escolhas locais a cada etapa, com o objetivo de encontrar a melhor solução global (RUSSEL; NORVIG, 1995). Assim, o algoritmo de Busca Gulosa sempre opta por fazer o que parece melhor em cada interação. Para tanto, uma função de custo que norteia o algoritmo é dada por $f(n_i)=h(n_i)$, onde h é uma heurística admissível associada ao nó n_i .

A decisão é baseada apenas nas informações disponíveis no momento, sem se preocupar com os efeitos futuros dessas decisões. No PCM, estratégias de busca gulosa não encontram as melhores soluções, mas o uso de uma heurística gulosa pode produzir soluções localmente ótimas que se aproximam de uma solução ótima global, dentro de um prazo razoável de computação (ATZINGEN *et al.*, 2007).

Muitas vezes, algoritmos gulosos não conseguem encontrar a solução ótima global, porque geralmente não operam de forma exaustiva em todos os dados. Eles podem assumir compromissos com algumas escolhas muito cedo que os impedem de encontrar a melhor solução global mais tarde. Se um algoritmo guloso pode comprovadamente produzir um ótimo global para uma determinada classe de problema, ele normalmente torna-se uma escolha viável, pois é mais rápido do que outros métodos de otimização, como a programação dinâmica.

No caso de roteamento de redes, usando um algoritmo guloso, uma mensagem é enviada ao nó vizinho que é "mais próximo" do destino. A noção de localização de um nó (e, portanto, "proximidade") pode ser determinada pela sua localização física.

O pseudocódigo a seguir descreve o funcionamento do algoritmo *Greedy Search*.

BuscaGulosa(α , s_0 , \bar{O})

```

1  $\Gamma \leftarrow \emptyset$ ;
2  $\Sigma \leftarrow \{s_0\}$ 
3 enquanto  $\Sigma \neq \emptyset$  faça
4      $s \leftarrow \text{removePrimeiro}(\Sigma)$ 
5     se  $s \in \bar{O}$  então devolva caminho(s)
6      $\Gamma \leftarrow \Gamma \cup \{s\}$ 
7     insereEmOrdem(sucessores $H(s, \alpha) \rightarrow \Gamma, \Sigma$ )
8 devolva fracasso

```

2.2.2.3 Algoritmo A*

O algoritmo A* (A estrela) combina as estratégias de busca empregadas nos algoritmos de Dijkstra e Busca Gulosa em um único algoritmo, utilizando uma função de custo $f(n_i) = g(n_i) + h(n_i)$, onde g e h são, respectivamente, uma função de custo e uma função heurística. Assim, A* encontra soluções ótimas para o PCM explorando um espaço de busca menor do que o examinado pelo algoritmo de Dijkstra (RUSSEL; NORVIG, 1995). Caso exista, o algoritmo A* sempre encontrará um caminho de menor custo entre o nós de origem e destino. Esse algoritmo foi descrito em 1968 por Peter Hart, Nils Nilsson, e Bertram Raphael, e foi chamado de algoritmo “A”. Sua aplicação vai desde aplicativos para encontrar as melhores rotas entre localidades à resolução de problemas em rede, além de ser muito usado em jogos (ATZINGEN *et al.*, 2007).

O pseudocódigo a seguir descreve o funcionamento do algoritmo A*.

A*(α , s_0 , \bar{O})

```

1  $\Gamma \leftarrow \emptyset$ ;
2  $\Sigma \leftarrow \{s_0\}$ 
3 enquanto  $\Sigma \neq \emptyset$  faça
4      $s \leftarrow \text{removePrimeiro}(\Sigma)$ 
5     se  $s \in \bar{O}$  então devolva caminho(s)
6      $\Gamma \leftarrow \Gamma \cup \{s\}$ 
7     insereEmOrdem(sucessores $F(s, \alpha) \rightarrow \Gamma, \Sigma$ )
8 devolva fracasso

```

2.2.3 Problema do Fluxo Máximo

O PFM é um dos clássicos problemas de otimização combinatória com muitas aplicações em sistemas de energia elétrica, redes de comunicação, redes de computadores e redes logísticas. O objetivo do problema é encontrar o valor do fluxo máximo a' de um nó inicial para um nó final (HAN; PENG; WANG, 2014).

A motivação para o PFM veio do sistema ferroviário Soviético. Considerando uma rede ferroviária que liga duas cidades por meio de um número intermediário de cidades, onde cada elo da rede tem um número atribuído à sua capacidade e assumindo que existe uma condição de estado estacionário, encontra-se um fluxo de carga máximo de uma determinada cidade para a outra. Devido às suas muitas aplicações em sistemas de energia elétrica, comunicação, redes de computadores e redes de logística, o PFM tem sido amplamente estudado ao longo dos últimos 50 anos. Portanto, do ponto de vista do valor de aplicação, o estudo do PFM é de grande importância (HAN; PENG; WANG, 2014). Na literatura, existem muitos algoritmos para resolver o PFM. O algoritmo de Ford-Fulkerson foi aplicado no objeto de aprendizagem proposto neste trabalho, denominado TASNOP, e brevemente descrito na seção 2.2.3.1.

2.2.3.1 Algoritmo Ford-Fulkerson

O algoritmo de Ford-Fulkerson é um algoritmo que calcula o fluxo máximo em uma rede de fluxo. Ele é muito eficaz na resolução do problema de maximizar o fluxo. Funciona com um conjunto de arestas que representam o fluxo de um nó de origem até um nó de destino. Para melhor compreensão da teoria do algoritmo, alguns conceitos devem ser entendidos:

- Capacidade da aresta: fluxo máximo entre dois nós;
- Capacidade restante de uma aresta: se por meio de uma aresta, fluir uma quantidade menor do que a sua capacidade, essa aresta possui capacidade restante;

- Aresta Saturada: a aresta cuja capacidade remanescente é nula;
- Caminho saturado: é um caminho sem nós repetidos; com nenhum fluxo negativo e pelo menos uma de suas arestas saturadas (SOUZA; MELO; PINTO, 2014).

O algoritmo procura caminhos aumentantes na rede residual. A cada iteração é encontrado um caminho aumentante do ponto inicial ao ponto final para que se escolha o valor da menor capacidade residual dentre as arestas deste caminho e aumenta-se o fluxo de todas as arestas do caminho nesse valor. Repete-se essa operação até que se encontre uma rede residual onde não haja um caminho aumentante ligando o nó inicial ao nó final (CORMEN, 2001).

O tempo de execução do algoritmo de Ford-Fulkerson depende de como o caminho aumentante é escolhido na rede residual, após a escolha do caminho pode ocorrer uma busca em largura ou profundidade, conforme descreve o pseudocódigo a seguir.

1. Escolhe-se um caminho qualquer desde a origem até o nó destino, mas em que as arestas tenham capacidade positiva (>0);
2. Procurar nesse caminho a aresta com menor capacidade;
3. Diminuir a capacidade de fluxo em cada aresta do caminho no sentido considerado;
4. Regressar ao 1º passo. Se não existir nenhum caminho em que todas as arestas tenham capacidade positiva, isso significa que já se determinou o fluxo máximo

2.2.4 Problema do Caixeiro Viajante

O PCV é um dos mais clássicos problemas de redes. Dado um conjunto de cidades e uma matriz de distâncias, encontre o caminho que tenha a menor distância a ser percorrida visitando todas as cidades passando exatamente uma única vez em cada cidade e retornando a cidade de origem.

Sua origem é creditada a William Rowan Hamilton, que inventou um jogo cujo objetivo era o de traçar um roteiro por meio dos nós de um dodecaedro (nós equivalem a cidades) que iniciasse e terminasse no mesmo nó (na mesma cidade), sem repetir uma visita. (GOLDBARG; LUNA, 2005).

O PCV tem importantes aplicações práticas, tais como identificar itinerários de cobertura de regiões, como, por exemplo, a rota de um carteiro, visto que diminuir o trajeto pode significar economia de tempo e de pessoal. Outra aplicação prática é o processo de furacão de placas para circuito impresso. Como existem vários furos de diâmetros diferentes, e a troca da ferramenta pode demandar tempo, os furos de mesmo tamanho devem ser feitos de maneira sequencial, percorrendo-se a menor distância possível, para diminuir o tempo gasto numa produção em série (CALADO; LADEIRA, 2011). Na literatura, existem muitos algoritmos para resolver o PCV. O algoritmo do Vizinho mais Próximo foi aplicado no objeto de aprendizagem proposto neste trabalho, denominado TASNOP, e brevemente descrito na seção 2.2.4.1.

2.2.4.1 Algoritmo do Vizinho Mais Próximo

A heurística do vizinho mais próximo utiliza uma matriz para definir a distância entre os nós. O percurso é construído com base na distância entre estes nós, sendo o nó mais próximo da origem adicionado primeiro e os demais pontos adicionados posteriormente conforme a necessidade da criação de novas rotas. Um ponto é adicionado a uma rota, segundo a sua proximidade em relação ao último ponto adicionado (SOLOMON, 1987).

O problema do Vizinho mais próximo envolve determinar o ponto em um conjunto de dados que está mais próximo para um determinado ponto de consulta. É frequentemente usado em sistemas de informação, onde os pontos estão associados com alguma localização geográfica (por exemplo, cidades). A pergunta típica é: "Que cidade é mais próxima de meu atual ponto de localização?" (BEYER et. al., 1999).

O pseudocódigo a seguir sumariza o funcionamento do algoritmo Vizinho Mais Próximo

1. Escolha um nó inicial;
2. Encontre a aresta de menor custo que esteja conectada ao nó atual e a um nó não visitado N;
3. Faça o nó atual ser visitado;
4. Marque o nó atual como visitado;
5. Se todos os nós do domínio estiverem visitados, encerre o algoritmo;
6. Se não volte para o passo 2.

No quadro 3 é possível visualizar uma compilação dos POR estudados nessa pesquisa, bem como os algoritmos empregados em suas soluções.

Quadro 3 – Compilação dos POR abordados e seus algoritmos solucionadores

Problema	Algoritmos
Caminho Mínimo	Dijkstra
	Busca Gulosa
	A*
Fluxo Máximo	Ford-Fulkerson
Caixeiro Viajante	Vizinho mais Próximo

A seguir apresenta-se a biblioteca JUNG, utilizada na implementação dos algoritmos mencionados no quadro 3 e que compõem o TASNOP.

2.3 JUNG (*Java Universal Network Graph*)

A biblioteca JUNG ⁵ é de uso livre e fornece uma linguagem comum e extensível para a modelagem, análise e visualização de dados que podem ser representados como um grafo ou rede. Ela foi escrita em Java e permite o uso das funcionalidades embutidas na sua API⁶ (interface de programação de aplicações), como o mecanismo para anotação nos grafos e relações com metadados. Isso facilita a criação de ferramentas computacionais para examinar as relações entre as entidades de uma rede, bem como os metadados anexados a cada entidade e relação.

A distribuição atual da JUNG inclui implementações de uma série de algoritmos oriundos da teoria dos grafos, mineração de dados e análise de redes sociais, tais como decomposição, otimização, geração de grafos aleatórios, análise estatística e cálculo de distâncias e coeficientes de redes. A biblioteca fornece também uma estrutura de visualização que facilita a construção de mecanismos para a exploração interativa de dados da rede. Os usuários podem usar os algoritmos disponíveis ou a estrutura da biblioteca para projetar os seus próprios algoritmos e *layouts*.

Essas características da JUNG consistiram no principal motivo para escolha desta biblioteca na implementação do OA proposto neste trabalho.

2.4 TRABALHOS CORRELATOS

Na literatura podem ser encontrados trabalhos propondo o uso de ferramentas computacionais para o auxílio da aprendizagem de algoritmos de otimização. O objetivo, a forma de aplicação da ferramenta bem como os resultados obtidos em alguns desses trabalhos são apresentados a seguir.

⁵ Disponível em: <http://jung.sourceforge.net/>

⁶ *Application Programming Interface*

Lawrence, Badre e Stasko (1994) conduziram uma pesquisa que relata a avaliação empírica do uso de animações para ensinar os algoritmos Kruskal's e Árvore Geradora Mínima. Os resultados dessa pesquisa indicaram que os grupos que utilizaram animações obtiveram um desempenho mais alto do que os outros grupos. Esses resultados baseiam-se em um máximo possível de dezenove pontos, um ponto de cada questão correta. As perguntas do teste eram de alternativas verdadeiro/falso ou múltipla escolha.

Iqbal e Tahir (2003) relatam o uso do algoritmo Bucket para resolver uma série de problemas de otimização em redes (POR), inclusive o PCM (Problema de caminho mínimo). A pesquisa conduzida pelos autores demonstrou como um professor pode ajudar os alunos a descobrir uma série de algoritmos em grafos com alguma ajuda inicial, começando com algo aparentemente simples mas capaz de ser transformado em um número de algoritmos poderosos, com pequenas modificações.

Alstes e Linqdvist (2007) desenvolveram um objeto de aprendizagem chamado VERKKOKE: *Learning Routing and Network Programming Online*, para a aprendizagem de roteamento e programação de redes. O VERKKOKE permite combinar a teoria e a prática, tendo em vista que os alunos ganham prática escrevendo programas de rede reais que precisam ser autenticados por meio da internet e implementam algoritmos baseados em dados recebidos a partir da rede. O VERKKOKE foi implantado e usado pela primeira vez durante o outono de 2006 e com base no *feedback*, os alunos sentiram que a nova estrutura de atribuição dessa ferramenta era bom e ajudou no processo de aprendizagem. No entanto, mais estudos são necessários para entender como os alunos reagem e como ela realmente ajuda a aprender. Por fim, os autores mencionam que futuras melhorias incluem a adição de suporte para o ensino de protocolos de rede, além dos protocolos de roteamento dinâmicos.

Bridgeman *et al.* (2000) elaboraram uma ferramenta chamada PILOT para o ensino dos algoritmos Árvore Geradora Mínima, Árvore de Busca e Menor Caminho. O PILOT foi criado com vários objetivos em mente. Em primeiro lugar, ser interativo e que pudesse ser utilizado em sala de aula para ajudar na exposição dos problemas. Contudo, os resultados do uso do PILOT não foram disponibilizados pelos autores.

Malmi *et al.* (2004) desenvolveram uma ferramenta chamada TRAKLA2 para o ensino dos algoritmos Árvore Geradora Mínima, Prim's, Menor Caminho Dijkstra e

BuildHeap. Exercícios usando o TRAKLA2 foram aplicados pela primeira vez nos cursos de estruturas básicas de dados e algoritmos na Universidade de Tecnologia de Helsinki (HUT). Durante dois anos, mais de 1000 alunos utilizaram o sistema. Em 2004, a Universidade de Turku (UTU) também adotou o sistema para seu curso de estrutura de dados com mais de 100 alunos. Como um todo, o TRAKLA2 tem funcionado bem com resultados surpreendentemente bons. Em 2004, 30% dos estudantes na HUT alcançaram o número máximo de pontos para todos os 26 exercícios, e 55% atingiram pelo menos 90% do máximo. Apenas 15% dos alunos não conseguiram o mínimo exigido, que é de 50% dos pontos. Na prática, esses foram estudantes que abandonaram o curso. Na UTU os resultados foram ainda melhores, mesmo para aqueles que nunca utilizaram a ferramenta antes. No geral, os alunos gostaram muito do TRAKLA2 e consideraram que ele foi útil para a aprendizagem.

3. MATERIAIS E MÉTODOS

Neste capítulo são descritos o método de pesquisa, os materiais empregados e a metodologia para a condução dos experimentos, bem como os parâmetros empregados no objeto de aprendizagem proposto e os critérios para a avaliação do seu desempenho.

3.1 MÉTODO DE PESQUISA

De acordo com Lakatos e Marconi (1996), “pesquisar não é apenas procurar a verdade; é encontrar respostas para questões propostas, utilizando métodos científicos”. Por meio dessa definição pode-se perceber que pesquisa não é algo simples e não é um simples processo investigativo. Ela visa obter compreensões aprofundadas em relação aos problemas estudados. Pesquisar requer um planejamento minucioso das etapas, como seleção do tema de pesquisa, definição do problema, processo de coleta, análise, tratamento dos dados, e apresentação dos resultados.

Esta pesquisa seguiu os seguintes passos: primeiramente foi feita uma pesquisa bibliográfica para identificar corretamente a lacuna em relação ao objeto de aprendizagem, também foi necessário identificar quais são algoritmos mais utilizados para resolução dos POR considerados neste trabalho. A partir desse ponto identificou-se a linguagem de programação que atenderia às necessidades de desenvolvimento do OA (TASNOP). Após o desenvolvimento do OA, foram feitos experimentos com os alunos visando avaliar sua aplicabilidade no ensino. Para tanto, um questionário foi aplicado e suas respostas foram compiladas e apresentadas no próximo capítulo.

Os itens elencados a seguir têm como objetivo caracterizar esta pesquisa sob diferentes aspectos ou pontos de vistas.

- Do ponto de vista de sua natureza, ela se caracteriza como uma pesquisa aplicada. De acordo com Appolinário (2006), a pesquisa aplicada caracteriza-se pelo seu interesse prático, isto é, que os resultados sejam aplicados ou utilizados imediatamente na solução de problemas que ocorrem na realidade.
- Do ponto de vista da forma de abordagem do problema, ela se enquadra como uma pesquisa qualitativa, que consiste em um tipo de pesquisa que não é somente traduzida em números, mas que pretende verificar a relação da realidade com o objeto de estudo, gerando interpretações a partir de análises indutivas realizadas pelo pesquisador.
- Do ponto de vista de seus objetivos, ela se caracteriza como exploratória. De acordo com Appolinário (2006), a pesquisa exploratória visa a maior familiaridade com o problema no intuito de torná-lo mais explícito ou construir hipóteses.
- Do ponto de vista dos procedimentos técnicos, ela se caracteriza como um estudo de campo. Segundo Gil (2008), o estudo de campo trabalha com amostras de tamanhos suficientes para permitir análises estatísticas, sem se preocupar com a representatividade da amostra. É realizada por meio da observação direta das atividades do grupo estudado para captar as explicações e interpretações do que ocorre naquela realidade.

3.2 MATERIAIS EMPREGADOS

O desenvolvimento do OA foi baseado em quatro etapas:

- I. Estudo dos algoritmos de Dijkstra, Busca Gulosa, A* (A Estrela), Ford-Fulkerson e Vizinho mais Próximo, utilizados para a resolução dos POR abordados nesta pesquisa;
- II. Planejamento do OA;
- III. Desenvolvimento do OA;
- IV. Testes e avaliação do OA.

Na etapa de estudo dos algoritmos, foi feito um extenso levantamento bibliográfico acerca de cada um deles a fim de entender completamente seu funcionamento para futura implementação.

Na etapa de planejamento foi realizado o levantamento dos requisitos necessários para o desenvolvimento de um OA, a definição do escopo do projeto, a modelagem do OA, a escolha da linguagem de programação mais adequada para o desenvolvimento do OA e a elaboração do problema e do questionário a ser aplicado aos alunos.


Na etapa de desenvolvimento, foi levada em consideração a necessidade de facilidade de interação entre o aluno e o OA. No desenvolvimento do objeto de aprendizagem proposto empregou-se a linguagem de programação JAVA além da biblioteca JUNG para auxiliar na criação e visualização dos grafos, que facilitam sua disponibilização na *web*. No que diz respeito ao ambiente operacional de programação, empregou-se o NETBEANS⁷ sendo executado sobre o sistema operacional Windows 7.

Por fim, na etapa de testes e avaliação, o TASNOP foi instalado nos computadores de alguns laboratórios de informática da UNINOVE. Em seguida, duas turmas de alunos do último ano do curso de Ciência da Computação e uma turma da pós graduação em Engenharia de Produção foram convidados a participar do

⁷ ambiente de desenvolvimento integrado desenvolvido pela Oracle. O NETBEANS é uma ferramenta para desenvolvimento de aplicações que introduz um ambiente gráfico para a utilização da linguagem.

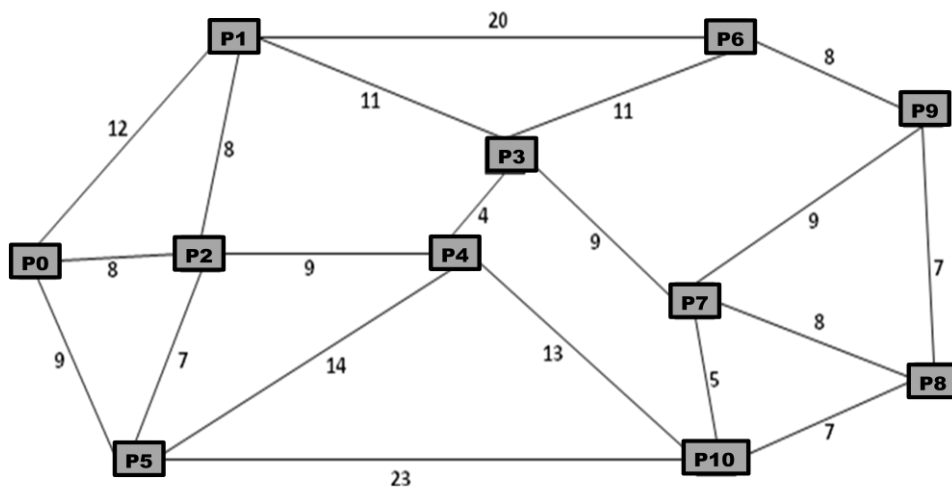
experimento e instruídos por seus professores sobre o uso do TASNOP e a forma de responder aos itens do questionário (figura 1) após a realização do experimento. Mais detalhes sobre como foram conduzidos os experimentos com o TASNOP são apresentados na seção a seguir.

Figura 1 – Exercício aplicado e questionário sobre o TASNOP

 UNINOVE Universidade Nove de Julho	Atividade Individual			
	Disciplina			Turma
	Curso			Nota
Professor				
Nome			RA	

EXERCÍCIO: APLICAÇÃO DE ALGORITMOS DE BUSCA EM UM PROBLEMA OTIMIZAÇÃO DE RECURSOS.

A rede ilustrada a seguir é a representação de um conjunto de 11 prédios residenciais (P_0, \dots, P_{10}) construídos em uma área afastada da cidade. As linhas ligando os prédios são os canos que foram instalados para a passagem de toda a fiação elétrica, telefônica, etc. e os números próximos às linhas indicam o comprimento (em metros) destes encaamentos. Um esquema de comunicação de dados a ser adotado pelo condomínio implica na colocação de dois dispositivos eletrônicos que deverão ser interligados por um cabo, ficando um na portaria do prédio **P0** e outro na portaria do prédio **P9**. Que caminho deverá percorrer o cabo para conectar o dispositivo do prédio **P0** ao dispositivo do prédio **P9**, de forma que o tamanho deste cabo seja o mínimo possível? Quantos metros de cabo serão necessários?



Escolha o algoritmo mais adequado (A*, Dijkstra ou Busca pela Melhor Estimativa/Greedy Search) para resolver o problema descrito e responda as questões a seguir:

- Qual foi o algoritmo escolhido para resolver o problema? Justifique a sua escolha.
- Quais são as respostas para as perguntas do enunciado do problema?
- Em relação à ferramenta computacional (TASNOP), assinale as seguintes alternativas:

	Muito fraco	Fraco	Médio	Bom	Muito bom
Contribuição para a aprendizagem dos algoritmos	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Usabilidade	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Potencial para ser uma ferramenta de ensino	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Confiabilidade dos resultados	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Interface	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

- Por favor, cite três itens que você mudaria/acrescentaria na ferramenta para melhorá-la.

3.3 METODOLOGIA PARA A CONDUÇÃO DOS EXPERIMENTOS

Para avaliar o aprendizado dos alunos sobre os conceitos discutidos em sala de aula (antes da utilização do TASNOP), foi proposto um exercício simples adaptado de (SANTOS, 2003) e apresentado na Figura 1.

No primeiro semestre de 2014, 78 alunos resolveram esse exercício sem o uso da ferramenta. Em adição, no segundo semestre desse ano, 90 alunos dos cursos de graduação em Ciência da Computação e pós graduação em Engenharia de Produção foram convidados a resolver o mesmo exercício utilizando o TASNOP. Desses 90 alunos, 54 resolveram o problema dado no enunciado do exercício e responderam a um questionário (figura 1) que incluía questões relativas ao uso da ferramenta. Dessa amostra de 54 alunos, 51 eram do sétimo e oitavo semestres do curso de graduação em Ciência da Computação e 3 do segundo quadrimestre do curso de Mestrado em Engenharia de Produção. Contudo, os resultados referentes a esses 3 últimos alunos não foram compilados, devido à sua baixa representatividade perante o total de respondentes.

A pesquisa foi realizada com as turmas da área de Computação e Informática, na disciplina Inteligência Artificial, por três motivos principais:

1. Facilidade de acesso do autor dessa pesquisa às turmas;
2. Trata-se do mesmo conteúdo que é aplicado nas turmas de Engenharia de Produção (público alvo da ferramenta);
3. As turmas de Ciência da Computação possuem conhecimentos para avaliar o OA do ponto de vista técnico, levando em consideração os aspectos de desenvolvimento de sistemas. Podemos citar como exemplo, o caso do aluno RC, o qual cita “Não é possível maximizar e redimensionar a janela principal. Recomendo usar a interface padrão do sistema operacional para herdar as características do S.O corrente:

//seta o look and feel do SO corrente

```

try {
    UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
} catch (Exception e)
    {e.printStackTrace();}

//abre o jframe maximizado

principal.setExtendedState(JFrame.MAXIMIZED_BOTH);".

```

As questões relativas ao uso do TASNOP foram separadas em dois grupos. No primeiro foram avaliados os quesitos usabilidade e a interface, enquanto no segundo grupo foram avaliados o nível de contribuição para a aprendizagem, o potencial da ferramenta para ser usada como um ambiente computacional de ensino e a confiabilidade dos resultados. Uma breve descrição de cada um dos itens avaliados é apresentada a seguir:

- Usabilidade. Este item teve como objetivo a avaliação da capacidade do TASNOP de ser compreendido, aprendido, operado e atraente ao usuário, quando usado sob as condições especificadas. Ele foi inserido no questionário por se tratar de um elemento importante na avaliação de qualidade de software, conforme especifica a norma NBR ISO/IEC 9126-1 de 2003 (NBR, 2003).
- Interface. Este item teve como objetivo avaliar a interface do TASNOP como meio que estabelece a interação entre o aluno e o OA. O modo como *layouts* de tela, menus, advertências e *helps* são projetados determina quão eficiente é uma interface e consequentemente o quão competente o produto será no meio pretendido (PONTES, ORTH, 1999);
- Nível de contribuição para a aprendizagem dos algoritmos. Este item teve como objetivo avaliar o quanto o TASNOP contribuiu, na percepção do aluno, no seu processo de aprendizagem dos algoritmos.
- Potencial para ser usado como um ambiente computacional de ensino. Este item teve como objetivo avaliar o OA sob o ponto de vista do potencial pedagógico computacional (PPC), o qual requer apenas que sejam avaliados

aspectos imediatos da aprendizagem que um recurso pedagógico se propõe (UCHOA, 2003);

- Confiabilidade dos resultados. Este item teve como objetivo avaliar a capacidade do produto de software de manter um nível de desempenho especificado, quando usado em condições especificadas. Ele foi inserido no questionário por se tratar, assim como a usabilidade, de um elemento importante na avaliação da qualidade de software (NBR, 2003).

Para avaliar cada um dos cinco itens acima descritos, optou-se por apresentar aos alunos o questionário com base na escala Likert, criada por Likert (1932) com o intuito de realizar a medição de atitudes. De acordo com Mattar (2005), essa escala contém uma série de afirmações sobre o objeto pesquisado, onde o respondente não apenas concorda ou discorda das afirmações, mas indica o grau de concordância ou discordância.

A escala Likert pode ter diversos itens em sua escala, variando de um trabalho para outro, o que pode ser um problema, haja visto que o entrevistado é quem interpretará essas escalas. Quando o entrevistado analisa os itens da escala, ele interpreta a mesma a sua maneira, o que pode comprometer a validade da pesquisa. Com isso, quanto mais itens existem em uma escala de Likert, mais complexa a mesma se torna, assim como aumenta a complexidade para o respondente (CAMPBELL, 1988).

Por conta disso, e a fim de minimizar a complexidade do questionário, nesse trabalho empregou-se a escala Likert de 5 (cinco) pontos associados a notas variando de 1 a 10, conforme ilustra a figura 2, que é mais familiar aos alunos que participaram da pesquisa.

Figura 2 – Escala de Likert de 5 pontos



Fonte: autor

Conforme Morettin e Bussab (2004), população é o conjunto de todos os elementos ou resultados sob investigação, amostra é qualquer subconjunto da população e amostragem é a maneira que será usada para selecionar uma amostra.

Com base nesses conceitos, pode-se considerar que nesta pesquisa há duas amostras: uma de 78 alunos que resolveram o exercício sem o uso do TASNOP e uma outra de 51 alunos que usaram o AO e responderam o questionário.

É importante ressaltar que os conceitos teóricos sobre os problemas de Caminho Mínimo, Fluxo Máximo e Caixeiro Viajante, além dos algoritmos, haviam sido previamente explicados pelos professores no decorrer das aulas que antecederam o uso do TASNOP. Não obstante, o uso da escala Likert para responder ao questionário também foi explicado pelo professor em sala de aula.

Os resultados bem como algumas conclusões obtidas a partir das respostas dos alunos estão detalhados no próximo capítulo.

4. RESULTADOS

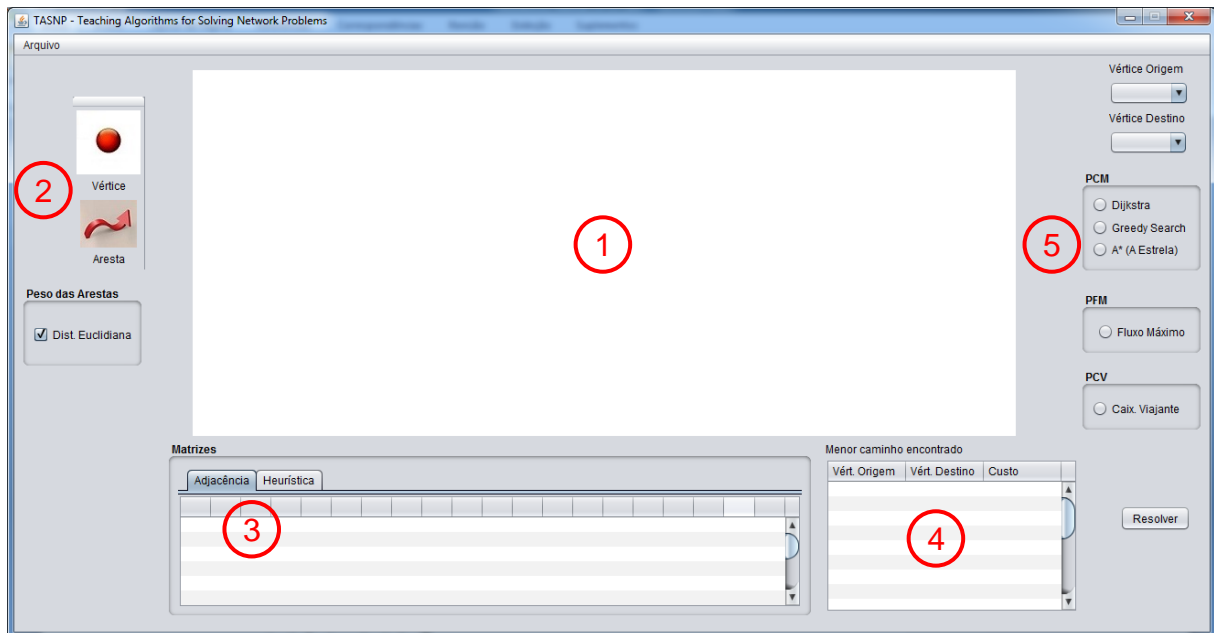
Neste capítulo apresentam-se o ambiente desenvolvido bem como os resultados obtidos nos experimentos realizados.

4.1 DETALHAMENTO DO OBJETO DE APRENDIZAGEM PROPOSTO

O OA desenvolvido e denominado TASNOP – *Teaching Algorithms for Solving Network Optimization Problems*, consiste de uma ferramenta computacional com uma *interface* intuitiva (figura 3), recursos e propriedades que permitem ao aluno desenhar e manipular o grafo que representa o problema a ser resolvido, usando um dos algoritmos disponíveis (A*, Busca Gulosa, Dijkstra, Ford-Fulkerson e Vizinho mais Próximo). Ele interpreta interações, reações e comportamentos do algoritmo escolhido, de acordo com o grafo desenhado e poderá ser acessada, desde que exista uma conexão com a internet, por qualquer máquina e em qualquer localidade, via *web*, sem a necessidade de uma instalação prévia.

O TASNOP pode ser caracterizado como um OA, uma vez que atende aos requisitos descritos na seção 2.1.1 (que apresenta as definições e características dos OA).

Figura 3 – Interface do TASNOP



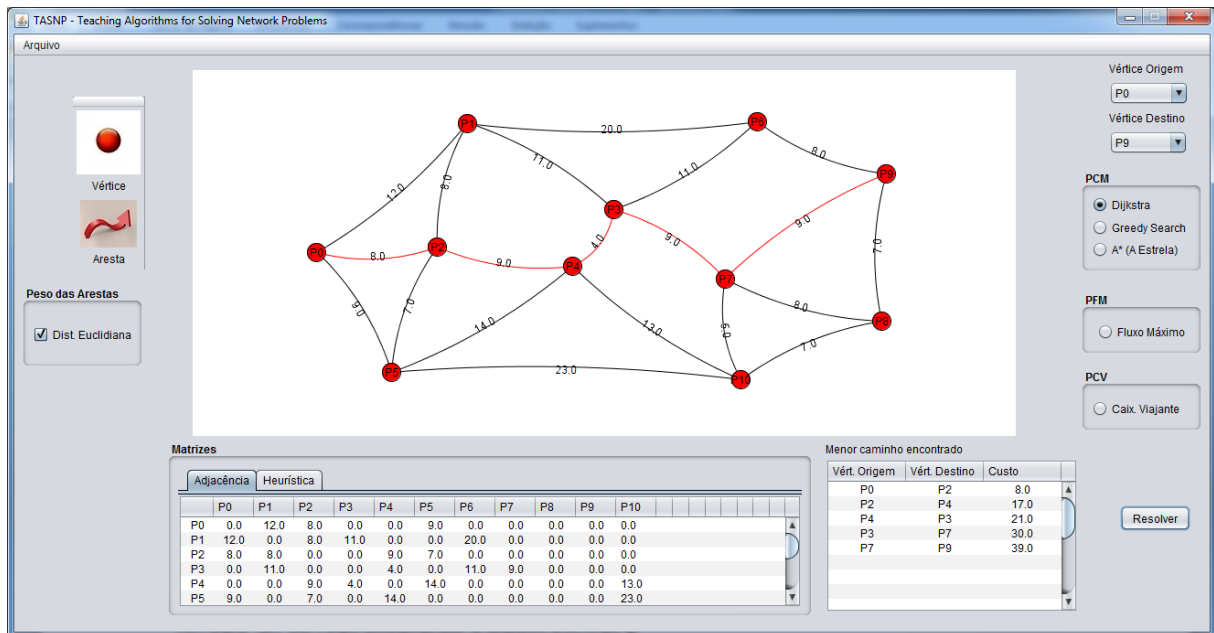
A interface do TASNOP consiste basicamente em:

- Um painel de simulação (área para desenho do grafo) (1);
- Um menu de ferramentas (2) contendo o botão para incluir vértices (nós), o botão para incluir arestas e a opção de recalculer o valor das arestas automaticamente ao arrastar os nós;
- Abas para exibir a matriz de adjacência e a matriz heurística (3);
- Uma tabela para exibir a solução (caminho ou fluxo, dependendo do problema) (4);
- Um menu de opções para selecionar os nós de origem e destino, além do algoritmo a ser usado na solução do problema (5).

No desenvolvimento da *interface*, foi levada em consideração a necessidade de facilidade de interação entre o aluno e o OA. O painel de simulação permite a interação do aluno com o grafo desenhado, ou seja, inclusão, exclusão ou alteração dos elementos (nós e arestas) por meio do uso do *mouse*. Depois de selecionados os nós de origem e destino, o TASNOP apresenta o caminho ou indica o fluxo (no caso do Problema de Fluxo Máximo) calculado com base no algoritmo escolhido.

Na figura 4 é possível verificar no exemplo de uso do TASNOP que o exercício proposto aos alunos (figura 1) foi resolvido usando o algoritmo Dijkstra, que é o mais adequado para solução do problema. Neste exemplo, é possível visualizar o caminho destacado em vermelho, o qual representa a solução ótima do problema bem como o custo desse caminho. É possível observar ainda, na parte inferior, a matriz de adjacências que representa o grafo.

Figura 4 – Tela do TASNOP com a solução do exercício empregado na sua avaliação



Por fim, considerando a proposta de padrão IEEE 1484, usualmente referida como *Learning Object Metadata* (LOM) e que contém um conjunto de atributos conhecidos como metadados para categorizar objetos de aprendizagem em categorias, apresentam-se, no quadro 4, os metadados do TASNOP. Eles descrevem características relevantes que servem para catalogação do OA em repositórios de objetos educacionais.

Quadro 4 - Conjunto de metadados do TASNOP

Atributo	Categoria	Conteúdo
Descrição	Geral	Consiste de uma ferramenta computacional, a qual procura fornecer uma <i>interface</i> intuitiva, recursos e propriedades que permitem ao aluno desenhar e manipular grafos para representar problemas de caminho mínimo, fluxo máximo ou caixeiro viajante, usando um dos algoritmos disponíveis (A*, Busca Gulosa, Dijkstra, Ford-Fulkerson e Vizinho mais Próximo)
Versão	Ciclo de Vida	1.0 – Janeiro/2015.

Referência para catalogação	Metadados	Problemas de Otimização em Redes; Caminho Mínimo; Fluxo Máximo; Caixeiro Viajante. Algoritmos Dijkstra; A*; Busca Gulosa; Vizinho mais Próximo
Tipo de Tecnologia	Técnica	JAVA versão 1.7.0_67-b01; API JUNG
Público alvo	Educacional	Ensino Superior; Séries finais; Otimização; Pesquisa Operacional; Inteligência Artificial; Engenharia; Computação; Logística; Administração.
Direito autoral e Condições de uso	Direitos	O autor autoriza, sob licença <i>Creative Commons</i> – Atribuição 2.5 Brasil, cópia, distribuição, exibição e execução do material desenvolvido de sua titularidade, sem fins comerciais, assim como a criação de obras derivadas, desde que se atribua o crédito ao autor original da forma especificada por ele ou pelo licenciante, assim como a obra deverá compartilhar Licença idêntica a esta. Estas condições podem ser renunciadas, desde que se obtenha permissão do autor. O não cumprimento desta Licença acarretará nas penas previstas pela Lei nº 9.610/98
Observações	Anotação	A execução depende da instalação do JVM – <i>Java Virtual Machine</i>
Identificação	Classificação	TASNOP

Fonte: autor

4.2 EXPERIMENTOS REALIZADOS PARA AVALIAÇÃO DO OA PROPOSTO

Nos experimentos para a avaliação do TASNOP, aplicou-se o exercício de otimização em redes juntamente com o questionário para coleta de dados sobre seu uso, apresentados na figura 1, ao conjunto de 51 alunos, conforme detalhado na seção 3.3.

Sobre a realização do exercício com o uso do TASNOP, pode-se notar que 5 alunos (ou seja, 9,8%) indicaram o algoritmo de Busca Gulosa como sendo o algoritmo correto para resolver o problema e, dos outros 46 alunos que escolheram o algoritmo correto, 6 (ou seja, 13%) não encontraram a solução ótima. Cabe destacar

que o algoritmo mais adequado para solução do problema é o algoritmo de Dijkstra e a solução ótima consiste no caminho P1, P2, P4, P3, P7 e P9 com custo igual a 39.

É importante ressaltar que dos 78 (setenta e oito) alunos que fizeram o exercício sem o uso do TASNOP, 21 (ou seja, 26,9%) indicaram o algoritmo de Busca Gulosa como sendo o mais adequado para resolver o problema e, dos outros 57 alunos que escolheram o algoritmo correto, 23 (ou seja, 40,3%) não encontraram a solução ótima. Com base nestes dados, pode-se observar que o rendimento dos alunos foi melhor com o uso do TASNOP, o que justifica sua contribuição no ensino.

Com relação a aplicação do questionário, os resultados obtidos para cada um dos itens são discutidos a seguir. Também é apresentada nesta seção uma sumarização dos resultados considerando todos os itens (figura 10) e também uma discussão sobre as sugestões dos alunos para melhoria do TASNOP.

Cabe destacar que o OA ainda estava em desenvolvimento quando foi usado no segundo semestre de 2014. Portanto, itens como help, a possibilidade de arrastar os nós sem alterar o valor das arestas e a possibilidade de salvar/carregar os grafos ainda não haviam sido implementados, impactando na avaliação dos alunos.

4.2.1 Usabilidade

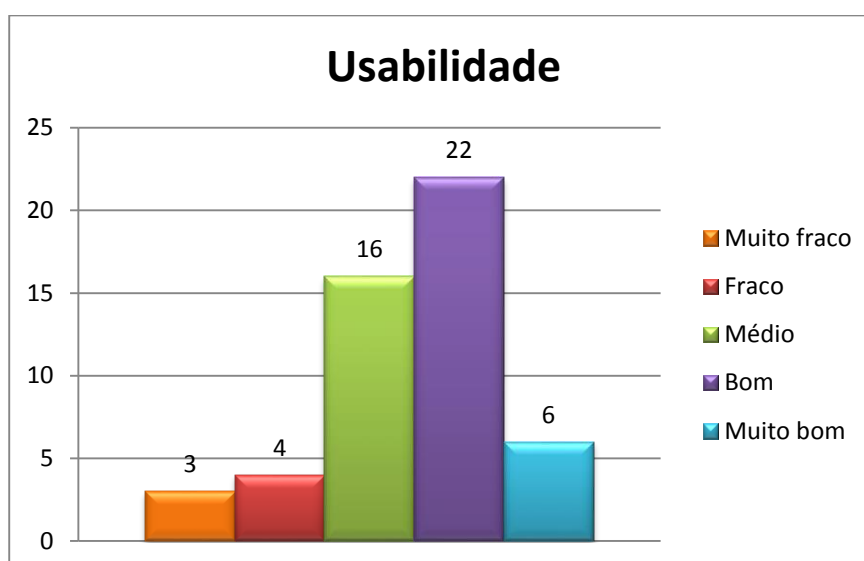
Aqui se esperava que os alunos avaliassem a facilidade de interação entre eles e o TASNOP, e se eles tiveram uma experiência simples e intuitiva.

De acordo com os resultados apresentados na tabela 1 e na figura 5, pode-se notar que a usabilidade foi o item que recebeu o pior resultado. Somando os votos bons e muito bons chegou-se a 54,9%, o que aponta a necessidade de melhorias neste quesito.

Tabela 1 – Usabilidade do TASNOP

Usabilidade		
Classificação	Respostas	Porcentagem
Muito fraco	3	5,9%
Fraco	4	7,8%
Médio	16	31,4%
Bom	22	43,1%
Muito bom	6	11,8%
TOTAL	51	100%

Figura 5 - Gráfico das respostas para o item “Usabilidade”



4.2.2 Interface

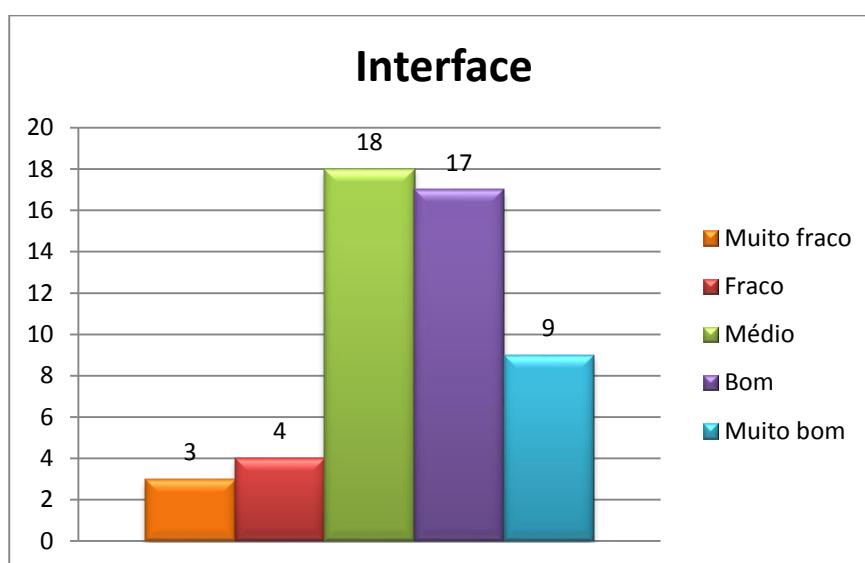
A ideia aqui era que o aluno avaliasse a tela do TASNOP a fim de identificar a disposição dos objetos na mesma e a facilidade de acesso aos seus recursos.

Com base nos resultados descritos na tabela 2 e na Figura 6, nota-se que 51% acharam o TASNOP bom ou muito bom neste quesito. A interface bem como a usabilidade, receberam as menores pontuações, possivelmente devido à falta de um *help*. Algumas sugestões dos alunos como o *Help*, a possibilidade de se optar pelo recálculo das arestas ou a possibilidade de salvar os grafos, foram implementados após a realização dos testes.

Tabela 2 – Interface do TASNOP

Interface		
Classificação	Respostas	Porcentagem
Muito fraco	3	5,9%
Fraco	4	7,8%
Médio	18	35,3%
Bom	17	33,3%
Muito bom	9	17,7%
TOTAL	51	100%

Figura 6 - Gráfico das respostas para o item “Interface”



4.2.3 Contribuição para aprendizagem dos algoritmos

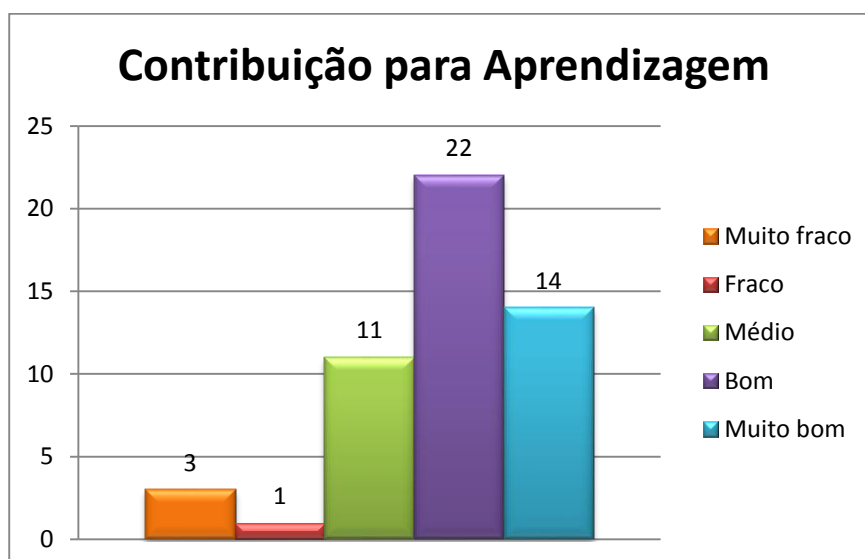
Nesse tópico, esperava-se que o aluno avaliasse se o TASNOP contribuiu na sua aprendizagem, permitindo maior compreensão dos conceitos passados anteriormente pelos professores.

Em geral, os alunos que usaram o TASNOP, indicaram que ele contribuiu substancialmente para a sua aprendizagem. Como mostrado na tabela 3 e na figura 7, 70,6% dos alunos avaliaram a contribuição do TASNOP na sua aprendizagem como bom ou muito bom. Considerando também a quantidade de alunos que classificaram o TASNOP como médio, então a percentagem de sua aceitação sobe para 92,2%.

Tabela 3 - Tabulação das respostas para o item Contribuição para a aprendizagem dos algoritmos

Contribuição para aprendizagem		
Classificação	Respostas	Porcentagem
Muito fraco	3	5,9%
Fraco	1	1,9%
Médio	11	21,6%
Bom	22	43,1%
Muito bom	14	27,5%
TOTAL	51	100%

Figura 7 – Gráfico das respostas para o item “Contribuição para a aprendizagem dos algoritmos”



4.2.4 Potencial para uso como um objeto de aprendizagem

Nesse tópico, o objetivo foi avaliar o TASNOP como um OA adequado a ser incorporado no processo de aprendizagem dos algoritmos contemplados pelo mesmo.

Neste sentido, a pesquisa indicou que o TASNOP tem um potencial promissor para ser utilizado como material didático. No geral, ele agradou a maioria dos alunos, pois 74,5% o classificaram como bom ou muito bom, como pode ser visto na tabela 4 e na figura 8.

Tabela 4 – Potencial para ser um objeto de aprendizagem

Potencial para ser um ambiente computacional de ensino		
Classificação	Respostas	Porcentagem
Muito fraco	3	5,9%
Fraco	3	5,9%
Médio	7	13,6%
Bom	14	27,5%
Muito bom	24	47,1%
TOTAL	51	100%

Figura 8 - Gráfico das respostas para o item “Potencial para ser um objeto de aprendizagem”



4.2.5 Confiabilidade dos resultados

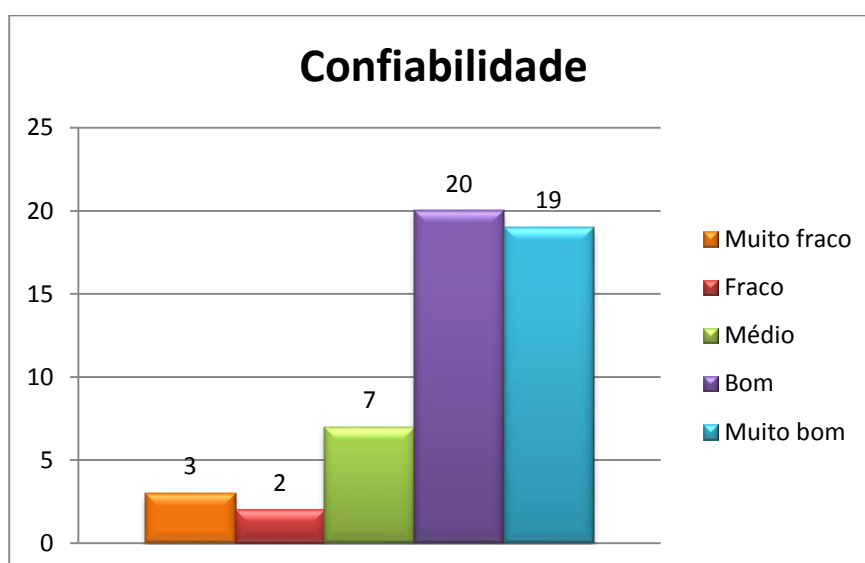
Nesse tópico, o objetivo foi avaliar se os resultados apresentados no TASNOP eram os mesmos resultados obtidos pelos alunos quando da realização do exercício sem o uso do OA.

Os alunos apontaram um grau satisfatório de confiabilidade no TASNOP. Como mostrado na tabela 5 e na figura 9, 76,5% dos estudantes o classificaram como um OA confiável (bom ou muito bom), com bom funcionamento e com resultados confiáveis.

Tabela 5 – Confiabilidade dos resultados apresentados pelo TASNOP

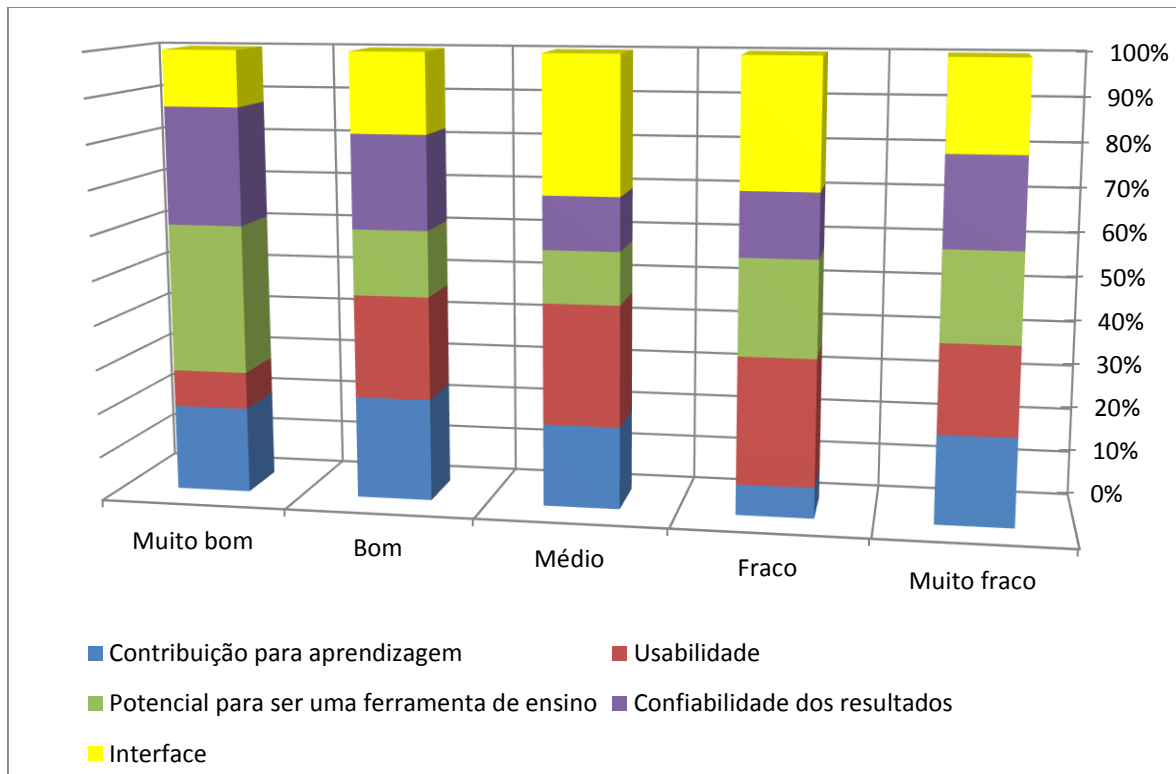
Confiabilidade dos resultados		
Classificação	Respostas	Porcentagem
Muito fraco	3	5,9%
Fraco	2	3,9%
Médio	7	13,7%
Bom	20	39,2%
Muito bom	19	37,3%
TOTAL	51	100%

Figura 9 - Gráfico das respostas para o item “Confiabilidade dos resultados”



Na Figura 10 pode-se verificar um resumo dos resultados descritos nas tabelas 1 a 5, sendo possível notar que a porcentagem de alunos que classificaram o TASNOP como fraco ou muito fraco é muito baixa. Por outro lado, a porcentagem de classificações como média indicava a necessidade de melhorias, muitas das quais feitas após a realização dos testes.

Figura 10 - Sumarização dos resultados



Por meio da sumarização e análise de todos os resultados, pode-se demonstrar que o TASNOP contribui para o ensino dos algoritmos solucionadores de POR abordados nessa pesquisa e também que ele tem potencial para ser um AO, pois possibilita a realização dos exercícios em uma plataforma que permite representação dos problemas por meio de grafos, a visualização dos resultados de maneira ágil, além de alterações das variáveis do experimento que está sendo realizado, em tempo real. Além disso, pode-se verificar que os resultados são confiáveis pois entregam ao aluno o mesmo resultado que seria encontrado caso o exercício fosse feito sem o uso do computador. Porém, sua interface precisa de alterações para melhorar sua usabilidade, trazendo aos alunos uma experiência mais amigável e intuitiva.

No questionário apresentado aos alunos que usaram o OA havia uma questão pedindo que eles citassem 3 (três) itens que eles mudariam ou acrescentariam no OA com o objetivo de melhorá-lo, sendo uma questão livre, e os mais citados foram:

- Melhorar a interface do AO (25,5% dos alunos). Melhorias na interface, como a possibilidade de arrastar os nós sem alterar o valor das arestas que estão ligadas a esses nós e a possibilidade de salvar e/ou carregar um grafo, já foram implementadas.
- Acrescentar mais algoritmos de busca (23,5% dos alunos). Essa solicitação encontra-se pendente.
- Apagar todo o grafo com apenas um clique (11,8% dos alunos). Essa solicitação encontra-se pendente.

5. CONCLUSÕES

Nessa pesquisa foi desenvolvida uma ferramenta computacional, caracterizada como um Objeto de Aprendizagem, para contribuir na aprendizagem dos algoritmos A* (estrela), Busca Gulosa, Dijkstra, Ford-Fulkerson e Vizinho mais Próximo, utilizados para solução de alguns problemas de otimização em redes (POR). Tal pesquisa contribui academicamente para suprir uma demanda de ferramentas que auxiliem no ensino dos algoritmos mencionados, principalmente nas áreas de Engenharia e Computação.

Os experimentos realizados demonstraram que o TASNOP contribuiu para a o entendimento dos alunos sobre os algoritmos, uma vez que o percentual de acertos na escolha do algoritmo correto para a resolução do problema proposto passou de 73,1% antes da utilização do OA para 90,2% após a sua utilização.

Os alunos que utilizaram o TASNOP sugeriram melhorias técnicas e a implementação de novas funcionalidades. Dentre elas, podemos destacar a melhoria da usabilidade, a criação de um help, a possibilidade de arrastar os nós sem alterar o valor das arestas e a possibilidade de salvar e carregar os grafos. Infelizmente tais melhorias só foram implementadas após a utilização do OA e do feedback dos alunos. Ainda assim, os alunos sentiram que a inserção do TASNOP no ambiente de ensino contribuiu no processo de aprendizagem.

Para continuidade dessa pesquisa sugerem-se: melhorar a interface do TASNOP com base nas sugestões apresentadas pelos alunos; acrescentar mais algoritmos de busca; realizar mais experimentos com a nova versão do TASNOP envolvendo cursos de graduação e pós-graduação das áreas de Engenharia e Informática; implementar um mecanismo que demonstre, passo a passo, a construção da solução sugerida pelo algoritmo e, por fim, incorporar o TASNOP em repositórios de objetos educacionais como, por exemplo, a Rede Interativa Virtual de Educação (RIVED) e/ou o Banco Internacional de Objetos Educacionais (BIOE).

REFERÊNCIAS BIBLIOGRÁFICAS

AHUJA, Ravindra K.; MAGNANTI, Thomas L.; ORLIN, James B. **Network flows – theory, algorithms, and applications**. 1. ed. New Jersey: Prentice–Hall, 1993. 864 p.

ALARCÃO I.. **Reflective teachers in a school reflective (4th ed.)**. São Paulo: Cortez, 2003.

ALAVI, M.; LEIDNER, D. E. knowledge management and knowledge management systems: conceptual foundations and research issues. **MIS QUARTERLY**, Minneapolis, v. 25, n. 1, p. 107-136, mar. 2001.

ALMEIDA, C. Z.; VIEIRA, M. B.; LUCIANO, N. A.. Ambiente virtual de aprendizagem: uma proposta para autonomia e cooperação na disciplina de informática. **XII SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO**, Vitória, p. 431-439, 2001.

ALSTES, A.; LINQDVIST, J.. VERKKOKE: learning routing and network programming online. **12TH ANNUAL CONFERENCE ON INNOVATION AND TECHNOLOGY IN COMPUTER SCIENCE EDUCATION, ITICSE'07**, Dundee, v.39, n. 4, p. 91-05, set. 2007.

APPOLINÁRIO, F. **Metodologia da Ciência – filosofia e prática da pesquisa**. São Paulo: Editora Pioneira Thomson Learning, 2006.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR ISO/IEC 9126-1**: Engenharia de Software: Qualidade de Software – Parte 1: Modelo de Qualidade. Rio de Janeiro, 2003. 21 p.

ARENALES, Marcos; ARMENTANO, Vinícius; MORABITO, Reinaldo; YANASSE Horácio. **Pesquisa operacional**. Rio de Janeiro: Elsevier, 2007. 523 p.

ATZINGEN, J. V; CUNHA, C. B.; NAKAMOTO, F. Y.; RIBEIRO, F. R.; SCHARDONG, A.. Análise comparativa de algoritmos eficientes para o problema de caminho mínimo. **XXI ANPET: CONGRESSO DE PESQUISA E ENSINO EM TRANSPORTES**, Rio de Janeiro, .2007. p. 1-12.

BALLOU, Ronald H. **Business logistics: supply chain management**. 5 ed. New Jersey: Prentice-Hall, 2003. 816 p.

BARBOSA, Cláudia Maria Arôso Mendes. A aprendizagem mediada por: interação e cognição em perspectiva. **Revista Brasileira de Aprendizagem Aberta e a Distância**, São Paulo, v. 11, p.83-100, 2012.

BARROS, E. A. R.; PAMBOUKIAN, S. V. D.; ZAMBONI, L. C.. Algoritmo de Dijkstra: apoio didático e multidisciplinar na implementação, simulação e utilização computacional. **INTERNATIONAL CONFERENCE ON ENGINEERING AND COMPUTER EDUCATION**, São Paulo, mar. 2007.

BENNETT, S.; LOCKYER, L.. Becoming an online teacher adapting to a changed environment for teaching and learning in higher education. **INTERNATIONAL COUNCIL FOR EDUCATIONAL MEDIA**, Wollongong, v. 41, n. 3, p. 231-248, jan. 2004.

BERÓN, M. M.; HENRIQUES, P. R.; PEREIRA, M. J.. A system for evaluate and understand routing algorithms. **II CONFERENCIA NACIONAL EM INTERACÇÃO PESSOA-MÁQUINA**, Braga, 2006.

BERTSEKAS, Dimitri P. **Network optimization: continuous and discrete models**. Belmont: Athena Scientific, 1998. 585 p.

BEYER, Kevin; GOLDSTEIN, Jonathan; RAMAKRISHNAN, Raghu, SHAFT, Uri. When is “nearest neighbor” meaningful?. **Database Theory—ICDT’99**. p. 217-235, 1999.

BRIDGEMAN, S.; GOODRICH, M. T.; KOBOUROV, S. G.; TAMASSIA, R.. PILOT: an interactive tool for learning and grading. **THIRTY-FIRST SIGCSE TECHNICAL SYMPOSIUM ON COMPUTER SCIENCE EDUCATION**, Austin, p. 139-143, mar. 2000.

CALADO, Fabiano das Mercês; LADEIRA Ana Paula. Problema do caixeiro viajante: um estudo comparativo de técnicas de inteligência artificial. **Revista Exacta**, Belo Horizonte, v.4, n.1, 2011. Disponível em <<http://revistas.unibh.br/index.php/dcet/article/view/304/163>>. Acesso em 17 nov. 2014.

CAMPBELL, Donald J. Task complexity: a review and analysis. **Academic Management Review**. v. 13, n. 1, p. 40-52, 1988.

CAMPOS, Vânia Barcellos G. **Algoritmos para Resolução de Problemas em Redes**. Rio de Janeiro, 28 p. Apostila do Curso de Pós Graduação em Engenharia de Transportes – Instituto Militar de Engenharia, 2013.

CORMEN, Thomas H.; LEISERSON, Charles E.; RIVEST, Ronald L.; STEIN, Clifford. **Introduction to algorithms**. 3 ed. Massachusetts: MIT Press, 2009.

GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas, 2008.

GIBBONS, Andrew S.; NELSON, Jon; RICHARDS, Robert. The Nature and origin of instructional objects. **Association for Educational Communications and Technology**, Bloomington, 2000.

GOLDBARG, Marcos Cesar; LUNA, Henrique Pacca L. **Otimização combinatória e programação linear: modelos e algoritmos**. 2. ed. Rio de Janeiro: Campus, 2005. 536 p.

HAN, Shengwei; PENG, Zixiong; WANG, Shunqin. The maximum flow problem of uncertain network. **Information Sciences**, v. 265, n. 1, p. 167-175, 2014.

HODGINS, H. W.. The future of learning objects, **PROCEEDINGS OF THE 2002 ETEE CONFERENCE**, Davos, p. 76-82, aug. 2002.

IEEE. Institute of Electrical and Electronics Engineers. **WG12: Learning Object Metadata**. 2005. Disponível em: <<http://grouper.ieee.org/groups/ltsc/wg12/>>. Acesso em: 18 dez. 2014

IQBAL, M. Ashraf; TAHIR, Sara. Should we teach algorithms? **Iranian Journal of Electrical and Computer Engineering**, Iran, v. 2, n. 2, p. 134-140, 2003.

KELLER, Christina. User acceptance of virtual learning environments: a case study from three northern european universities. **Communications of the Association for Information Systems**, Sweden, v. 25, n. 38, p. 465-486, nov. 2009.

LAKATOS, Eva Maria; MARCONI, Marina de Andrade. **Técnicas de pesquisa: planejamento e execução de pesquisas, amostragens e técnicas de pesquisas, elaboração, análise e interpretação de dados**. 3. ed. São Paulo: Atlas, 1996.

LAWRENCE, A. W.; BADRE, A. M.; STASKO, J. T.. Empirically evaluating the use of animations to teach algorithms. **PROCEEDINGS OF THE 1994 IEEE SYMPOSIUM ON VISUAL LANGUAGE**, Los Alamitos , p. 48-54, 1994.

LÁZARO, A. C.; TEZANI, T. C. R.. Educação a distância: uma modalidade de ensino e inúmeras possibilidades de aprendizagem. **SIED: ENPED-SIMPÓSIO INTERNACIONAL DE EDUCAÇÃO A DISTÂNCIA E ENCONTRO DE PESQUISADORES EM EDUCAÇÃO A DISTÂNCIA**, São Carlos, set. 2014.

LIMA, S. J. A.; ARAÚJO, S. A. A computational tool for helping to teach routing algorithms. **22ND INTERNATIONAL CONFERENCE ON PRODUCTION RESEARCH (ICPR 2013)**, Foz do Iguaçu, v. 1. p. 1-5, 2013.

MALMI, Lauri; KARAVIRTA, Ville; KORHONEN, Ari; NIKANDER, Jussi; SEPPÄLÄ, Otto; SILVASTI, Panu. Visual algorithm simulation exercise system with automatic assessment: TRAKLA2. **Informatics in Education**, Vilnius, v.3, n. 2, p. 267-288, 2004.

MASON, Robin; PEGLER, Chris; WELLER, Martin. A learning object success story. **Journal of Asynchronous Learning Networks**, v.9, n.1, p. 97-105, 2005.

MATTAR, Fauze Najib. **Pesquisa de marketing: metodologia e planejamento**. 6. ed. São Paulo: Atlas, 2005.

MAYER, Richard E.; MORENO, Roxana. Aids to computer-based multimedia learning. **Learning and Instruction**, v. 12, n. 1, p. 107-119, 2002.

MCGREAL, Rory. **Online education using learning objects**. New York: Psychology Press, 2004.

MELO, F. G. O.; BARROS B. R.; MORAES A. S.. Educational Objects with Geogebra for Aid to Pedagogical Practices in Engineering. **PROC. OF 15TH BRAZILIAN CONGRESS ON ENGINEERING EDUCATION – COBENGE**. Belém (PA), Brazil, 1-12, 1997.

MENDES, José Fernando F. Física de redes complexas. **Gazeta da Física**, Coimbra, v. 28, n. 4, p. 10-16, 2005.

MIGUEL, Paulo Augusto Cauchick; FLEURY, Afonso; MELLO, Carlos Henrique Pereira; NAKANO, Davi Novboru; LIMA, Edson Pinheiro de; TURRIONI, João Batista; HO, Linda Lee; MORABITO, Reinaldo; MARTINS, Roberto Antônio; SOUSA,

Rui; COSTA, Sérgio E. Gouvêa da; PUREZA, Vitória. **Metodologia de pesquisa em Engenharia de Produção e Gestão de Operações**. 2. ed. Rio de Janeiro: Elsevier, 2012.

MORETTIN, Pedro A.; BUSSAB, Wilton de O. **Estatística básica**. 5. ed. São Paulo: Saraiva, 2004. 526 p.

NGAI, Eric W. T.; POON, J.; CHAN, Y. H. C. Empirical examination of the adoption of WebCT using TAM. **Computers & Education**, v. 48, n. 2, p. 250-267, feb. 2007.

PAULA, L. Q.; PIVA, Jr. D.; FREITAS R. A.. Importance of Reading and Problem Abstraction in the formation process of reasoning logical-abstract in students of Computer Science. **PROC. OF XII WORKSHOP ON COMPUTING IN EDUCATION**, Bento Gonçalves, 2009

PEGDEN, C. D.; SHANNON, R. E.; SADOWISKI, R. P.. **Introduction to Simulation Using SIMAN**. New York, NY: McGrawHill, 1990.

PONTES, A. M.; ORTH, A. I. Uma proposta de interface de software orientada à linguagem de sinais. **WORKSHOP SOBRE FATORES HUMANOS EM SISTEMAS COMPUTACIONAIS**, Campinas, 1999.

RIVED - **Rede interativa virtual de educação, conheça o RIVED**, disponível em <http://rived.mec.gov.br/site_objeto_lis.php>. Acesso em 18 dez. 2014.

RUSSEL, Stuart J.; NORVIG, Peter. **Artificial intelligence a modern approach**. New Jersey: Prentice Hall, 1995.

SAADE, R.; NEBEBE, F.; MAK, T.. Knowledge management systems development: theory and practice. **INTERDISCIPLINARY JOURNAL OF INFORMATION, KNOWLEDGE, AND MANAGEMENT**, v.6, p. 35-72, 2011.

SANTOS, M. P. **Pesquisa operacional**. Universidade do Estado do Rio de Janeiro, 2003.

SEVASTIAN, Paula; RUS, Andrei Bogdan; DOBROTA, Virgil. Simulation of the Ford-Fulkerson algorithm using Omnet++. **Acta Technica Napocensis Electronics and Telecommunications**, v. 53, n. 2, p. 23-29, 2012.

SHEPERD, Clive. Objects of interest. **Brighton East Sussex: Fastrak Consulting Limited.**, 2000. Disponível em: < <http://www.fastrak-consulting.co.uk/tactix/Features/objects/objects.htm>>. Acesso em: 18 dez. 2014.

SILVA A. N.; MARTINS D. D. S.; NIGRO I. C.. **Computer labs in the teaching of Production Engineering**. INGEPRO–Inovation, Management and Production, 2(12), 23-29, 2010.

SILVA, Edna Lúcia da; MENEZES, Estera Muszkat. **Metodologia de Pesquisa e Elaboração de Dissertação**. Universidade Federal de Santa Catarina, 2001.

SILVA, Eli Lopes da. Uma experiência de uso de objetos de aprendizagem na educação presencial: ação-pesquisa num curso de sistemas de informação. 2006. 145f. Dissertação (Mestrado em Educação) – Pontifícia Universidade Católica de Minas Gerais, Programa de Pós-Graduação em Educação, Belo Horizonte, 2006.

SOLOMON, Marius M. Algorithms for the vehicle routing and scheduling problems with time window constraints. **Operations Research**, v. 35, n. 2, p. 254-265. 1987.

SOUZA, Felipe Ribeiro; MELO, Michel; PINTO, Cláudio Lúcio Lopes. A proposal to find the ultimate pit using Ford-Fulkerson algorithm. **Revista Escola de Minas**, v. 67, n.4, p. 389-395. 2014.

TIGRE, Paulo Bastos; NORONHA, Vitor Branco. Do mainframe à nuvem: inovações, estrutura industrial e modelos de negócios nas tecnologias da informação e da comunicação. **Revista da Administração**, São Paulo, v. 48, n. 1, p. 114-127, jan./mar. 2013.

TORRES, F. A. Análise do uso da sala de tecnologias em escola pública e privada. **Revista Tecnologias na Educação**, v. 3, n. 1, jul. 2011.

TOSCANO, Wagner. Inteligência artificial - busca no espaço de estados. São Paulo, 12 p. Escola Politécnica – Universidade de São Paulo, 2009.

TYLER, Ralph W. **Princípios básicos de currículos e ensino**. 7 ed. Porto Alegre: Globo, 1981. 119p.

UCHOA, Antonio Ribeiro. Organizador prévio virtual para o ensino de física. 2003. Dissertação (Mestrado Profissionalizante em Computação) - Universidade Estadual do Ceará e do Centro Federal de Educação Tecnológica do Ceará, Fortaleza, 2003.

VENKATESH, V.; MORRIS, M. G.; DAVIS, G. B.; DAVIS, F. D.. User acceptance of information technology: toward a unified view. **MIS QUARTERLY**, v. 27, n. 3, p. 425-478, sep. 2003.

WELZL, M.; ALI, M.. Teaching routing with IRV-Tool. **CONFERENCE ICL**, sep. 2006.

WERTHEIN, Jorge. A sociedade da informação e seus desafios. **Revista Ciência da Informação**, Brasília, v. 29, n.2, p. 71-77, 2000.

WILEY, D. A. **The instructional use of learning objects**. 2002. Disponível em: <<http://www.reusability.org/read/>>. Acesso em: 18 dez. 2014.

_____. **On the inanimate nature of learning objects**. 2006. Disponível em: <<http://opencontent.org/blog/archives/244>>. Acesso em: 18 dez. 2014.

YI, M. Y.; HWANG, Y.. Predicting the use of web-based information systems self-efficacy, enjoyment, learning goal orientation, and the technology acceptance model. **INTERNATIONAL JOURNAL OF HUMAN-COMPUTER STUDIES**, v. 59, n. 4, p. 431-449, 2003.

GLOSSÁRIO

AD HOC: tipo de rede redes sem fio que dispensam o uso de um ponto de acesso comum aos computadores conectados a ela, de modo que todos os dispositivos da rede funcionam como se fossem um roteador, encaminhando comunitariamente informações que vêm de dispositivos vizinhos.

ALGORITMO: sequência de passos que mostra os procedimentos necessários para a resolução de uma tarefa.

BROWSER: *software* que permite a navegação na internet (também conhecido como navegador).

CLOUD: utilização da memória e das capacidades de armazenamento e processamento de computadores e servidores compartilhados e interligados por meio da internet.

E-LEARNING: *electronic learning* (ensino eletrônico) - modelo de ensino não presencial suportado por tecnologia.

FEEDBACK: retorno sobre o desempenho apresentado.

FRAMEWORK: conjunto de ferramentas para o desenvolvimento de sistemas.

GRAFO: conjunto de nós ligados por arestas.

HARDWARE: parte física do computador.

INTERFACE: meio de comunicação entre o usuário e o *software*.

LAYOUT: disposição dos componentes na interface do *software*.

METADADO: dados que descrevem outros dados.

MULTIMÍDIA: técnica para apresentação de informações que recorre simultaneamente a diversos meios de comunicação, mesclando texto, som, imagens fixas e animadas.

ON-LINE: conectado diretamente a um computador e pronto para uso.

REAL-TIME: em tempo real.

SOFTWARE: programas instalados no computador; conjunto de componentes lógicos do computador.

APÊNDICE A – Trabalhos resultantes das pesquisas realizadas durante o Mestrado

LIMA, S. J. A.; LOURENÇO, W. S.; ARAÚJO, S. A.. *ESHOPPS: A Computational Tool to Aid the Teaching of Shortest Path Algorithms*. **JESTEC - JOURNAL OF ENGINEERING SCIENCE AND TECHNOLOGY**, jul. 2015.

LOURENÇO, W. S.; LIMA, S. J. A.; ARAÚJO, S. A.. Proposta de Ferramenta Web para Aprendizagem de Algoritmos para Solução do Problema de Caminho Mínimo. **34º ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO (ENEGEP 34)**, 2014, Curitiba, Paraná.

SANTOS, C. S.; LIMA, H. F. H.; SILVA, R. A. R.; SOUZA, R. A.; ARAUJO, S. J.; LOURENÇO, W. S.; ARAUJO, S. A.. Proposta de uma Ferramenta para Auxílio ao Ensino de Algoritmos de Roteirização. **X ENCONTRO DE INICIAÇÃO CIENTÍFICA DA UNINOVE**, 2013.