

## CAMINHO MÍNIMO DE REDES CONECTADAS UTILIZANDO GRAFOS

### *Shortest Path of Connected Networks Using Graphs*

João Vitor Azevedo Jacundá Santos<sup>1</sup>, George Lauro Ribeiro Brito<sup>2</sup>, Gentil Veloso Barbosa<sup>3</sup>

#### RESUMO

Neste trabalho propõe-se desenvolver e implementar modelos matemáticos capazes de descrever o comportamento do caminho mínimo em redes conectadas, aplicando técnicas de otimização em problemas apresentados na literatura e buscando relacionar um problema real que possibilite utilizar os conhecimentos adquiridos. Com esses objetivos, foram implementados e analisados quatro algoritmos encontrados na literatura: de Dijkstra, Ford/More, Floyd, Dantzig. Para validação das técnicas os algoritmos foram aplicados à Rede Ipê da RNP para verificar a funcionalidade e definição do caminho ótimo entre os PoPs utilizando o atraso como ponto de decisão. O problema do caminho mínimo está intimamente ligado ao nosso cotidiano, e embora algum algoritmo tenha se mostrado mais eficiente, não devemos nos limitar a ele. Cada algoritmo encaixa-se melhor numa determinada situação.

**Palavras-chave:** Grafo. Caminho Mínimo. Algoritmos. RNP.

#### ABSTRACT

The purpose of this research is develop and implement mathematical models capable to determine the path or smaller route (distance, time or any cost of interconnected networks, applying optimization techniques in problems presented in the literature and seeking to relate a real problem that allows to use the knowledge acquired. The minimum route problem it's so common in our lives. For these purposes, four algorithms have been implemented and analyzed: from Dijkstra, Ford / More, Floyd and Dantzig. To validate the techniques, the algorithms were applied to the National Research Network (RNP) to verify the functionality and definition of the shortest path between the PoPs using the delay as a decision point. The minimum path problem is closely tied to our everyday lives, many applications dealing with transport networks and communications require the shortest path calculation, and although an algorithm has proved to be more efficient, we should not limit ourselves to it. Each algorithm fits best in a given situation.

**Keywords:** Graph. Shortest Path. Algorithm. RNP.

<sup>1</sup> Graduando em Ciência da Computação – Universidade Federal do Tocantins.

E-mail: jacunda@uft.edu.br

<sup>2</sup> Professor Associado da Universidade Federal do Tocantins no Curso de Ciência da Computação. Tem Doutorado em Engenharia Elétrica pela Universidade de Brasília - UnB e Pós-Doutorado em Tecnologias Educacionais pela Universidade de Lisboa – Ulisboa.

E-mail: gbrito@uft.edu.br

<sup>3</sup> Professor Adjunto da Universidade Federal do Tocantins no Curso de Ciência da Computação. Tem Doutorado em Engenharia de Sistemas e Computação pela Universidade Federal do Rio de Janeiro - UFRJ.

E-mail: gentil@uft.edu.br

## 1. INTRODUÇÃO

Em uma rede qualquer, dependendo das suas características, podem existir vários caminhos entre um par de nós, definidos como origem (s) e destino (t). Entre os vários caminhos aquele que possui o menor “peso” é chamado de caminho mínimo. Este peso representa a soma total dos valores dos arcos que compõem o caminho e estes valores conforme referenciados anteriormente podem ser: o tempo de viagem, a distância percorrida ou um custo qualquer do arco.

Assim, os algoritmos de Caminho Mínimo determinam a rota de menor tempo, distância ou custo entre um par ou pares de origem e destino.

O Caminho ótimo é aquele que apresenta uma sequência de arcos conectando o nó de origem e o nó de destino de tal forma que a soma dos valores dos arcos no caminho é minimizada.

Se o caminho é mínimo entre um par de nós então qualquer caminho que é parte deste caminho também é mínimo

Existem basicamente dois tipos de estrutura de algoritmos para cálculo de caminhos mínimos: árvores e matrizes.

Nos algoritmos em árvore, determinam-se os caminhos mínimos de um nó para todos os outros nós da rede ou simplesmente entre um par de nós. Neste tipo de algoritmo, a solução é obtida construindo-se passo a passo uma árvore de caminhos mínimos. Para se obter os caminhos mínimos de todos os nós para todos os outros nós da rede, utilizam-se os algoritmos com estrutura de matriz e neste caso os caminhos mínimos entre todos os pares de nós são obtidos simultaneamente.

Na literatura pesquisada encontramos os seguintes algoritmos mais comumente utilizados, com estrutura em árvore: Dijkstra e Ford/Moore, e os algoritmos em matriz: de Floyd e Dantzig. Para este projeto foram testados algoritmos com estrutura em árvore, que possibilitam a descoberta do caminho mínimo de um nó para outro, e em matriz que possibilitam verificar as interligações através de diversos caminhos.

## 2. MATERIAIS E MÉTODOS

### Rede Ipê

A rede Ipê é uma infraestrutura de rede Internet, operada pela RNP, dedicada à comunidade brasileira de ensino superior e pesquisa, que interconecta universidades, institutos de pesquisa e instituições culturais.

A infraestrutura da rede Ipê engloba 27 Pontos de Presença (PoPs), um em cada unidade da federação, além de ramificações para atender 1219 campi e unidades de instituições de ensino, pesquisa e saúde em todo o país, beneficiando mais de 3,5 milhões de usuários.

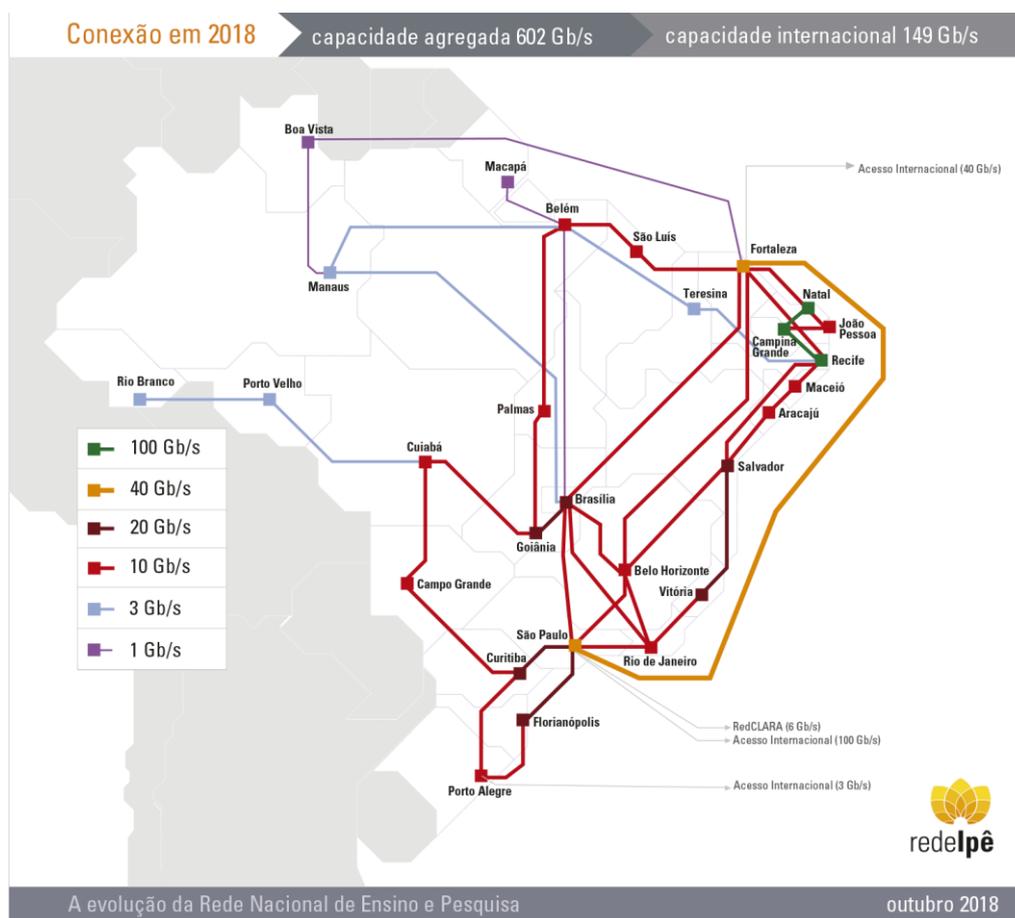


Figura 1. Rede Ipê. Fonte: www.rnp.br

### Grafos e Caminho mínimo

Um grafo é um conjunto de vértices e um conjunto de arestas que ligam pares de vértices distintos, estes podem ser usados como modelos de muitos tipos de relações e

processos. Assim, é utilizado para resolução de diversos problemas em uma grande variedade de áreas.

Na Computação, grafos são usados para representar redes de comunicação, organização de dados, dispositivos computacionais, fluxo de computação, redes elétricas, e etc.

Neste artigo serão abordados os principais algoritmos utilizados para modelagem de redes de comunicação utilizando grafos para obtenção do caminho mínimo.

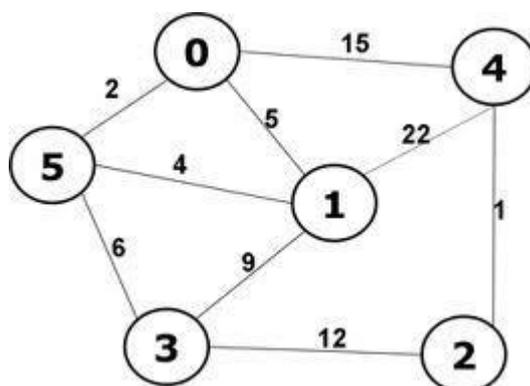


Figura 2. Exemplo de Grafo

O Caminho mínimo é encontrado a partir de dois vértices em um grafo, de modo que a soma dos pesos de suas arestas constituintes sejam minimizadas. Equação geral de atualização (relaxamento):

$$dist(v, i) = \min_{\{(u, v) \in E\}} \{dist(u, i - 1) + l(u, v)\}$$

## Algoritmos

O Algoritmo de **Dijkstra** é capaz de determinar o caminho mínimo de um grafo ponderado partindo de um vértice de início para todos os outros vértices com pesos positivos, mostrando um ótimo desempenho em tempo computacional  $O([m+n]\log n)$  onde  $m$  é o número de arestas e  $n$  é o número de vértices. O algoritmo inicia-se na sua origem e a cada passo é selecionado um conjunto de vértices que estão perto da origem, se ao passar pelo vértice acrescentado, a distância ficar menor, essa nova distância será memorizada.

O algoritmo de **Ford/Moore** difere do algoritmo de Dijkstra por suportar arestas com pesos negativos. O algoritmo de Bellman-Ford executa em tempo  $O(V \times A)$  onde  $V$  é o

número de vértices e  $A$  o número de arestas. É dividido em três etapas: inicialização, relaxamento (onde é feito o cálculo do caminho mínimo) e a verificação de ciclos negativos.

O algoritmo em Matriz de **Floyd-Warshall** encontra o menor caminho entre todos os pares de vértices, no entanto ele apenas encontra os valores de tais caminhos, e não a sequência de arestas a ser percorrida. A sua complexidade computacional é de  $O(N^3)$ .

O algoritmo de **Dantzig** é semelhante ao de Floyd, assim como este, ele encontra o menor caminho entre todos os pares de vértices, ignorando a sequência de arestas percorridas.

## Experimento

Para validar e analisar as técnicas, os algoritmos foram implementados na linguagem de programação Matlab, os resultados foram obtidos em um notebook core i7 com 8GB de RAM. Os algoritmos propostos foram testados no grafo da figura 3 que representa os atrasos entre os PoPs da Rede Ipê, cada PoP representa um nó, totalizando 27. Os programas lêem os dados, montam o grafo na memória, executa o algoritmo de otimização e mostra os resultados.

Para testes, foram feitas as simulações dos atrasos entre PoP-TO e PoP-RS, e entre os PoP-RO e PoP-PE. Todos os algoritmos testados conseguiram chegar aos mesmos resultados, no entanto, os algoritmos de matriz não nos possibilita a visão da rota, apenas a distância. Deste modo, visto que todos os algoritmos testados foram capazes de encontrar o melhor caminho, resta saber qual o algoritmo mais rápido para encontrar a solução ótima

## 3. RESULTADOS E DISCUSSÃO

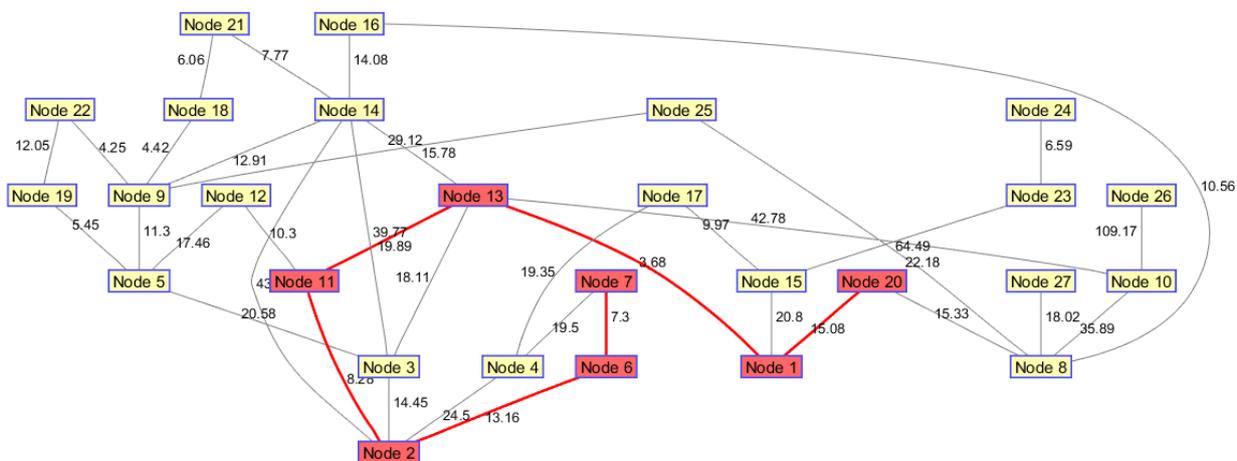
Para os resultados encontrados no presente estudo para obtenção e análise de caminho mínimo de redes conectadas, foram feitos os levantamentos, análises e seleções de materiais bibliográficos que foram utilizados como fonte de pesquisa para implementação dos métodos e modelos de caminho mínimo de redes conectadas. Duas estruturas e seus algoritmos de programação foram implementadas em ambiente Matlab, de estrutura em árvore: Dijkstra e Ford/Moore, e os algoritmos em matriz de Floyd e Dantzig que foram aplicados na Rede Ipê para resolução do caminho mínimo através do atraso.

Para resolver o problema, algumas informações foram levantadas da literatura. Sabe-se que os algoritmos de estrutura em árvore possibilitam a visão de todas as arestas

passadas e o seu respectivo valor. Os algoritmos em matriz, apesar de serem mais rápidos, não há a possibilidade de saber qual foi o caminho, mostrando apenas as distâncias, portanto, os algoritmos em árvore se encaixam na maioria dos problemas. O algoritmo de Ford/Moore tem conveniência de suportar arestas com pesos negativos, executando em tempo  $O(V \times A)$ , diferentemente do Algoritmo de Dijkstra com tempo computacional  $O([A+V]\log V)$ . Nos testes feitos o algoritmo de Bellman-Ford foi o que se saiu melhor, com uma ínfima vantagem.

Vários fatores precisam ser considerados para definir uma boa rota em redes conectadas. Atraso, vazão, disponibilidade e peso estão entre os principais elementos modificantes avaliação deste tipo de projeto. A modelagem em grafos da Rede Ipê foi apresentada considerando o tempo de atraso de um vértice ao outro. Foram implementados quatro algoritmos para o caminho mínimo encontrados na literatura e testados.

Aplicando nos algoritmos obtiveram-se o seguinte resultado numa conexão entre o PoP-TO e PoP-RS:



**Figura 3.** Simulação caminho mínimo entre PoP-TO e PoP-RS, usando Dijkstra ou Ford/Moore.

No primeiro teste o algoritmo de Dijkstra realizou o teste tempo computacional igual a 0.01625s, o de Bellman Ford rodou em 0.0155s, o de Dantzig em 0.011s e o de Floyd em

0.009s. O teste de validação foi feito considerando os 27 nós da rede Ipê.

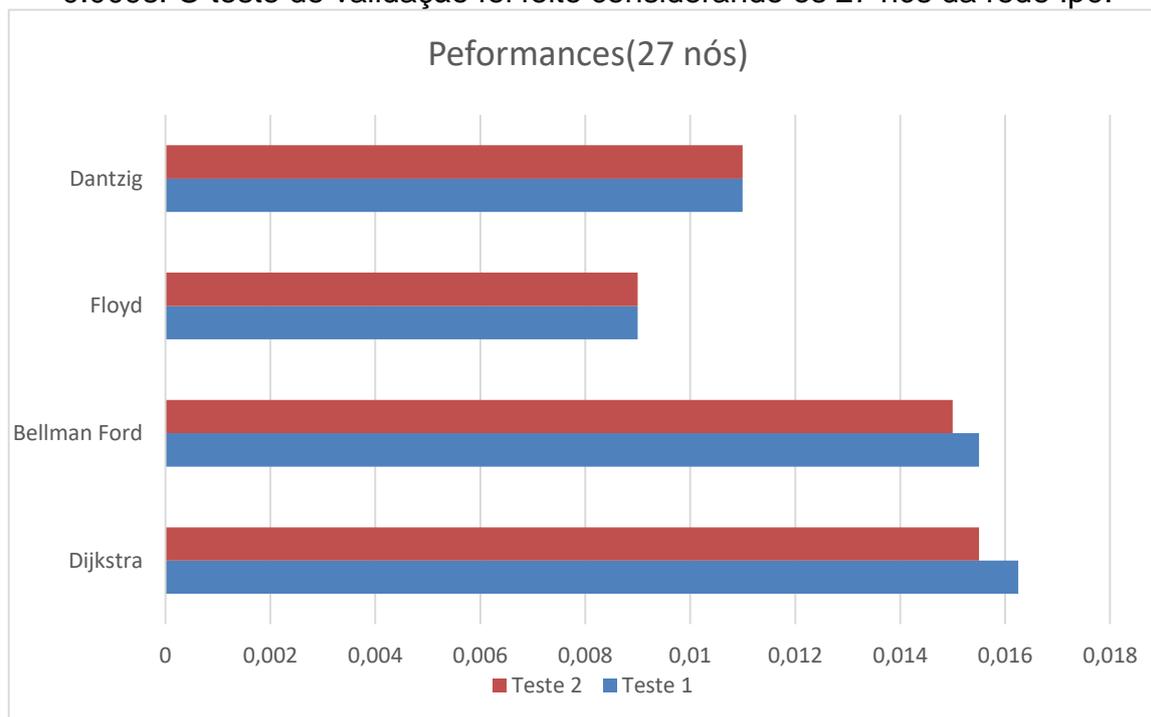


Figura 4. Comparação em tempo de execução de todos os algoritmos

#### 4. CONSIDERAÇÕES FINAIS

Durante esse tempo de desenvolvimento do projeto, foram estudadas as teorias e algoritmos de Grafos no âmbito de minimização do caminho e seus algoritmos, para obtenção da menor rota possível. Através desses estudos foram vistos diversos algoritmos de estrutura em árvore e matriz. Embora os algoritmos em matriz tenham se mostrado mais rápidos, não dá a possibilidade de saber qual foi o caminho, mostrando apenas as distâncias de um ponto para todos, portanto, neste caso, foram escolhidos os algoritmos sem árvore para devidos testes pela possibilidade de sabermos qual foi a rota mais confiável. Ambos os algoritmos, de Dijkstra e Bellman-Ford, tiveram resultados semelhantes nos testes feitos. O problema do caminho está intimamente ligado ao nosso cotidiano, cada algoritmo encaixa-se melhor numa determinada situação. Em casos de grafos com pesos negativos, recomenda-se utilizar Ford/Moore; outros onde a rota é desconsiderada, é mais interessante usar os algoritmos em matriz.

## REFERÊNCIAS

- BAZARAA M. S., Jarvis J. J., Sherali H. D. (1990). **Linear Programming and Network Flows**, 2nd. Ed., John Wiley & Sons.
- BOAVENTURA NETTO, P. O. (1996). **Grafos: Teoria, Modelos, Algoritmos**. Editora Edgar Blucher Ltda, São Paulo.
- DAUDT C., MIRANDA C. **Análise da Complexidade do Algoritmo de Floyd-Warshall**. Universidade Federal do Rio Grande do Sul, 2010.
- EVANS, James & MINIEKA, Edward (1978). **Optimization Algorithms for Networks and Graphs**.
- FEOFILOFF P., KOHAYAKAWA Y., WAKABAYASHI Y. **Uma Introdução Sucinta à Teoria dos Grafos**. São Paulo, 2011.
- GOLDBARG, M. C. e LUNA, H. P. L. (2000). **Otimização Combinatória e Programação Linear: Modelos e Algoritmos**. Editora Campus, Rio de Janeiro.
- HILLIER, Fraderick S. & LIEBERMAN, Gerald (1995). **Introduction to Operations Research, Mc Graw-Hill**.
- LOBO, Fernando. **Caminho mais curto a partir de um nó**, Disponível em: <<http://www.fernandolobo.info/aed-ii/teoricas/a24e25.print.pdf>> Acesso em 03 de março de 2017.
- LACHTERMACHER, G (2002). **Pesquisa Operacional na Tomada de Decisões**. Editora Campus, Rio de Janeiro.
- RNP, **Rede Ipê**, Disponível em: <<https://www.rnp.br/servicos/conectividade/rede-ipe>>. Acesso em 04 de março de 2017.
- TONINI A., SCETTINO D. **MATLAB para Engenharia**. 1ª Edição. Belo Horizonte, 2002.
- WINSTON, W. L. (1993) **Operations Research: Applications and algorithms, Duxbury Press, Belmont**.
- WIKIPEDIA, **Algoritmo de Dijkstra**, Disponível em: <[https://pt.wikipedia.org/wiki/Algoritmo\\_de\\_Dijkstra](https://pt.wikipedia.org/wiki/Algoritmo_de_Dijkstra)> Acesso em 03 de março de 2017.
- WIKIPEDIA, **Algoritmo de Floyd-Warshall**, Disponível em: <[https://pt.wikipedia.org/wiki/Algoritmo\\_de\\_Floyd-Warshall](https://pt.wikipedia.org/wiki/Algoritmo_de_Floyd-Warshall)> Acesso em 08 de março de 2017.