

Phân bổ tối thiểu chi phí (Minimum Cost Flow)

Giới thiệu

Đây là chương trình **hoạch định phân phối tối thiểu**, giúp tìm cách vận chuyển hàng hóa từ các **trạm phát (nơi có hàng)** tới các **trạm thu (nơi cần hàng)** sao cho **tổng chi phí vận chuyển là thấp nhất**.

Chương trình áp dụng **thuật toán dòng chi phí cực tiểu (Min-Cost Flow)** – một bài toán kinh điển trong lĩnh vực tối ưu hóa mạng (network optimization).

Mục tiêu bài toán

Cho trước:

- Một tập **trạm phát (supply nodes)**; mỗi trạm có lượng hàng **Cung (Supply)**.
- Một tập **trạm thu (demand nodes)**; mỗi trạm có nhu cầu **Nhu cầu (Demand)**.
- Một bảng **chi phí vận chuyển (Cost Matrix)**; biểu diễn chi phí từ mỗi trạm phát đến mỗi trạm thu.

Nhiệm vụ: tìm kế hoạch phân phối hàng sao cho:

- Tổng nhu cầu được đáp ứng (hỗ trợ tối đa có thể).
- **Tổng chi phí vận chuyển là nhỏ nhất**.

Dữ liệu đầu vào

Chương trình sẽ đọc dữ liệu từ một file Excel có 3 sheet:

1. `TramPhat` – Thông tin các trạm phát

TenTram	Cung
A1	500
A2	850
A3	450

2. `TramThu` – Thông tin các trạm thu

TenTram	NhuCau
B1	900
B2	300
B3	150
B4	450

3. `ChiPhi` – Ma trận chi phí

TramPhat	B1	B2	B3	B4
A1	31100	18000	45000	8500
A2	26000	25000	42300	22000
A3	29500	14000	45800	19000

Thuật toán giải (Min-Cost Flow)

Dưới đây là mô tả chi tiết **tổng bộ các thuật toán** sẽ được sử dụng trong chương trình:

Bước 1: Xây dựng đồ thị luồng (Flow Network)

- M i **tr m phát** và **tr m thu** c xem là m t **nút (node)** trong th .
- Thêm hai nút c bi t:
- `SOURCE` (ngu n t ng)
- `SINK` (ích t ng)

Các lo i c nh (edges):

1. `SOURCE` Tr m phát: dung l ng = cung c a tr m, chi phí = 0
2. `Tr m phát` Tr m thu: dung l ng r t l n, chi phí = giá v n chuy n
3. `Tr m thu` SINK: dung l ng = nhu c u, chi phí = 0

B c 2: Thu t toán tìm dòng chi phí c c ti u

Ch ng trình s d ng **Successive Shortest Path Algorithm** (Lu ng ng n nh t l p l i):

1. Kh i t o t ng lu ng = 0, t ng chi phí = 0.
2. Trong khi v n còn nhu c u ch a c áp ng:
 - Tìm ** ng i chi phí th p nh t** t `SOURCE` `SINK` trong th d (residual graph) b ng **Dijkstra**.
 - Xác nh **l ng hàng có th g i thêm** (bottleneck).
 - C p nh t:
 - Lu ng m i trên t ng c nh.
 - T ng chi phí += (l ng hàng) × (chi phí trên ng ó).
3. L p l i cho n khi t t c nhu c u c th a mãn.

B c 3: K t qu t i u

Sau khi hoàn t t:

- Ch ng trình s trích xu t b ng phân ph i cu i cùng:
- C t `From` – tr m phát.
- C t `To` – tr m thu.
- C t `Quantity` – s l ng g i.
- `UnitCost` – chi phí n v .
- `Cost` – t ng chi phí cho tuy n ó.

K t qu u ra

File k t qu : **allocation_optimal.xlsx**

Bao g m 4 sheet:

1. `OptimalPlan` – chi ti t k ho ch phân ph i t i u
2. `SupplySummary` – t ng hàng ã g i c at ng tr m phát
3. `DemandSummary` – t ng hàng ã nh n c at ng tr m thu
4. `Summary` – t ng chi phí t i thi u

Cách ch y ch ng trình

1 Cài t môi tr ng

```
python -m venv venv
venv/Scripts/activate
```

```
pip install -r requirements.txt
```

2 Ch y ch ng trình

```
python3 distribution_optimizer_from_file.py du_lieu_phan_phoi.xlsx
```

3 Xem kết quả

- Màn hình sẽ hiển thị kết quả tóm tắt.
- File Excel `allocation_optimal.xlsx` sẽ được tạo ra trong cùng thư mục.

Cấu trúc thư mục

project/

```
distribution_optimizer_from_file.py # Mã nguồn chính  
requirements.txt # Các thư viện cần cài  
du_lieu_phan_phoi.xlsx # File đầu vào mẫu  
allocation_optimal.xlsx # File kết quả (tạo ra)
```