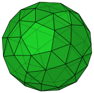
Tối ưu tổ hợp và quy hoạch tuyến tính nguyên

Quốc Bảo, Thu Huế, Ngọc Huy, Nam Kiệt



Mentors: Minh Quân, Hoàng Việt

July 28, 2024

Nội dung

Giới thiệu

Bài toán quy hoạch tuyến tính (LP):

- Đã có nhiều thuật giải tối ưu (thời gian đa thức).
- Kết quả có thể không hợp lý trong nhiều bài toán thực tế.
  - Xây thêm 5.4 nhà hàng hay mua thêm 4.5 cái bàn

  $\rightarrow$ Để giải quyết vấn đề thì chúng ta cần ràng buộc các biến phải là số nguyên.

Quy hoạch tuyến tính nguyên

Bài toán quy hoạch tuyến tính nguyên (ILP):

- Là trường hợp đặc biệt của bài toán LP,
- Với các biến được giới hạn chỉ nhận các giá trị nguyên.

So sánh giữa LP và ILP

- Có nhiều ứng dụng/ý nghĩa thực tế hơn LP.

So sánh giữa LP và ILP

- Có nhiều ứng dụng/ý nghĩa thực tế hơn LP.
- Trong trường hợp bài toán bị chặn, tập nghiệm của bài toán ILP là hữu hạn trong khi LP là vô hạn $\rightarrow$ dễ hơn?

So sánh giữa LP và ILP

- Có nhiều ứng dụng/ý nghĩa thực tế hơn LP.
- Trong trường hợp bài toán bị chặn, tập nghiệm của bài toán ILP là hữu hạn trong khi LP là vô hạn $\rightarrow$ dễ hơn?
- Độ khó của LP là P, còn ILP là NP.

ILP trong các bài toán tối ưu tổ hợp

Nhiều bài toán (tối ưu) tổ hợp có thể mô hình hóa về bài toán ILP.

- Bài toán cặp ghép cực đại (perfect matching problem),
- Bài toán người du lịch (travelling saleman problem),
- Bài toán lát cắt cực tiểu (minimum cut),
- Bài toán đường đi ngắn nhất,
- ...

Nội dung

Các phương pháp giải

Phương pháp mặt phẳng cắt (cutting-plane method):

- Thu hẹp miền nghiệm chấp nhận được bằng cách thêm các ràng buộc tuyến tính (lát cắt).
- VD: thuật toán Gomory (Gomory cut) (1958).

Các phương pháp giải

Phương pháp nhánh cận (branch and bound method):

- Với ý tưởng chia để trị, khám phá miền nghiệm một cách hệ thống để giải quyết các bài toán con nhỏ hơn (branch - phân nhánh) và sử dụng các điều kiện chặn (bound - cận) để tìm nghiệm tối ưu.
- VD: thuật toán nhánh cắt (branch and cut) $=$ phương pháp nhánh cận $+$ phương pháp mặt phẳng cắt $\rightarrow$ state of the art.

Các thuật toán phán đoán (heuristic algorithms):

- Được dùng để tìm lời giải với độ chính xác chấp nhận được và độ phức tạp về thời gian tính toán là hàm đa thức.
- Ví dụ: thuật toán leo đồi, thuật tối ưu đàn kiến,...

Nội dung

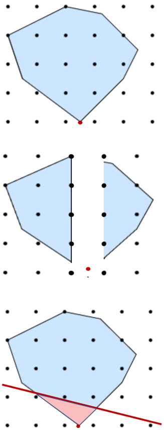
- Cách phân hoạch: $P := \{D_i \subseteq D \mid i \in I\}$
- Khái niệm không gian con

$$D = \bigcup_{i \in I} D_i \text{ và } D_i \cap D_j \neq \varnothing, \forall i \neq j$$

- Giải bài toán LP nới lỏng: $(ILP)_k \to (LP)_k$
- Kỹ thuật chia để trị:

$$D_i^{(ilp)} \to D_i^{(lp)} \to x^{(i)} \to D_{i+1}^{(ilp)} \text{ và } D_{i+2}^{(ilp)}$$

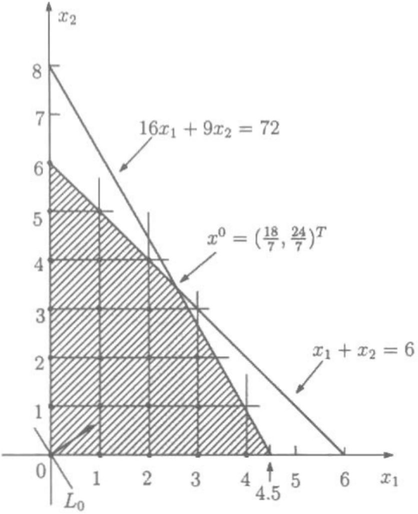
- Kỹ thuật cắt: $D := (D \setminus \{D_i\}) \cup \{D_{i+1}, D_{i+2}\}$



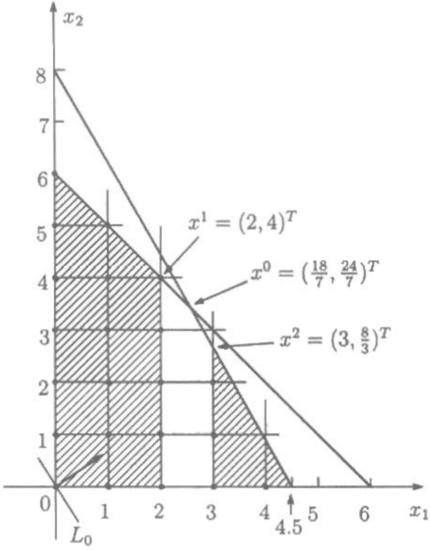
Algorithm 1 Phương pháp nhánh cận (tổng quát)

Khởi tạo $\hat{x}$
$\mathcal{L} \leftarrow \{\mathcal{D}\}$
while $\mathcal{L} \neq \emptyset$ **do**
    Chọn một tập $\mathcal{S}$ từ hàng đợi $\mathcal{L}$
    if một nghiệm $\hat{x}' \in \{x \in \mathcal{S} \mid f(x) > f(\hat{x})\}$ được tìm thấy **then** $\hat{x} \leftarrow \hat{x}'$
    if $\mathcal{S}$ không thể bị loại bỏ **then**
        Phân hoạch $\mathcal{S}$ thành $\mathcal{S}_1, , \ldots, \mathcal{S}_r$
        Thêm $\mathcal{S}_1, \ldots, \mathcal{S}_r$ vào $\mathcal{L}$
    $\mathcal{L} \leftarrow \mathcal{L} \setminus \{\mathcal{S}\}$
return $\hat{x}$

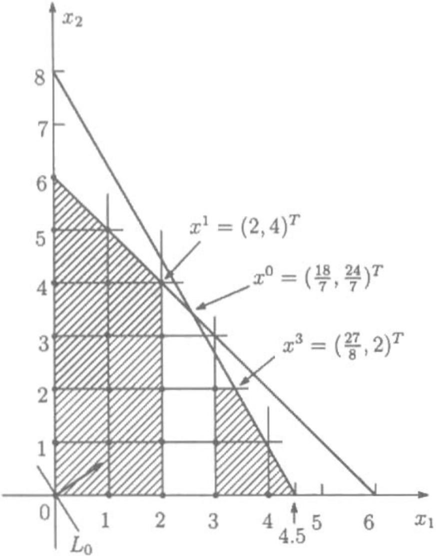
Ví dụ nhánh và cận



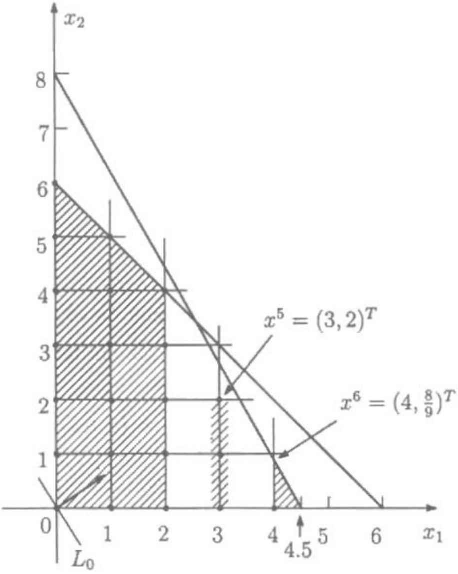
Ví dụ nhánh và cận



Ví dụ nhánh và cận



Ví dụ nhánh và cận



PiMA 2024

- **Chọn biến**: chọn biến không nguyên nào để **chia nhánh**?
- **Chọn lát cắt**: thêm ràng buộc tuyến tính nào để chia cắt D?
- **Chọn đỉnh để xét**: chọn miền con $D_i \subset D$ nào để tính toán?
- **Loại bỏ nhánh**: loại bỏ miền con D_i như thế nào?

Nội dung

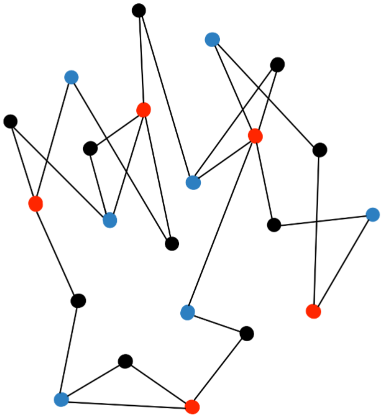
Định nghĩa số domatic

- Tập con D của $V(G)$ được gọi là tập thống trị nếu mỗi đỉnh của G hoặc thuộc D hoặc có một đỉnh kề thuộc D.
- Một phân hoạch domatic là một phân hoạch của tập đỉnh thành các tập thống trị đôi một rời nhau.
- Số domatic $(\mathrm{dom}(G))$: số lượng tối đa các tập hợp trong một phân hoạch domatic của đồ thị G.

domatic $=$ domination $+$ chromatic

Mạng cảm biến không dây

- Khu vực cần đo – đỉnh của đồ thị.
- Cảm biến đo được các khu vực lân cận (các đỉnh kề).



Một số ứng dụng khác

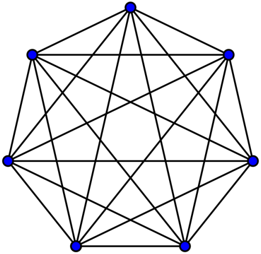
- Phân bổ tài nguyên trong mạng máy tính.
- Chia sẻ tài nguyên giữa các thị trấn trong một khu vực địa lý.
- ...

Nội dung

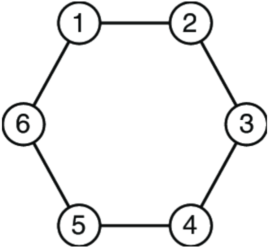
Kết quả về chặn trên của số domatic

Định lí 1

Với mọi đồ thị G, ta có $\mathrm{dom}(G) \leq \delta(G) + 1$.



(a) Đồ thị đầy đủ K_7

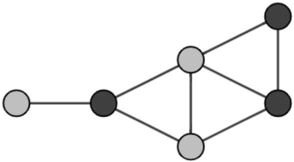


(b) Đồ thị chu trình C_6

Kết quả về chặn dưới của số domatic

Định lí 2

Nếu G liên thông thì $\mathrm{dom}(G) \geq 2$.



Nội dung

Mô hình 1

Input: $k \in \mathbb{N}^*$, đồ thị G.
Output: Tồn tại phân hoạch domatic gồm k phần của G?

Mô hình 1 (tt.)

- Đặt $N = |V(G)|$.
- Xét kN biến nhị phân $x_{u,i}$, với $u \in V(G)$ và $i \in \{1, \ldots, k\}$,
  - $x_{u,i} = 1$ nếu đỉnh u được phân hoạch vào tập thống trị i.
  - $x_{u,i} = 0$ nếu ngược lại.

Mô hình 1 (tt.)

Do mỗi đỉnh phải thuộc đúng 1 trong k tập nên:

$$\sum_{i=1}^{k} x_{u,i} = 1, \quad \forall u \in V(G).$$

Mô hình 1 (tt.)

Do mỗi đỉnh phải thuộc đúng 1 trong k tập nên:

$$\sum_{i=1}^{k} x_{u,i} = 1, \quad \forall u \in V(G).$$

Vì mỗi tập trong k tập trên là một tập thống trị nên :

$$x_{u,i} + \sum_{v \in N(u)} x_{v,i} \geq 1, \quad \forall u \in V(G),\ i = 1, \ldots, k.$$

Mô hình 1 (tt.)

Tồn tại hay không phân hoạch domatic gồm k phần của G?

$$
\begin{aligned}
\text{maximize} \quad & h = 1 \\
\text{subject to} \quad & \sum_{i=1}^{k} x_{u,i} = 1, && u \in V(G), \\
& x_{u,i} + \sum_{v \in N(u)} x_{v,i} \geq 1, && i = 1, \ldots, k, \\
& x_{u,i} \in \{0, 1\}, && u \in V(G),\ i = 1, \ldots, k.
\end{aligned}
$$

Mô hình 1 (tt.)

Lưu ý:

- Ta chỉ cần xét k từ 2 đến $\delta(G) + 1$.
- Số k lớn nhất thoả mãn bài toán trên chính là $\operatorname{dom}(G)$.

Mô hình 2

Xét k tập $V_1, \ldots, V_k$ với $k = \delta(G) + 1$.

Đặt $y_1, \ldots, y_k$ là các biến nhị phân thoả mãn với mỗi $i = 1, \ldots, k$:

- $y_i = 1$ nếu tập V_i được chọn trong phân hoạch cần tìm.
- $y_i = 0$ nếu ngược lại.

Hàm mục tiêu: $h = \sum_{i=1}^{k} y_i$.

Mô hình 2 (tt.)

- Đặt $N = |V(G)|$.
- Xét kN biến nhị phân $x_{u,i}$, với $u \in V(G)$ và $i \in \{1, \ldots, k\}$,
  - $x_{u,i} = 1$ nếu đỉnh u được phân hoạch vào tập thống trị i.
  - $x_{u,i} = 0$ nếu ngược lại.

Mô hình 2 (tt.)

Do mỗi đỉnh chỉ thuộc đúng 1 trong k tập nên:

$$\sum_{i=1}^{k} x_{u,i} = 1, \quad \forall u \in V(G).$$

Mô hình 2 (tt.)

Do mỗi đỉnh chỉ thuộc đúng 1 trong k tập nên:

$$\sum_{i=1}^{k} x_{u,i} = 1, \quad \forall u \in V(G).$$

Vì mỗi tập trong k tập trên là một tập thống trị nên :

$$x_{u,i} + \sum_{v \in N(u)} x_{v,i} - y_i \geq 0, \quad \forall u \in V(G),\ i = 1, \ldots, k$$

Mô hình 2 (tt.)

Do mỗi đỉnh chỉ thuộc đúng 1 trong k tập nên:

$$\sum_{i=1}^{k} x_{u,i} = 1, \quad \forall u \in V(G).$$

Vì mỗi tập trong k tập trên là một tập thống trị nên :

$$x_{u,i} + \sum_{v \in N(u)} x_{v,i} - y_i \geq 0, \quad \forall u \in V(G),\ i = 1, \ldots, k$$

Cần chắc chắn rằng u chỉ được thêm vào các tập V_i được chọn:

$$y_i - x_{u,i} \geq 0, \quad \forall u \in V(G),\ i = 1, \ldots, k$$

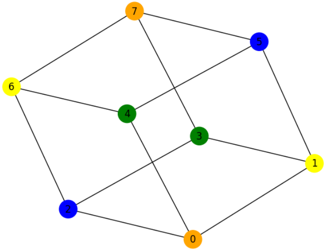
Mô hình 2 (tt.)

Mô hình ILP cho bài toán số domatic của đồ thị.

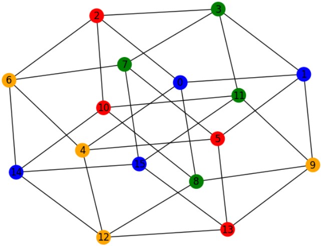
$$
\begin{aligned}
\text{maximize} \quad & \sum_{i=1}^{k} y_i \\
\text{subject to} \quad & \sum_{i=1}^{k} x_{u,i} = 1, && u \in V(G), \\
& x_{u,i} + \sum_{v \in N(u)} x_{v,i} - y_i \geq 0, && u \in V(G),\ i = 1, \ldots, k, \\
& x_{u,i} \in \{0, 1\}, && u \in V(G),\ i = 1, \ldots, k, \\
& y_i \in \{0, 1\}, && i = 1, \ldots, k.
\end{aligned}
$$

Nội dung

Số domatic của đồ thị siêu khối Q_n

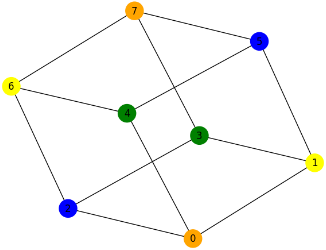


(a) $\mathrm{dom}(Q_3) = 4$

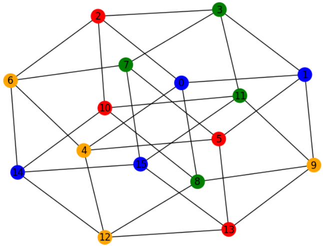


(b) $\mathrm{dom}(Q_4) = 4$

Số domatic của đồ thị siêu khối Q_n



(a) $\mathrm{dom}(Q_3) = 4$



(b) $\mathrm{dom}(Q_4) = 4$

Tổng quát: $\mathrm{dom}(Q_{2^k-1}) = \mathrm{dom}Q_{2^k} = 2^k$ ([Zel82a]).

[CH01] L. Caccetta and S. P. Hill. "Branch and cut methods for network optimization". In: *Mathematical and Computer Modelling* 33.4–5 (Feb. 2001), pp. 517–532.

[CH75] E. Cockayne and S. Hedetniemi. "Optimal domination in graphs". In: *IEEE Transactions on Circuits and Systems* 22.11 (Nov. 1975), pp. 855–857.

[CJP83] H. Crowder, E. L. Johnson, and M. Padberg. "Solving Large-Scale Zero-One Linear Programming Problems". In: *Operations Research* 31.5 (Oct. 1983), pp. 803–834.

[FYK00] Satoshi Fujita, Masafumi Yamashita, and Tiko Kameda. "A Study on r-Configurations—A Resource Assignment Problem on Graphs". In: *SIAM Journal on Discrete Mathematics* 13.2 (Jan. 2000), pp. 227–254.

Tham khảo II

[Gom58] R. E. Gomory. "Outline of an algorithm for integer solutions to linear programs". In: *Bulletin of the American Mathematical Society* 64.5 (1958), pp. 275–278.

[LD60] A. H. Land and A. G. Doig. "An Automatic Method of Solving Discrete Programming Problems". In: *Econometrica* 28.3 (July 1960), p. 497.

[MW] T. Moscibroda and R. Wattenhofer. "Maximizing the Lifetime of Dominating Sets". In: *19th IEEE International Parallel and Distributed Processing Symposium*. IEEE.

[PR87] M. Padberg and G. Rinaldi. "Optimization of a 532-city symmetric traveling salesman problem by branch and cut". In: *Operations Research Letters* 6.1 (Mar. 1987), pp. 1–7.

[Tac16] Leonardo Taccari. "Integer programming formulations for the elementary shortest path problem". In: *European Journal of Operational Research* 252.1 (July 2016), pp. 122–130.

[Yu+14] J. Yu et al. "Domatic partition in homogeneous wireless sensor networks". In: *Journal of Network and Computer Applications* 37 (Jan. 2014), pp. 186–193.

[Zel82a] B. Zelinka. "Domatic numbers of cube graphs". In: *Mathematica Slovaca* 32.2 (Feb. 1982), pp. 117–119.

[Zel82b] B. Zelinka. "Domatic numbers of cube graphs". In: *Mathematica Slovaca* 32.2 (1982), pp. 117–119.