

Algoritmo de dijkstra: Análise e Aplicação

Alessandro dos Santos Marques¹, Carlos Alexandre Ernst², Vitor Souza de Jesus³,
Mateus Eduardo Kina⁴

¹Rua Paulo Malschitzki, nº 10 Campus Universitário - Distrito Industrial
cep:89219-710 – Joinville – SC– Brasil

²Curso de bacharelado em engenharia de software

³Departamento de Sistemas e Computação
Universidade da Região de Joinville (UNIVILLE) – Joinville, SC – Brasil

{alessandromarques, carlosernst, vitor.jesus, mateuskina}@univille.br

1. Introdução

Um algoritmo prevê uma sequência ordenada de passos para o desenvolvimento de um determinado processo ou produto, existem alguns algoritmos que são usados basicamente em dodo tipo de "código-fonte" criado por ser funções intrínsecas de procura ou comparação. Este não é o caso do algoritmo de dijkstra, que tem peculiaridade em suas funções tão particulares que sua utilização é limitada.

Existe uma dificuldade principalmente ao programador iniciante em entender como proceder para implementar o algoritmo, assim será explanado seu funcionamento através de imagens de exemplos bibliográficos e comentários a cada iteração visualizada, também será examinado algumas propostas de para implementação real do algoritmo para que sua aplicação possa ser visualizada e massificada.

Para concluir traçaremos alguns elementos a serem considerados para que o sucesso na utilização do algoritmo de dijkstra seja atingido, também entenderemos como sua padronização em implementação se faz necessária para economizar tempo.

2. Funcionamento do algoritmo de "Dijkstra"

O algoritmo de dijkstra destaca-se por ser aplicável em determinadas situações em que existe um conjunto de entidades já nomeada a serem comparadas entre si com atributos que permitem quantificar sua posição hierárquica em meio a um conjunto de entidades em uma árvore. I

O algoritmo define um rótulo para cada folha em um grafo "árvore" e um dado quantitativo para a "distância" entre cada uma dessas folhas, essa "distância" não precisa ser necessariamente relativo a espaço físico, como exemplo podemos dar o valor total de vendas em um mês por loja em uma rede de lojas, e usar os menores dados para definir um caminho de visitas para auditoria ou controle de rendimentos, imaginando que os menores valores são correspondentes as lojas que necessitam de correções em seu processo de vendas.

Importante salientar que, apesar de ser possível criar inúmeros atributos para comparar para definir o "peso" de cada folha do grafo que representa os dados usados para encontrar o "caminho mais curto", apenas um valor, que pode ser a consequência de uma análise por outro algoritmo, sera usado a definir quais direções serão tomadas. Assim cada

folha pode ter vários atributos que terão de ser avaliados segundo um outro algoritmo de comparação que define um valor quantitativo para indexação desse rótulo.

Imaginando um grafo e sua tabela de informações representado pela (*Figura. 1*) mostra um exemplo de montagem das informações em um gráfico e em uma tabela para melhor visualização. [de Carvalho 2008]

O algoritmo inicia a procura analisando os "atributos", que nesse caso corresponde ao comprimento entre "folhas" do grafo, estes são comparados e armazenados em conjuntos de dados semelhantes a tabela na (*Figura. 2*), podemos perceber que a partir do rótulo "A" é possível seguir por 4 caminhos diferentes que levam conseqüentemente para 4 rótulos diferentes por 4 caminhos com dimensões diferentes. Na tabela preenchemos os valores referentes a distâncias entre os rótulos considerando o precedente "A" como origem e o status de "fechado" ao rótulo de origem.

A seguir temos a primeira escolha referente ao caminho que liga o rótulo "A" e o rótulo "E" por tem menor valor quantitativo, essa escolha gera uma nova concorrência entre os caminhos entre E-D, A-C e A-B, e usamos uma nova tabela para demonstrar os dados armazenados para novas comparações. (*Figura. 3*)

Repetindo a comparação prosseguimos para o rótulo "D" deixando com status "fechado" os rótulos "A", "D" e "E". (*Figura. 4*)

Observando o gráfico (*Figura. 5*), vemos que as distâncias entre D-C e D-E-A-C tem o mesmo peso quantitativo, de forma que quaisquer um dos dois caminhos pode ser usados sem prejuízo. Seguindo as imagens da (*Figura. 5*) e (*Figura. 6*) vemos a finalização de "traçagem" de trajeto e finalização do preenchimento da tabela, observando que apesar de ser uma quantidade de informações pequenas o funcionamento do algoritmo pôde ser verificado.

3. Conclusão

Existe três requisitos nos quais podemos nos basear para analisar a aplicabilidade do algoritmo, complexidade de implementação, modelagem matemática das situações a serem comparadas e o tamanho dos dados a serem analisados.

1. Aplicabilidade do algoritmo

No exemplo retirado de [de Carvalho 2008] vemos que o conjunto de dados que foi utilizado era extremamente pequeno, assim a visualização do caminho seria bem mais simples mesmo sem a utilização do algoritmo. Dessa forma entendemos que em muitos casos veremos situações que descrevem muitas variáveis, entretanto nem todas essas variáveis tem efeitos de mudança a situação observada. Um exemplo é a queda de uma bola de aço de 130kg de uma altura de 5 metros, apesar de existir a força de arrasto do vento lateralmente, essa variável não influi no efeito final, assim cabe desconsiderarmos essa variável na posição final de chegada da bola de aço.

2. Complexidade de Implementação.

Quando você tem a necessidade de trabalhar com um número elevado de trabalhadores é possível que a necessidade de padronização de processo para implementação desse algoritmo seja ditada como regra, pois a manutenção ou escalabilidade pode ser prejudicada se a mesma for ignorada. Entretanto a com-

plexidade de implementação depende única e exclusivamente do que é necessário fazer.

3. Modelagem matemática

A dificuldade de modelar matematicamente as "n" situações que podem demonstrar as relações que mudam os fatos observados com demonstrado em [Méndez 2008] que leva em consideração o vento e as ondas do mar como paralelo que pode alterar o tempo de deslocamento de uma embarcação.

4. Imagens

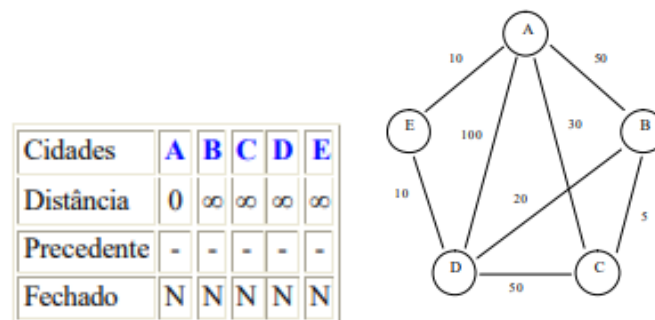


Figura 1. Montagem de grafo para visualização de rótulos e comprimento de caminhos e tabela para acompanhamento de progressão entre "folhas". [de Carvalho 2008]

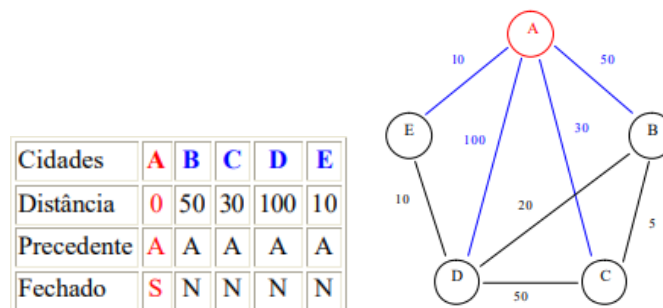


Figura 2. Análise dos caminhos segundo o rótulo de origem "A". [de Carvalho 2008]

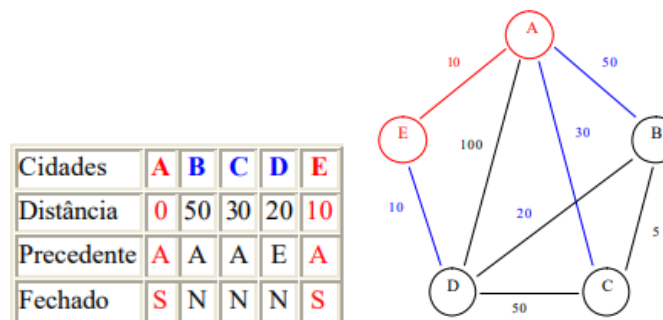


Figura 3. Primeiro caminho escolhido. [de Carvalho 2008]

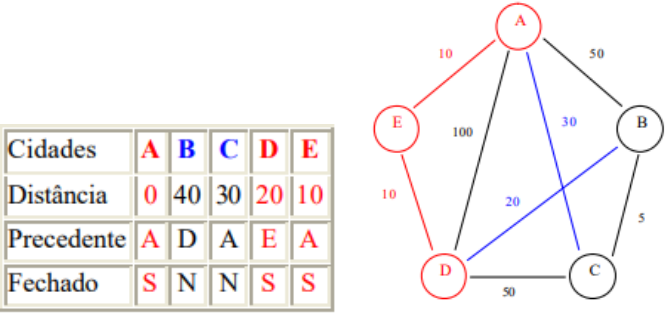


Figura 4. Continuidade da análise de distância entre rótulos. [de Carvalho 2008]

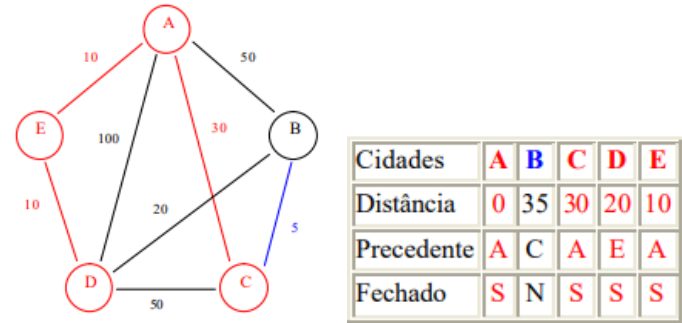


Figura 5. Caminho de "A"até "C"[de Carvalho 2008]

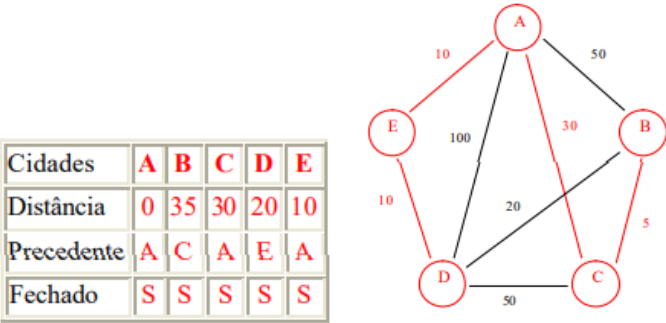


Figura 6. Finalização do caminho em "B"como sendo o trajeto de menor valor quantitativo. [de Carvalho 2008]

5. Referências

Anexo de imagens referente as figuras 1, 2 e 3.

Para análise de aplicação foi utilizado o [André Luís Silva 2010] onde apresentava a aplicação do algoritmo de dijkstra para implementação de um roteiro entre pontos turísticos de uma cidade, já em [Edson A. R. Barros 2007] vemos uma complexidade de implementação genérica ondem acrescentando dados através de objetos filtradores podemos criar uma classe que categoriza qualquer dado quantitativo que possa ser implementado, observando métodos utilizados para simulações computacionais.

Também utilizamos o artigo [Daiana Fernandes da Silva 2007] que exemplifica uma análise combinatoria para o método do "caixeiro viajante" mas que reconhecidamente falha em algumas situações em que o autor implementa com o algoritmo de dijkstra para melhor atender a sua aplicabilidade.

Referências

- André Luís Silva, Lucas Machado de Oliveira, R. Q. d. A. (2010). Metodologia baseada no algoritmo de dijkstra para roteirização de pontos turísticos em ouro preto. *Universidade federal de viçosa*, 1:1–10. Disponível em: <http://www.saepro.ufv.br/wp-content/uploads/2010.2.pdf> - Acesso em: 22 nov. 2021.
- Daiana Fernandes da Silva, A. L. S. (2007). Aplicação conjunta do método de dijkstra e otimização combinatória para solução do problema do caixeiro viajante. *Universidade Mackenzie*, 1:1–9. Disponível em: https://www.aedb.br/seget/arquivos/artigos09/224_224_224_Artigo_Seget.pdf - Acesso em: 25 nov. 2021.
- de Carvalho, B. M. P. S. (2008). Algoritmo de dijkstra. *Universidade de Coimbra*, 1:1–6. Disponível em: <https://student.dei.uc.pt/~brunomig/cp/Artigo.pdf>. - Acesso em: 22 nov. 2021.
- Edson A. R. Barros, Sergio V. D. Pamboukian, L. C. Z. (2007). Algoritmo de dijkstra: Apoio didático e multidisciplinar na implementação, simulação e utilização computacional. *Universidade Mackenzie*, 1:1–4. Disponível em: http://meusite.mackenzie.br/edsonbarros/publicacoes/ICECE2007_212.pdf - Acesso em: 24 nov. 2021.
- Méndez, Y. S. (2008). Problema do caminho mais curto - algoritmo de dijkstra. *Universidade Federal Fluminense*, 1(2175-6295):1–12. Disponível em: https://www.marinha.mil.br/spolm/sites/www.marinha.mil.br/spolm/files/006_1.pdf. - Acesso em: 22 nov. 2021.