

Algoritmos, Experimentação e Teoria em Otimização Combinatória

Aluno: Marcelo Hashimoto (3464942) - mhashimo@linux.ime.usp.br

Supervisor: Prof. Dr. José Coelho de Pina Jr. - coelho@ime.usp.br

INICIAÇÃO CIENTÍFICA COM APOIO FINANCEIRO DA FAPESP
PROCESSO 04/00581-4 INICIADO EM 9 DE ABRIL DE 2004

1 Introdução

Otimização combinatória é um campo da matemática aplicada que se baseia no uso conjunto de técnicas de combinatória, programação matemática e teoria de desenvolvimento de algoritmos para resolver problemas de otimização formulados sobre estruturas discretas.

Problemas de otimização combinatória têm sido um tópico central para a evolução de algoritmos e da teoria de complexidade computacional. Pesquisadores têm apresentado muitas idéias criativas para o projeto de algoritmos eficientes, baseados em conceitos e resultados na área. A sub-área conhecida como *combinatória poliédrica*, por exemplo, apresenta métodos desenvolvidos a partir de conceitos de programação linear, como o primal-dual, e que têm se mostrado muito úteis no projeto e análise de uma variedade de algoritmos para problemas em outros domínios. Muitas das idéias inovadoras têm se baseado em um conjunto não muito grande de princípios comuns, como *scaling*, que são simultaneamente simples e poderosos.

Nem sempre, porém, é suficiente restringir a pesquisa de um problema ao estudo teórico de seus aspectos e ao desenvolvimento abstrato de algoritmos que o resolvem. Em diversos casos, existem fatores fundamentais que só podem ser resolvidos, ou mesmo descobertos, durante a implementação propriamente dita. Não é incomum que a obtenção de uma eficiência próxima àquela calculada na teoria, ou mesmo o próprio funcionamento de um algoritmo, dependa do uso de estruturas de dados sofisticadas e técnicas de programação que não são triviais.

Ademais, a complexidade de um algoritmo na teoria é muitas vezes um argumento questionável para classificá-lo na prática: não são incomuns algoritmos teoricamente eficientes mas lentos na prática, e algoritmos cuja análise de pior caso está longe de representar seu comportamento no caso médio. Diante dessa possibilidade, testes e experimentações são ferramentas poderosas e úteis, e que enfatizam ainda mais a importância da implementação.

2 Objetivos

O objetivo deste trabalho é me familiarizar com algumas das principais ferramentas de otimização combinatória através do estudo, implementação e teste de algoritmos para problemas clássicos da área envolvendo *fluxos em redes*. Este é um projeto de iniciação científica, desenvolvido em conjunto com a colega Juliana Barby Simão e sob a orientação de José Coelho de Pina. Pretendemos aproveitá-lo como preparação para um possível mestrado na área.

A fonte dos algoritmos que serão incluídos no estudo, e também a principal base teórica, será o livro de Ahuja, Magnanti e Orlin [1]. Também pretendemos utilizar como apoio secundário o livro de Cook, Cunningham, Pulleyblank e Schrijver [2]. Em nenhum momento, porém, vamos nos restringir a seguir somente tais referências. Consultas a outras fontes, além de adições e modificações às originais, são esperadas e, dependendo do contexto, desejadas.

O corolário do estudo será um texto no qual descreveremos toda a teoria envolvida no problema e em cada um dos algoritmos vistos. Pretendemos que a linguagem algorítmica utilizada siga os padrões das notas de Feofiloff [6], e permita descrever os algoritmos com base em seus invariantes, para destacar de maneira simples a razão pela qual eles funcionam.

Como queremos disponibilizar as implementações de modo que elas possam ser facilmente compreendidas e aplicadas, elas serão feitas na linguagem CWEB [11] de *literate programming* [9]. As principais estruturas de dados utilizadas serão as fornecidas pela plataforma SGB, desenvolvida por Knuth [10]. Após finalizar as implementações, pretendemos realizar um estudo comparativo entre todos os algoritmos e incluir os resultados na documentação final.

Esta iniciação é na verdade parte de um projeto maior, no qual estão envolvidos, além de mim e a Juliana, os alunos Roger Ricardo Flores de Araújo e Nelson Guedes Paulo Júnior. Roger está atualmente desenvolvendo uma iniciação científica similar à nossa, mas sobre algoritmos para emparelhamentos, e Nelson está desenvolvendo um programa escrito na linguagem C++, que permitirá a visualização gráfica do funcionamento de todos os algoritmos implementados.

Pretendemos reunir todos os textos, implementações e resultados do projeto em um sítio da Internet, preferencialmente nos moldes da *Network Optimization Library* de Goldberg [8].

3 Atividades já Realizadas

Inicialmente, estudamos alguns tópicos não-específicos de otimização combinatória através do livro de Cook et al. [2] e Cormen et al. [3] para nos familiarizar um pouco mais com a teoria envolvida no problema. A partir desse ponto, a base teórica de nossos estudos foi quase puramente o livro de Ahuja et al. [1], como estava previsto no planejamento inicial.

Antes de nos dedicarmos a fundo ao estudo de [1], trabalhamos em cima de uma implementação em C para o algoritmo de Ford e Fulkerson [7] que havia sido feita para a disci-

plina MAC0328 - ALGORITMOS EM GRAFOS. O algoritmo em questão é conhecido como *método dos caminhos de aumento*, e a implementação utilizava a plataforma SGB para aplicar o método de Edmonds e Karp [5], ou *método do aumento através de caminhos mínimos*.

A Juliana se baseou na implementação original para desenvolver uma versão alternativa do algoritmo, conhecida como *método do aumento através de fluxos bloqueadores*, que foi concebida por Dinic [4]. Eu trabalhei em cima do algoritmo original de Ford e Fulkerson, modificando-o para que utilizasse, ao invés do método de Edmonds e Karp, os métodos de *caminhos de maior aumento* e *capacity scaling*. Durante o desenvolvimento do primeiro, houve a necessidade da implementação de uma *fila de prioridade*. A primeira versão funcional utilizava uma fila trivial e muito ineficiente. A segunda utilizou uma estrutura de dados conhecida como *leftist heap*, que acabou se mostrando inadequada para o problema. A terceira e atual versão utiliza um *heap* tradicional, cuja implementação foi baseada em [3].

Após alguns testes com as novas implementações, ainda na linguagem C, começamos a estudar mais a fundo os conceitos abordados em [1] e a discutir algumas dúvidas teóricas entre nós e com o orientador. As principais discussões foram em torno da análise da complexidade dos algoritmos e da modelagem do problema sob o contexto de *programação linear e dualidade*. Por enquanto, estudamos a teoria envolvida nos conceitos básicos do problema do fluxo máximo, particularmente o *Teorema do Fluxo Máximo e Corte Mínimo*, e na análise dos já mencionados algoritmos de caminhos de aumento. Atualmente, estamos estudando outra classe de algoritmos para fluxo máximo, que utiliza o método conhecido como *pré-fluxo*.

4 Cronograma para o Segundo Semestre

- **Julho e início de Agosto:**

Redigir o texto referente a todo o material estudado até agora.

Passar as implementações já feitas em C para CWEB.

Preparar o primeiro relatório para a FAPESP.

- **Final de Agosto:**

Concluir o estudo dos algoritmos de pré-fluxo.

- **Setembro:**

Implementar os algoritmos de pré-fluxo.

Estudar conceitos básicos relacionados ao problema do fluxo máximo de custo mínimo.

- **Outubro:**

Estudar e implementar os algoritmos para fluxo máximo de custo mínimo.

- **Novembro:**

Estudar o algoritmo simplex para redes.

Preparar o pôster e a apresentação do trabalho.

Revisar e concluir o texto final da monografia.

O texto envolvendo toda a teoria estudada, o qual servirá como base para a monografia, deverá ser continuamente incrementado e revisado no decorrer do semestre, na medida em que cada tópico for sendo estudado. Além disso, é nossa intenção que as próximas implementações sejam realizadas diretamente em CWEB para já serem armazenadas no sítio da biblioteca.

Cabe observar que, no próximo semestre, tanto a Juliana como eu estaremos cursando a disciplina MAC05781 - OTIMIZAÇÃO COMBINATÓRIA. Uma vez que muitos dos assuntos tratados durante a disciplina serão comuns aos que estaremos estudando em nossa iniciação científica, é provável que algum trabalho seja adiantado, de forma a tornar possível nossa dedicação a tópicos complementares que podem eventualmente ser incluídos no projeto.

5 Estrutura da Monografia

- Introdução
- Parte Técnica
 - Objetivos e Metodologia
 - Tópicos Estudados
 - * Problema do Fluxo Máximo
 - Conceitos Básicos
 - Algoritmos Polinomiais
 - * Problema do Fluxo Máximo de Custo Mínimo
 - Conceitos Básicos
 - Algoritmos Polinomiais
 - * Simplex para redes
 - Resultados Obtidos
 - Referências
- Parte Subjetiva
 - Desafios e Frustrações
 - Relacionamento com o Curso
 - Interação com o Supervisor
 - Cooperação com a Colega

- Conclusões e Observações
- Planejamento Futuro

Referências

- [1] R.K. Ahuja, T.L. Magnanti, and J. Orlin, *Network flows: Theory, algorithms, and applications*, Practice Hall, 1993.
- [2] W.J. Cook, W.H. Cunningham, W.R. Pulleyblank, and A. Schrijver, *Combinatorial optimization*, Wiley-Interscience series in discrete mathematics and optimization, John Wiley & Son's, New York, 1998.
- [3] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, *Introduction to algorithms*, 2nd. ed., The MIT Press and McGraw-Hill, 2001.
- [4] E.A. Dinic, *Algorithm for solution of a problem of maximum flow in a network with power estimation*, Sov. Math. Dokl. **11** (1970), no. 5, 1277–1280.
- [5] J. Edmonds and R.M. Karp, *Theoretical improvements in algorithmic efficiency for network flow problems*, J. ACM **19** (1972), no. 2, 248–264.
- [6] P. Feofloff, *Notas de aula de MAC 5781 otimização combinatória*, <http://www.ime.usp.br/~pf/>, 1997.
- [7] L.R. Ford and D.R. Fulkerson, *Maximal flow through a network*, CJM **8** (1956), 399–404.
- [8] A.V. Goldberg, *Network optimization library*, <http://www.avglab.com/andrew/soft.html>.
- [9] D.E. Knuth, *Literate programming*, Center for the study of Language and Information (CSLI), 1992.
- [10] ———, *The stanford graphbase: A plataform for combinatorial computing*, ACM Press, 1993.
- [11] D.E. Knuth and S. Levy, *The CWEB system of structured documentation*, Addison-Wesley, 1994.