

# ***A-Graph: Uma ferramenta computacional de suporte para o ensino-aprendizado da disciplina Teoria dos Grafos e seus Algoritmos***

**Luis Arturo Pérez Lozada<sup>1</sup>**

<sup>1</sup>Centro de Matemática, Computação e Cognição – Universidade Federal do ABC  
CEP 09210-580 – Santo André – SP – Brasil

`luis.arturo@ufabc.edu.br`

***Resumo.** Este trabalho tem por objetivo apresentar a ferramenta computacional A-Graph de suporte para o ensino-aprendizado da disciplina Teoria dos Grafos e seus Algoritmos. O A-Graph basicamente é um programa de edição de grafos tanto dirigidos quanto não dirigidos. Entre as principais funcionalidades destaca-se a visualização da topologia subjacente ao grafo desenhado através das matrizes de adjacência e incidência; e a execução dos algoritmos de busca BFS e DFS, permitindo que o usuário defina livremente o vértice inicial da busca. É possível também, a exportação do grafo no formato bitmap **jpg** e no formato vetorial **svg**, além de permitir salvar a geometria e a topologia do grafo desenhado num arquivo de texto ASCII.*

## **1. Cenário de Uso**

*A-Graph* foi concebido com o propósito de oferecer aos discentes dos cursos superiores da área da computação (ciência da computação, engenharia da computação e sistemas de informação, etc.) de um software livre com código fonte aberto, de suporte ao ensino-aprendizado da disciplina de Teoria dos Grafos e de seus Algoritmos.

Os grafos são estruturas computacionais de reconhecida utilidade em Ciência da Computação, mas também em outras áreas como Engenharia e Pesquisa Operacional. Em particular, o *A-Graph* pode ser estendido de maneira tal que os vértices do grafo (representados tradicionalmente por círculos) sejam substituídos por outros símbolos gráficos que representem, por exemplo, roteadores, hubs, workstations, etc. de maneira a utilizar o *A-Graph* em disciplinas como Redes de Computadores onde sua funcionalidade de busca em grafos serve de base na ilustração dos algoritmos de roteamento em redes, por exemplo.

## 2. Desenvolvimento

A metodologia de desenvolvimento de software adotada no *A-Graph* foi a metodologia tradicional em cascata. A figura 1 ilustra o Diagrama de Casos de Uso do *A-Graph*. Pode-se apreciar as principais funcionalidades como: selecionar o tipo de grafo (orientado ou não orientado), a inserção de vértices, arestas (eventualmente laços); e a execução dos algoritmos de busca em grafos (Busca em Largura e Busca na Profundidade) conjuntamente com as funcionalidades para controlar a execução desses algoritmos. Outras funcionalidades secundárias (não constantes nesse diagrama) incluem a exibição da topologia do grafo através das matrizes de incidência e adjacência, assim como a exportação do modelo do grafo no formato vetorial **svg** (vide em Silva da, (2009)) e no formato bitmap **jpg**.

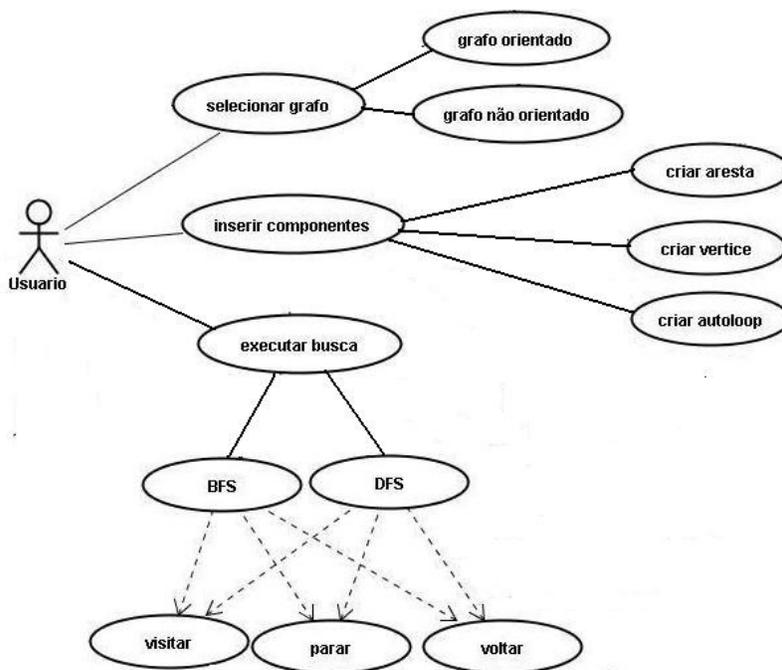


Figura 1: Diagrama de Casos de Uso no *A-Graph*.

O *A-Graph* foi completamente implementado na linguagem de programação Java utilizando recursos básicos de computação gráfica bidimensional suportada pela API Java2D. Não foi realizada a validação formal dos requisitos funcionais de software.

## 3. Apresentação do software

O *A-Graph* roda em modo *stand-alone*; sua interface gráfica (vide figura 1) é bastante simples com uma barra de menus simplificada (**Arquivo** e **Ajuda**) uma barra de ferramentas com principais funcionalidades do aplicativo e por último, abaixo da barra de ferramentas, localiza-se a barra de status para a exibição de dicas e alertas de uso do

aplicativo. Na figura 1, destaca-se a ferramenta **Selecionar Elemento** → **Aresta** (eventualmente laço, caso o grafo selecionado seja Orientado).

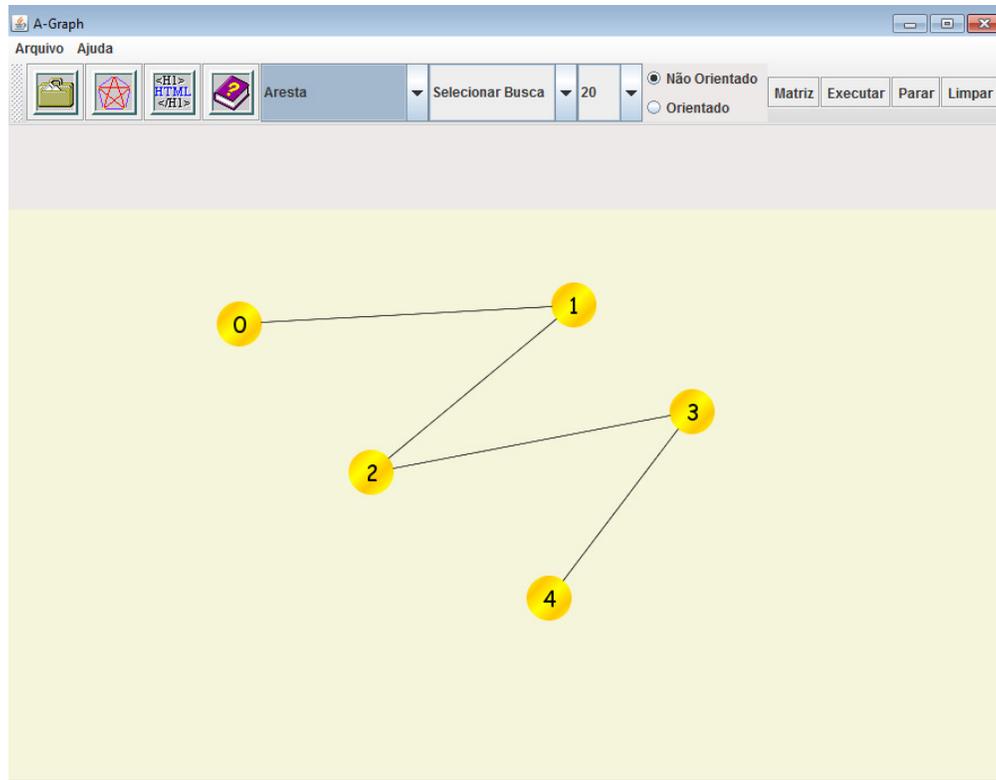


Figura 1: Interface Gráfica do *A-Graph* exibindo as barras de menu, ferramentas, status e a área de edição.

O *A-Graph* é uma ferramenta para o desenho de grafos (vide Dantas do, (1997)), que permite a visualização da topologia de um grafo previamente desenhado. As relações de adjacência e incidência entre os elementos do grafo podem ser exibidas através do botão **Matriz**, especificamente, a **Matriz de Adjacência** e a **Matriz de Incidência**, para maiores informações sugerimos consultar Cormen et. al. (2002) e Szwarcfiter, J. (1986). Em particular, a figura 2 ilustra a topologia do grafo na figura 1.



Figura 2: Topologia de um grafo: (a) Matriz de Adjacência, (b) Matriz de Incidência.

Os algoritmos de busca em grafos podem ser executados através da ferramenta **Selecionar Busca**. A implementação destas funcionalidades baseia-se no trabalho de Prado do, (2009). O *A-Graph* implementa a Busca em Largura **BFS** (*Breadth First Search*) e a Busca em Profundidade **DFS** (*Depth First Search*). Não é nossa intenção entrar em detalhes algorítmicos dessas buscas, existe uma vasta literatura que trata do assunto, como em Cormen et. al. (2002) e Szwarcfiter, J. (1986).

A figura 3 ilustra um grafo orientado com laços onde aplicaremos a busca DFS. Observe nesta figura, na barra de ferramentas, a existência dos botões circulares  **Não Orientado**  **Orientado** que nos permite a escolha do tipo de grafo a desenhar e/ou a transformação da versão orientada para não orientada.

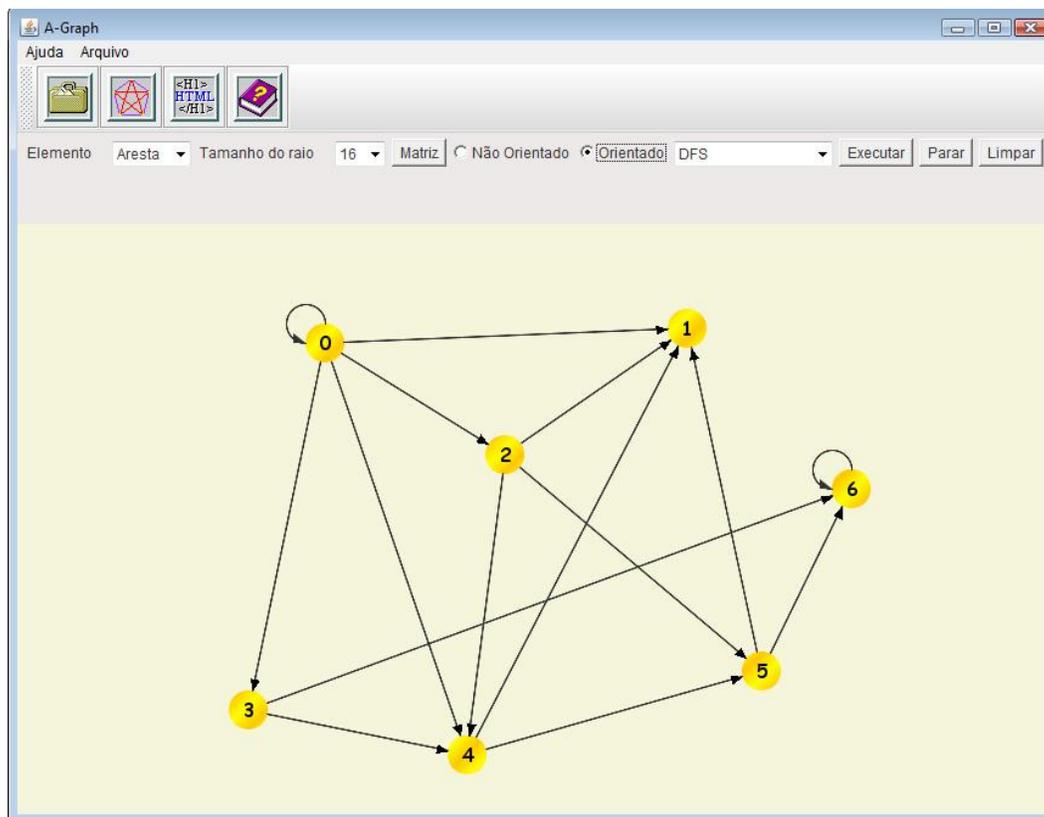


Figura 3: Grafo original antes da Busca em Profundidade.

A figura 4 exibe o grafo de percurso resultante (uma árvore, sendo mais exato), observe que o *A-Graph* exibe os tempos de início e fim da visita em profundidade. Um detalhe de interesse a relatar é a possibilidade do usuário escolher qual é o vértice inicial da busca em profundidade e conseqüentemente a raiz da árvore de percurso.

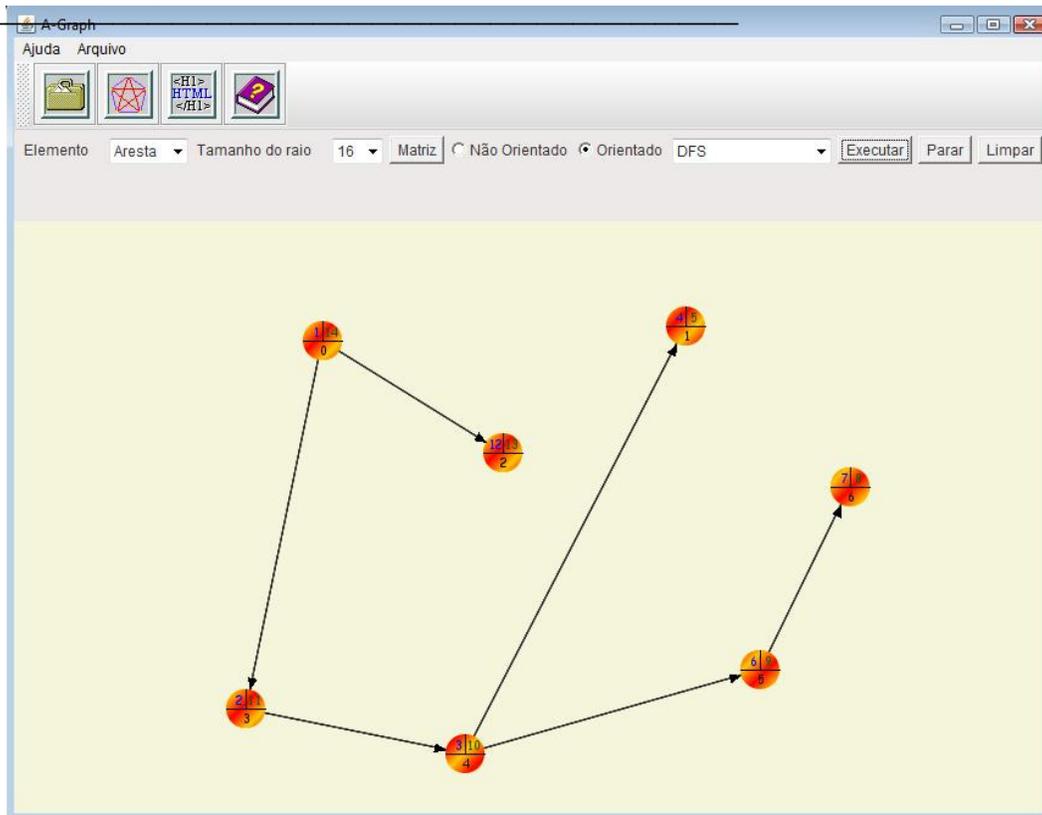


Figura 4: Grafo resultante após a Busca em Profundidade.

O *A-Graph* permite salvar as informações geométricas e topológicas de um grafo num arquivo de texto ASCII utilizando a extensão própria **grp** (vide figura 5). Para esse propósito é necessário a escolha da seqüência **Arquivo** → **Salvar Como** na barra de menus.

```
g1.grp - Bloco de notas
Arquivo Editar Formatar Exibir Ajuda
<grafo>
0
</grafo>
<vertices>
[indice][id][ x][ y][raio]
 0 0 279 172 20
 1 1 546 162 20
 2 2 377 310 20
 3 3 624 253 20
 4 4 531 394 20
</vertices>
<arestas>
[indice][id origem][id destino]
 0 3 4
 1 0 1
 2 2 3
 3 1 2
</arestas>
```

Figura 5: Arquivo ASCII com informações topológicas e geométricas do grafo na figura 1.

Existe também a possibilidade de exportação do grafo desenhado no ambiente *A-Graph* no formato vetorial **svg** e no formato bitmap **jpg**. Para esse propósito é necessário a escolha da seqüência **Arquivo** → **Exportar SVG** ou **Arquivo** → **Exportar JPG** na barra de menus.

#### 4. Considerações finais

O aplicativo *A-Graph* oferece uma interface de simples interação com o qual os alunos podem desenhar grafos, com ou sem direção, definindo livremente sua geometria e conseqüentemente a topologia subjacente; permite a execução dos algoritmos de busca em grafos, de maneira didática, inclusive implementando a heurística da escolha livre do vértice inicial a partir do qual a busca se iniciará.

Estão previstas diversas extensões para o aplicativo *A-Graph* de maneira a torna-lo mais funcional. Entre as quais mencionamos:

- O aprimoramento da Interface Gráfica voltado ao Usuário (GUI) uma vez que a interface na versão atual é muito simples;
- A implementação dos algoritmos de Kruskal e Prim para a construção de Árvores Geradoras de Peso Mínimo (*Minimum Spaning Tree*);
- A inclusão dos algoritmos de Dijkstra e Bellman-Ford para o problema de Caminhos de Custo Mínimo (*Shortest Paths Minimum*);
- Realizar a validação formal dos requisitos funcionais de software.

Com as extensões acima listadas acreditamos que a ferramenta *A-Graph* possa ser utilizada além dos contextos de Computação e Engenharia, extrapolando o propósito inicial de ser utilizado como ferramenta computacional no ensino-aprendizado da Teoria dos Grafos e de seus Algoritmos.

#### Referências

Cormen, Thomas H. et. al. (2002), Algoritmos: teoria e prática. Edt. Elsevier, 2ª Edição, 5ª reimpressão, Rio de Janeiro.

Dantas do N., H. A. (1997), Uma Abordagem para o Desenho de Grafos Baseada na Utilização de Times Assíncronos, Dissertação de Mestrado, IC-UNICAMP, Campinas, SP.

Prado do, A. T. (2009), Proposta de Ferramenta Gráfica para representação de Grafos com Aplicação de Métodos de Busca. Monografia de Conclusão de Curso de Engenharia da Computação, Universidade São Francisco, Campinas, SP, 2009.

Silva da, Laércio Rodolfo (2009). Um estudo das imagens vetoriais no formato SVG. Trabalho de Conclusão de Curso em Ciência da Computação, Faculdade Campo Limpo Paulista, SP.

Szwarcfiter, J. (1986). Grafos e algoritmos computacionais. 2. ed. Rio de Janeiro: Editora Campus,