Decomposição Lagrangeana

João C. Abreu[1]

[1] *Universidade Federal de Minas Gerais, DCC,*
Avenida Antônio Carlos 6627, Belo Horizonte, Brazil,
joao.junior@dcc.ufmg.br

Abstract Esse relatório apresenta a decomposição lagrangeana e o algoritmo do volume para resolver um problema proposto.

Keywords: Decomposição Lagrangeana, Método do Volume

1. Introdução

O Problema composto pela função objetivo (1) e as restrições (2)-(6), onde as restrições (2)-(4) formam uma árvore, e a restrição (5) representa a quantidade máxima de arestas escolhidas incidentes a um nó $i \in V$,

$$min \sum_{e \in E} c_e x_e \tag{1}$$

Sujeito à:

$$\sum_{e \in E} = n - 1 \tag{2}$$

$$t_i - t_j + |V| x_{ij} \leq |V| - 1, \forall (i,j) \in E \tag{3}$$

$$0 \leq t_i \leq |V|, \forall i \in V \tag{4}$$

$$\sum_{e \in \delta(i)} x_e \leq b_i, \forall i \in V \tag{5}$$

$$x_e \in \{0, 1\} \tag{6}$$

```
inicio
    Passo0: Dado um multiplicador π inicial, resolva o problema lagrangeano
    a = θ(π) e obtenha z e v. Faça z_0 = z, v_0 = v e t = 1.
    Passo1: Calcule v_t = z − v e π_t = π + s v_t. Resolva o problema lagrangeano
    a_t = θ(π_t), sendo v^t e z^t as soluções obtidas
    Passo2: if a_t > a then
        π = π_t
        a = a_t
    end
    t = t + 1 e vá para o passo 1
    Sendo:
    s = f (UB − a)/||v||^2
fin
```

Figure 1: Algoritmo do método do volume para o problema lagrangeano $\theta(\pi)$

Pode ser decomposto como:

$$min \frac{1}{2} \sum_{e \in E} c_e z_e + \frac{1}{2} \sum_{e \in E} c_e v_e \tag{7}$$

Sujeito à:

$$z \in T \tag{8}$$

$$v \in M \tag{9}$$

$$z = v \tag{10}$$

$$(z, v) \in \{0, 1\}^{2m} \tag{11}$$

Associando multiplicadores de lagrange π a restrição (10), teremos o seguinte problema de lagrange a ser resolvido:

$$\theta(\pi) = min \frac{1}{2} \sum_{e \in E} c_e z_e + \frac{1}{2} \sum_{e \in E} c_e v_e + \sum_{e \in E} \pi (z_e - v_e) \tag{12}$$

Sujeito à:

$$z \in T \tag{13}$$

$$v \in M \tag{14}$$

$$(z, v) \in \{0, 1\}^{2m} \tag{15}$$

2. Algoritmo

A figura 1 apresenta o algoritmo do volume revisado adaptado para o problema lagrangeano $\theta(\pi)$ retirado de [1].

3. Experimentos Computacionais

Os experimentos computacionais foram executados em uma máquina Intel Dual-Core de 2.81 GHz de clock e 2GB de memória RAM, rodando o sistema operacional Linux. O modelo matemático composto pela função objetivo (7) e as restrições (8)-(11) foi implementado no Ilog CPLEX 12.5.1 e o algoritmo da seção 2 foi implementado em Python 2.7, sendo que o otimizador utilizado para resolver o problema lagrangeano $\theta(\pi)$ presente nesse algoritmo foi o Ilog CPLEX 12.5.1. As instâncias de testes utiizadas são denominadas ANDINST e foram propostas em [2].

No experimento desse trabalho foi comparado a performance do modelo matemático composto pela função objetivo (7) e as restrições (8)-(11) que será chamado aqui de IP, com o algoritmo proposto na seção 2, que será chamado $Volume$. O modelo IP foi executado através do CPLEX com todos os parâmetros default. O modelo presente no algoritmo $Volume$ foi executado pelo CPLEX também com os valores default. O algoritmo $Volume$ foi executado com os parâmetros $f = 0.01$, com um número máximo de iterações igual a 500000. O modelo IP e o algoritmo $Volume$ foram executados com um tempo de execução máximo de 7200 segundos.

A tabela 1 apresenta os resultados obtidos para o conjuntos de instância de testes. Nessa tabela a coluna 1 mostra o nome da instância de teste, as colunas 2 e 3 são resultados referentes ao modelo IP e as colunas 4,5 e 6 são resultados referentes ao algoritmo $Volume$. A coluna 2 apresenta o custo da solução obtido pela modelo IP e a coluna 3 apresenta o tempo consumido para encontrar essa solução. A coluna 4 apresenta o custo da solução obtido pelo algoritmo $Volume$, a coluna 5 traz o número de iterações do algoritmo e a coluna 6 mostra o tempo consumido pelo algoritmo $Volume$.

Para todas as instâncias o CPLEX conseguiu encontrar soluções ótimas, conforme pode ser observado pela coluna 2 da tabela 1. O algoritmo $Volume$ não conseguiu encontrar uma solução ótima ou próxima da ótima para nenhuma dessas instâncias, conforme pode ser observado na coluna 4 da tabela 1.

Todo código fonte produzido por esse trabalho pode ser obtido no endereço eletrônico: $https://github.com/joaojunior/decomposicao\_lagrangeana$

Instância	IP		$Volume$		
	Custo Solução	Tempo(s)	Custo Solução	Iterações	Tempo(s)
tb1ct100_1	2722	4.01	2384.67	334	1340.68
tb1ct100_2	2808	3.92	2408.08	275	1122.68
tb1ct100_3	2958	4.00	2428.90	215	866.78
tb1ct200_1	4353	58.20	3871.71	335	7200.00
tb1ct200_2	4544	58.14	3851.35	219	7200.00
tb1ct200_3	4741	58.29	4050.24	335	7200.00
tb1ct300_1	5456	340.60	3572.75	43	7200.00
tb1ct300_2	5713	338.77	3631.13	42	7200.00
tb1ct300_3	5114	336.54	3222.22	42	7200.00
tb1ct400_1	6232	1302.72	3426.63	13	7200.00
tb1ct400_2	6458	1289.88	3450.62	13	7200.00
tb1ct400_3	6148	1285.12	3244.81	13	7200.00
tb2ct500_1	6725	3105.34	3450.36	5	7200.00
tb2ct500_2	6702	3111.46	3310.76	5	7200.00
tb2ct500_3	6928	3103.93	3367.94	5	7200.00

Table 1: Comparação entre os custos da solução e tempos obtidos entre o modelo IP e o algoritmo $Volume$

References

[1] Barahona, Francisco The volume algorithm: producing primal solutions with a subgradient method. Mathematical Programming, 87(3):385–399, 1991.

[2] R. Andrade, A. Lucena, and N. Maculan. Using Lagrangian dual information to generate degree constrained spanning trees. Discrete Applied Mathematics, 154(5):703–717, 2006.