



Một số kĩ thuật tối ưu quy hoạch động

Sinh viên: Nguyễn Quang Quý - 20153108



Nội dung trình bày

1. Kỹ thuật bao lồi
2. Kỹ thuật chia để trị



1. Kỹ thuật bao lồi

Lớp bài toán áp dụng: $dp[i] = \min_{j < i} (dp[j] + b[j] * a[i])$

Với điều kiện: $b[j] \geq b[j + 1]$



Bài toán cơ sở

Bài toán: Cho một tập n đường thẳng $D = \{d_1, d_2, \dots, d_n\}$ trong đó $(d_i) \quad y_i = a_i x + b_i, a_i \geq 0$ và k điểm trên trục Ox : p_1, p_2, \dots, p_k . Tìm q_1, q_2, \dots, q_k trong đó mỗi giá trị q_ℓ được định nghĩa như sau:

$$q_\ell = \min_{1 \leq i \leq n} (a_i \cdot p_\ell + b_i)$$

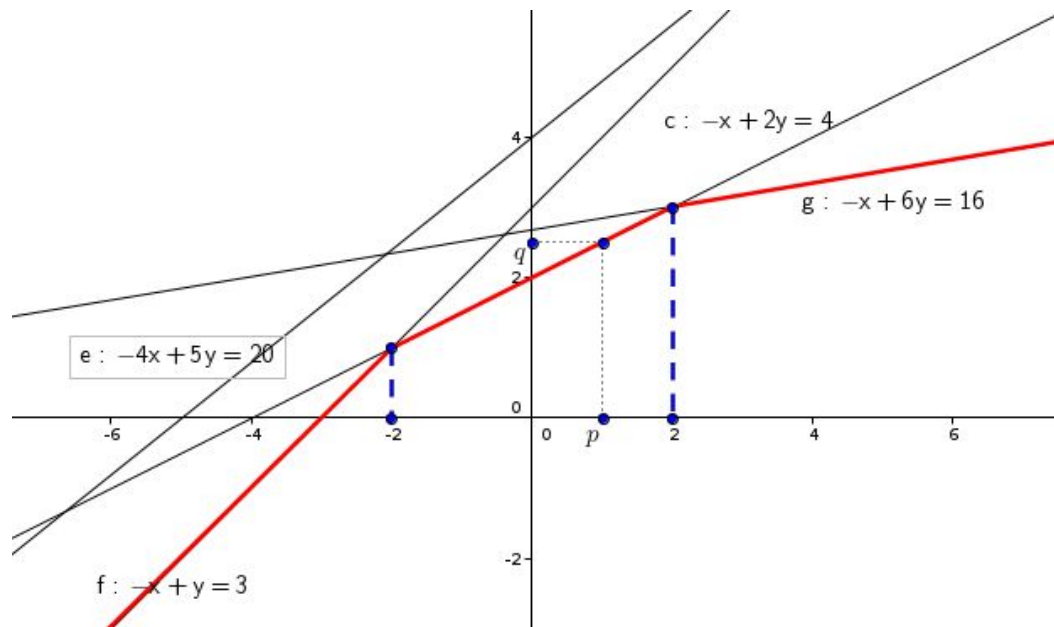
Ví dụ: D gồm 4 đường:

- $(d_1) - x + y = 3$
- $(d_2) - x + 2y = 4$
- $(d_3) - x + 6y = 16$
- $(d_4) - 4x + 5y = 20$

Bài toán cơ sở

Nhận xét:

1. Tồn tại một số đường thẳng dư thừa trong D
2. Đi từ trái sang phải, hệ số góc của các đường thuộc bao lồi giảm dần
3. Biểu diễn các bao lồi bằng các khoảng giá trị và đoạn thẳng tương ứng





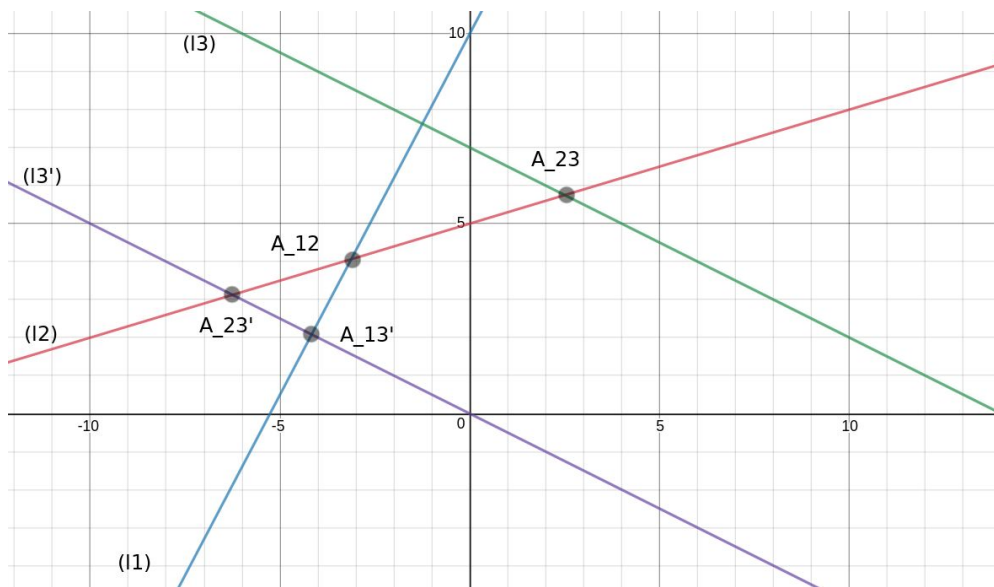
Bài toán cơ sở

Ý tưởng:

1. Sắp xếp các đường trong D giảm dần theo hệ số góc
2. Thêm lần lượt các đường vào Stack, mỗi đường thêm vào sẽ kiểm tra khả năng thay thế với các đường trong Stack

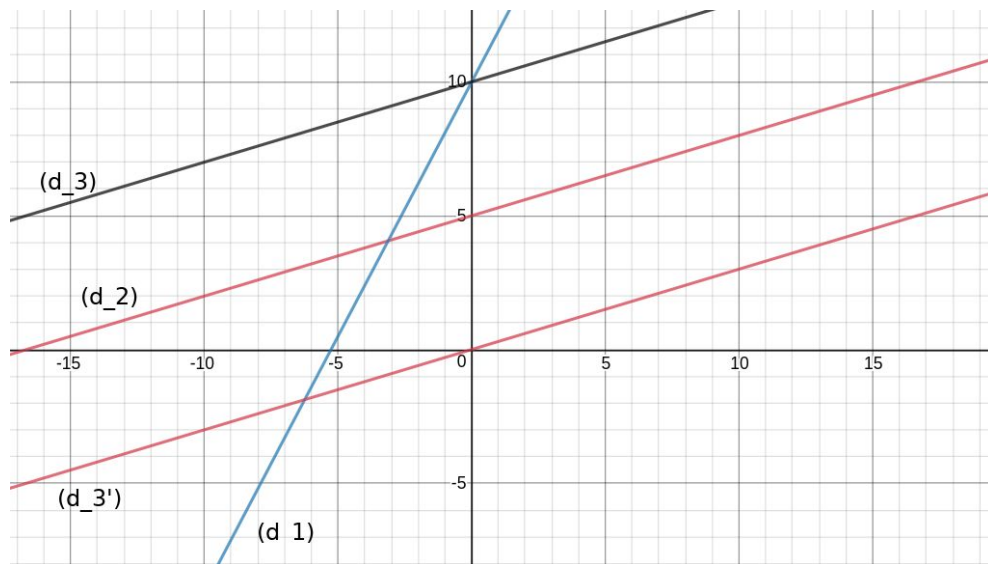
Bài toán cơ sở

Trường hợp D không tồn tại cặp song song



Bài toán cơ sở

Trường hợp D tồn tại cặp song song





Bài toán cơ sở

Độ phức tạp $O(n \log n)$

```
FINDHULL( $d_1, d_2, \dots, d_n$ ):  
  sort  $\{d_1, d_2, \dots, d_n\}$  by decreasing order of slope  
  Stack  $S \leftarrow \emptyset$   
   $S.push(d_1)$   
   $I_1 \leftarrow [-\infty, +\infty]$   
   $m \leftarrow 1$   
  for  $i \leftarrow 2$  to  $n$   
     $d \leftarrow S.peek()$  [[examine the top element]]  
     $x_p \leftarrow \text{FINDINTERSECTIONX}(d, d_i)$   
    while  $x_p \leq \text{left}(I_m)$  [[found a redundant line]]  
       $S.pop()$   
       $m \leftarrow m - 1$   
       $d \leftarrow S.peek()$   
       $x_p \leftarrow \text{FINDINTERSECTIONX}(d_i, d)$   
     $S.push(d_i)$   
     $I_m \leftarrow [\text{left}(I_m), x_p]$   
     $I_{m+1} \leftarrow [x_p, +\infty]$   
    associate  $I_{m+1}$  with  $d_i$   
     $m \leftarrow m + 1$ 
```

Minh họa thuật toán

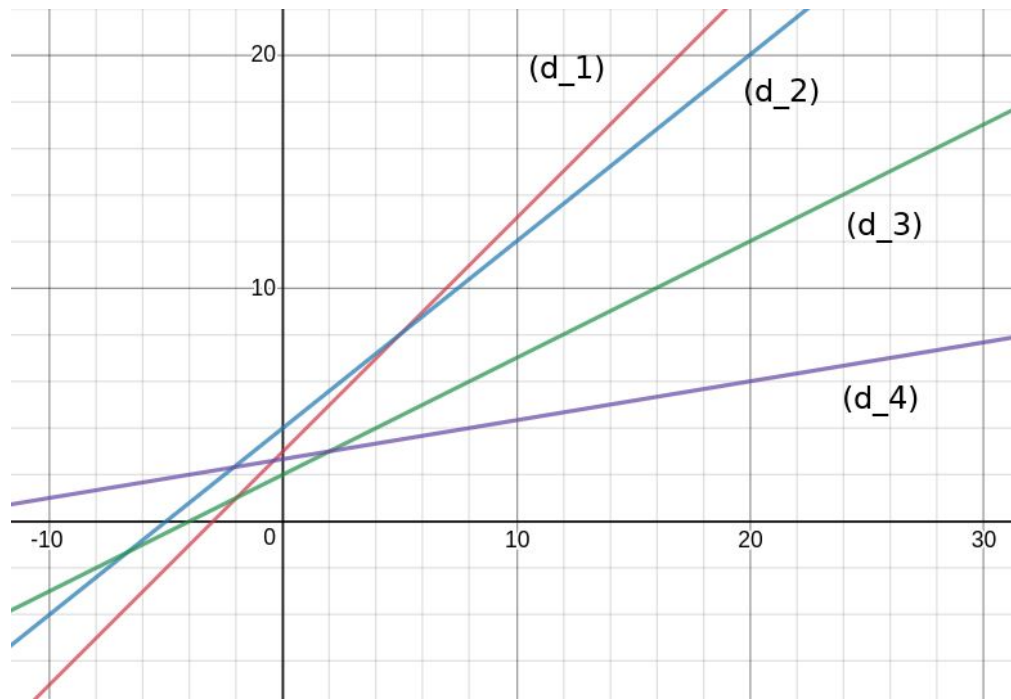
Sắp xếp các đường thẳng theo độ dốc giảm dần, ta được:

+ $(d_1) -x + y = 3$

+ $(d_2) -4x + 5y = 20$

+ $(d_3) -x + 2y = 4$

+ $(d_4) -x + 6y = 16$



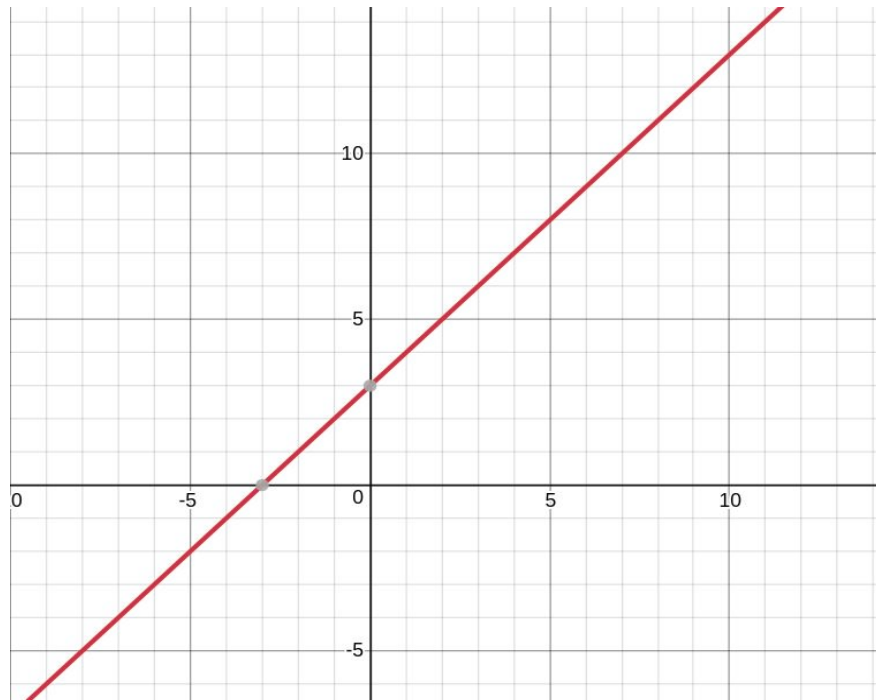
Minh họa thuật toán

Khởi tạo:

$$+ S = \{d_1\}$$

$$+ I = \{I_0\}$$

với $I_0 = [-\infty, +\infty]$



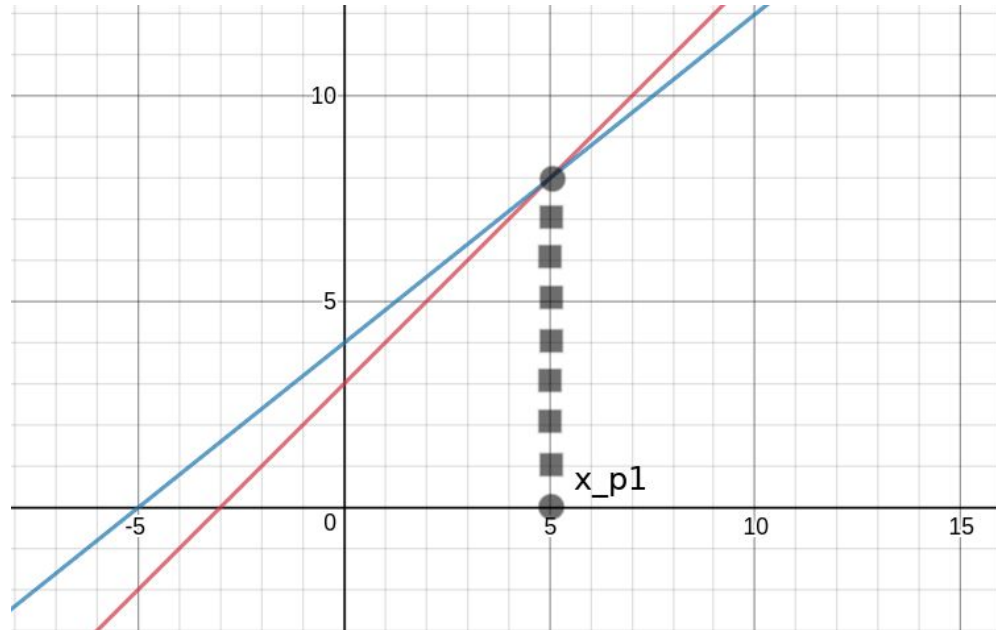
Minh họa thuật toán

Loop 1: Kiểm tra d_2

+ $S = \{d_1, d_2\}$

+ $I = \{I_0, I_1\}$

với $I_0 = [-\infty, x_{p1}]$, $I_1 = [x_{p1}, +\infty]$



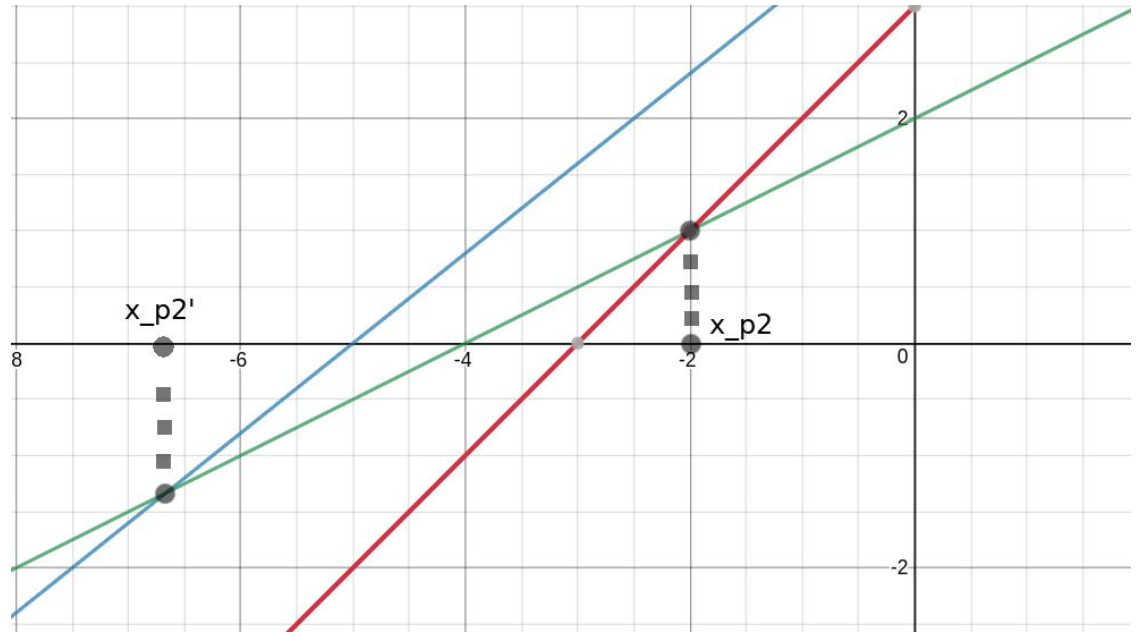
Minh họa thuật toán

Loop 2: Kiểm tra d_3

+ $S = \{d_1, d_3\}$

+ $I = \{I_0, I_1\}$

với $I_0 = [-\infty, x_{p2}]$, $I_1 = [x_{p2}, +\infty]$



Minh họa thuật toán

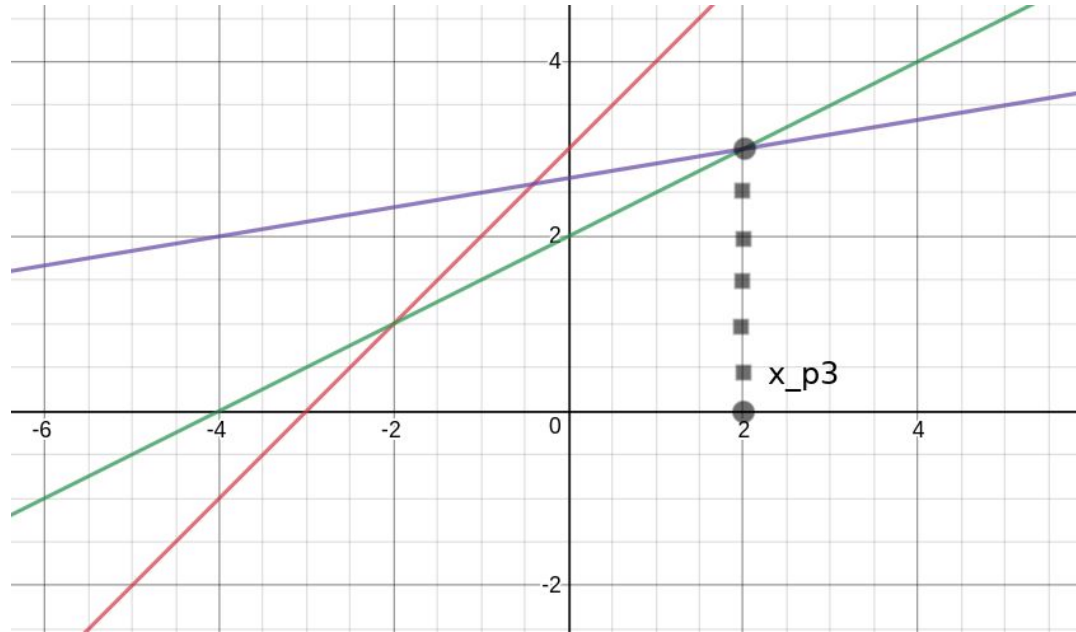
Loop 3: Kiểm tra d_4

+ $S = \{d_1, d_3, d_4\}$

+ $I = \{I_0, I_1, I_2\}$

với $I_0 = [-\infty, x_{p2}]$, $I_1 = [x_{p2}, x_{p3}]$

$I_2 = [x_{p3}, +\infty]$



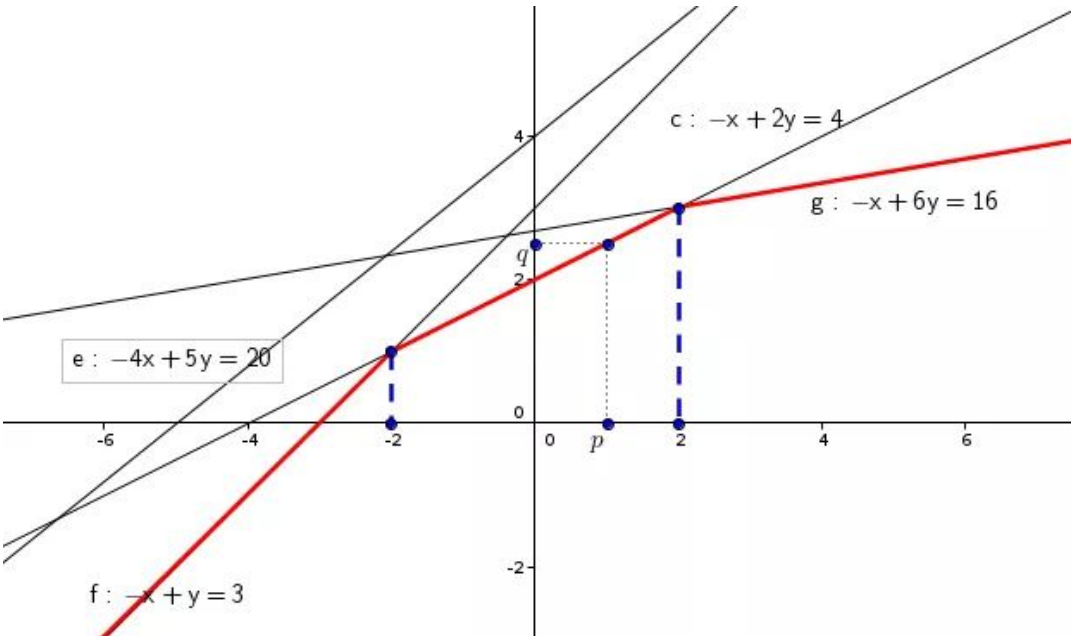
143

(c) (1), (2), (3)

1 1 (1 1 1)

4.1 [11] [11] [11]

Introduction





Bài toán áp dụng

Đề bài: Cho $n \leq 300000$ cặp số (x, y) ($1 \leq x, y \leq 1000000$). Ta có thể nhóm một vài cặp số lại thành một nhóm. Việc nhóm này sẽ mất một chi phí. Giả sử một nhóm gồm các cặp số a_1, a_2, \dots, a_m thì chi phí cho nhóm này sẽ là $\max(x_{a_1}, x_{a_2}, \dots, x_{a_m}) * \max(y_{a_1}, y_{a_2}, \dots, y_{a_m})$.

Yêu cầu: Tìm cách phân nhóm sao cho tổng chi phí là nhỏ nhất.

- Hạn chế thời gian: 0.661 giây
- Hạn chế bộ nhớ: 1536Mb

group.inp	group.out
4	500
100 1	
15 15	
20 5	
1 100	

Bài toán áp dụng

Nhận xét: Với hai cặp số (x_1, y_1) và (x_2, y_2) mà $x_1 > x_2$ và $y_1 > y_2$ thì ta nói (x_2, y_2) là dư thừa vì ta có thể thêm nó vào nhóm với cặp (x_1, y_1) mà không làm ảnh hưởng tới chi phí.

+ $C[i]$ - chi phí cực tiểu để phân nhóm được i cặp đầu tiên

$$C[i] = \min_{1 \leq j < i} (C[j] + x_i \cdot y_{j+1})$$

$$y = a.x + b$$



Bài toán áp dụng

Nhận xét:

- + Hệ số góc đã được sắp xếp giảm dần
- + Điểm truy vấn được sắp xếp theo thứ tự tăng dần

Trong thủ tục tìm kiếm interval cho x_i sẽ không cần xem xét tới các interval nằm bên trái x_{i-1}



Bài toán áp dụng

Nhận xét:

- + Hệ số góc đã được sắp xếp giảm dần
- + Điểm truy vấn được sắp xếp theo thứ tự tăng dần

Trong thủ tục tìm kiếm interval cho x_i sẽ không cần xem xét tới các interval nằm bên trái x_{i-1}

Bài toán áp dụng

Cài đặt:

- B là mảng lưu các hệ số góc,
- A là mảng lưu các điểm truy vấn

Độ phức tạp: Trong thủ tục cài đặt quy hoạch động:

- Tìm interval trong thời gian $O(n)$
- Loại bỏ đường dư thừa $O(n)$

Tổng thời gian: $O(n \log n)$

```
FASTFASTDYNAMIC( $A[1, 2, \dots, n], B[1, 2, \dots, n]$ ):  
   $d_1 \leftarrow B[1]x + C[1]$   
  Stack  $S \leftarrow \emptyset$   
   $S.push(d_1)$   
   $I_1 \leftarrow [-\infty, +\infty]$   
  associate  $I_1$  with  $A[1]$   
   $m \leftarrow 1$   
  for  $i \leftarrow 2$  to  $n$   
     $I_j \leftarrow$  the interval associated with  $A[i-1]$   
     $I \leftarrow \emptyset$   
     $\ell \leftarrow j-1$   
    while  $I = \emptyset$   
       $\ell \leftarrow \ell + 1$   
      if  $A[i] \in I[\ell]$   
         $I \leftarrow I[\ell]$   
     $d \leftarrow$  the line associated with  $I$   
     $C[i] \leftarrow a \cdot A[i] + b$  [[assuming (d)  $y = ax + b$ ]]  
     $d_i \leftarrow B[i]x + C[i]$   
     $d \leftarrow S.peek()$  [[examine the top element]]  
     $x_p \leftarrow \text{FINDINTERSECTIONX}(d, d_i)$   
    while  $x_p \leq \text{left}(I_m)$  [[found a redundant line]]  
       $S.pop()$   
       $m \leftarrow m - 1$   
       $d \leftarrow S.peek()$   
       $x_p \leftarrow \text{FINDINTERSECTIONX}(d_i, d)$   
     $S.push(d_i)$   
     $I_m \leftarrow [\text{left}(I_m), x_p]$   
     $I_{m+1} \leftarrow [x_p, +\infty]$   
    if  $\ell < m$  [[the line associated with  $I_\ell$  is not redundant]]  
      associate  $I_\ell$  with  $A[i]$   
    else  
      if  $A[i] \in I_m$   
        associate  $I_m$  with  $A[i]$   
      else  
        associate  $I_{m+1}$  with  $A[i]$   
    associate  $I_{m+1}$  with  $d_i$   
     $m \leftarrow m + 1$   
  return  $C[n]$ 
```



Kết quả

Bộ test:

- 1 test với $n = 5000$ cho trường hợp các cặp số có cùng giá trị y , còn giá trị x được sinh ngẫu nhiên (trường hợp tất cả các đường thẳng đều song song với nhau)
- 1 test với $n = 20$ cho trường hợp tất cả các cặp số đều giống nhau
- 1 test với $n = 30000$, trong đó các cặp số được sắp xếp giảm dần theo x
- 5 test với n nằm trong khoảng $3000 - 5000$
- 5 test với n nằm trong khoảng $6000 - 10000$
- 10 test với n nằm trong khoảng $10000 - 29999$

Kết quả

Themis

The image displays two side-by-side screenshots of the Themis application interface. Both windows show the test results for a group named 'gondolas'. The top bar of each window indicates the group name and a checkmark, followed by the score '230.00'. The main content area lists test results for tests 06 through 12 (left) and 18 through 23 (right). Each entry includes a test ID, a time value, and a status 'Kết quả khớp đáp án!'. At the bottom of each window, there are three checkboxes: 'Hiện test đúng', 'Hiện test sai', and 'Hiện test đúng một phần', all of which are checked. Below these checkboxes are two buttons: 'Copy' and 'Đóng' (Close).

Left Window Test Results:

- Test06: 10.00
Thời gian ≈ 0.079437600 giây
Kết quả khớp đáp án!
- Test07: 10.00
Thời gian ≈ 0.078679200 giây
Kết quả khớp đáp án!
- Test08: 10.00
Thời gian ≈ 0.080677500 giây
Kết quả khớp đáp án!
- Test09: 10.00
Thời gian ≈ 0.081664100 giây
Kết quả khớp đáp án!
- Test10: 10.00
Thời gian ≈ 0.081527400 giây
Kết quả khớp đáp án!
- Test11: 10.00
Thời gian ≈ 0.084043100 giây
Kết quả khớp đáp án!
- Test12: 10.00

Right Window Test Results:

- Thời gian ≈ 0.091991000 giây
Kết quả khớp đáp án!
- Test18: 10.00
Thời gian ≈ 0.094421800 giây
Kết quả khớp đáp án!
- Test19: 10.00
Thời gian ≈ 0.098549300 giây
Kết quả khớp đáp án!
- Test20: 10.00
Thời gian ≈ 0.100379800 giây
Kết quả khớp đáp án!
- Test21: 10.00
Thời gian ≈ 0.104485300 giây
Kết quả khớp đáp án!
- Test22: 10.00
Thời gian ≈ 0.104693900 giây
Kết quả khớp đáp án!
- Test23: 10.00
Thời gian ≈ 0.104012700 giây
Kết quả khớp đáp án!



Kết quả

SPOJ - Problem GROUP/ACQUIRE(USACO 2007)

ID	GIỜ NỘP	Tài khoản:	PROBLEM	RESULT	TIME	MEM	NGÔN NGỮ
22907082	2018-12-19 11:39:08	Zoe2i	Phân nhóm	accepted edit run	0.20	29M	CPP14



2. Kỹ thuật chia để trị

Lớp bài toán áp dụng: $dp[i][j] = \min_{k < j} dp[i-1][k] + C[k][j]$

Với điều kiện: $R[i][j] \leq R[i][j+1]$

$R[i][j]$ - điểm chia k để đạt được tối ưu trong công thức tính $dp[i][j]$.



Bài toán áp dụng

Đề bài: n người trong hàng đợi lên máy bay có k khoang. Cần sắp xếp mọi người vào các khoang sao cho:

- Mỗi khoang có ít nhất 1 người
- Chi phí sắp xếp ít nhất

+ Chi phí - tổng mức độ khó chịu trong k khoang.

+ Mức độ khó chịu trong một khoang - mức độ khó chịu giữa mỗi người với nhau trong khoang đó.

+ Ma trận đối xứng A - $A[i][j]$ là mức độ khó chịu giữa người i với j .

+ Ràng buộc: $1 \leq n \leq 4000$, $1 \leq k \leq \min(n, 800)$, $0 \leq A[i][j] \leq 9$



Bài toán áp dụng

Ví dụ: Khoảng 1: {1, 2}, Khoảng 2: {3}.

ciel_gondolas.inp	ciel_gondolas.out
3 2 0 2 0 2 0 3 0 3 0	2



Phân tích

- + $C[i][j]$ - Chi phí để xếp j người đầu vào i khoang đầu

$$C[i][j] = \begin{cases} 0, & i = j \\ \min_{i \leq \ell < j} C[i-1, \ell] + D[\ell+1, j], & i < j \end{cases}$$

- + $D[i][j]$ - Chi phí để xếp người thứ i tới người thứ j vào cùng một khoang

$$D[i][j] = \sum_{i \leq p < q \leq j} A[p, q]$$



Phân tích

+ $R[i][j]$ - Điểm chia trong công thức

$$R[i][j] = \operatorname{argmin}_{i \leq \ell < j} (C[i-1, \ell] + D[\ell+1, j])$$

Chứng minh được: $R[i][j] \leq R[i][j+1]$

Nhận xét: Điểm chia của $C[i][j]$ nằm trong khoảng $R[i][j-1]$ tới $R[i][j+1]$



Cài đặt

```
FASTDYNAMICPROGRAM( $A[1, 2, \dots, n][1, 2, \dots, n]$ ):  
  COMPUTEDISIMILARITY( $A[1, 2, \dots, n][1, 2, \dots, n]$ )  
  for  $i \leftarrow 1$  to  $k$   
     $R[i, i] \leftarrow i$   
     $R[i, n] \leftarrow \text{COMPUTER}(i, n, i, n)$   
    DIVCONDYNAMIC( $i, i + 1, n - 1, i, R[i, n]$ )  
  return  $C[k, n]$ 
```



Cài đặt

Tính toán mảng D trong thời gian $O(n^2)$

COMPUTEDISSIMILARITY($A[1, 2, \dots, n][1, 2, \dots, n]$):

for $i \leftarrow 1$ to n

$D[i, i] \leftarrow 0$

$B[i, i] \leftarrow 0$

for $d \leftarrow 1$ to $n - 1$

for $i \leftarrow 1$ to $n - d$

$B[i, i + d] \leftarrow B[i + 1, i + d] + A[i, i + d] \quad [[B[i, j] = \sum_{i \leq \ell < j} A[\ell, j]]]$

$D[i, i + d] \leftarrow D[i, i + d - 1] + B[i, i + d]$



Cài đặt

ComputeR - tính $C[i][j]$ và trả về $R[i][j]$ trong thời gian $O(R-L)$

COMPUTER (i, j, L, R) :

$\text{tmp} \leftarrow +\infty$

$p \leftarrow L$

for $\ell \leftarrow L$ to R

$\text{tmp} \leftarrow \min\{\text{tmp}, C[i-1, \ell] + D[\ell+1, j]\}$

$p \leftarrow \ell$

$C[i, j] \leftarrow \text{tmp}$

return p



Cài đặt

DivConDynamic - tính $C[i][x], \dots, C[i][y]$

DivConDynamic(i, x, y, L, R):

if $y - x \leq 2$

 use brute force

else

$m \leftarrow \lfloor \frac{x+y}{2} \rfloor$

$R[i, m] \leftarrow \text{COMPUTER}(i, m, L, R)$

 DivConDynamic($i, x, m - 1, L, R[i, m]$)

 DivConDynamic($i, m + 1, y, R[i, m], R$)



Độ phức tạp thuật toán

$T(n, m)$ - thời gian tính của `DivConDynamic(i, x, y, L, R)` với $n = y - x$ và $m = R - L$

$$T(n, m) = \begin{cases} O(n), & \text{if } m = O(1) \\ O(m), & \text{if } n = O(1) \\ T(\frac{n}{2}, k) + T(\frac{n}{2}, m - k) + O(m), & \text{otherwise} \end{cases}$$

⇒ $T(n, m) = O((m+n)\log n)$

Kết luận: Độ phức tạp thuật toán là $O(n^2 + kn\log n)$



Kết quả

Bộ test:

- 1 test với n trong khoảng 10 - 50 và $k = n+1$
- 5 test với n trong khoảng 100 - 500 và k trong khoảng 1 - 20
- 5 test với n trong khoảng 300 - 500 và k trong khoảng 100 - $n/2$
- 5 test với n trong khoảng 1000 - 2000, k trong khoảng 1 - 200
- 5 test với n trong khoảng 2000 - 3999, k trong khoảng 200 - 800\$
- 2 test với $n = 4000$ với k trong khoảng 1 - 100 và k trong khoảng 500 - 800

Kết quả

Themis

✓ ceil_and_gondolas		✓ group	✓ ceil_and_gondolas	✓ group
✓ 230.00		✓ 230.00	✓ 230.00	✓ 230.00
<div><div>Chi tiết chấm</div><div><p>Thời gian ≈ 0.057886100 giây Kết quả khớp đáp án!</p><ul style="list-style-type: none">◆ Test06: 10.00 Thời gian ≈ 0.059716500 giây Kết quả khớp đáp án!◆ Test07: 10.00 Thời gian ≈ 0.279885200 giây Kết quả khớp đáp án!◆ Test08: 10.00 Thời gian ≈ 0.315913200 giây Kết quả khớp đáp án!◆ Test09: 10.00 Thời gian ≈ 0.333868400 giây Kết quả khớp đáp án!◆ Test10: 10.00 Thời gian ≈ 0.423092600 giây Kết quả khớp đáp án!◆ Test11: 10.00 Thời gian ≈ 0.696860800 giây Kết quả khớp đáp án!</div><div><input checked="" type="checkbox"/> Hiện test đúng<input checked="" type="checkbox"/> Hiện test sai<input checked="" type="checkbox"/> Hiện test đúng một phần</div><div><div>Copy</div><div>Đóng</div></div></div>			<div><div>Chi tiết chấm</div><div><p>Kết quả khớp đáp án!</p><ul style="list-style-type: none">◆ Test18: 10.00 Thời gian ≈ 1.097489800 giây Kết quả khớp đáp án!◆ Test19: 10.00 Thời gian ≈ 1.045635600 giây Kết quả khớp đáp án!◆ Test20: 10.00 Thời gian ≈ 1.135317100 giây Kết quả khớp đáp án!◆ Test21: 10.00 Thời gian ≈ 1.391780600 giây Kết quả khớp đáp án!◆ Test22: 10.00 Thời gian ≈ 0.077104400 giây Kết quả khớp đáp án!◆ Test23: 10.00 Thời gian ≈ 0.076857000 giây Kết quả khớp đáp án!</div><div><input checked="" type="checkbox"/> Hiện test đúng<input checked="" type="checkbox"/> Hiện test sai<input checked="" type="checkbox"/> Hiện test đúng một phần</div><div><div>Copy</div><div>Đóng</div></div></div>	

Kết quả

Codeforces Round #190 (Div. 1) Problem E. Ciel and Gondolas

General										
#	Author	Problem	Lang	Verdict	Time	Memory	Sent	Judged		
47160065	Practice: zoe2i	321E - 19	GNU C++11	Accepted	872 ms	200964 KB	2018-12-17 12:16:52	2018-12-17 12:16:52		<button>Compare</button>