

TRƯỜNG ĐẠI HỌC HỌC VĂN LANG
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN MÔN HỌC DIT0350
NHẬP MÔN TRÍ TUỆ NHÂN TẠO

Chủ đề:

**ỨNG DỤNG GIẢI THUẬT HEURISTIC
TRONG BÀI TOÁN NGƯỜI ĐƯA THU'**

GVHD: Th.S Phan Hồ Viết Trường

TP. Hồ Chí Minh – năm 2022

STT	Họ và tên	MSSV	Vai trò
1	Lâm Chí Phi	197CT22462	Trưởng nhóm, làm file báo cáo, tìm và xây dựng bài toán, source code.
2	Trần Văn Hân	197CT33937	Thành viên, chỉnh sửa báo cáo, làm powerpoint
3	Kiều Anh Phát	197CT09919	Thành viên, chỉnh sửa báo cáo, làm powerpoint
4	Nguyễn Thành Đạt	207CT58538	Thành viên, chỉnh sửa báo cáo, làm powerpoint
5	Nguyễn Phú Đạt	207CT58537	Thành viên, chỉnh sửa báo cáo, làm powerpoint

MỞ ĐẦU

Các kỹ thuật tìm kiếm mù rất kém hiệu quả, nhiều trường hợp không thể áp dụng được. Vì vậy trong báo cáo này chúng em sẽ nghiên cứu các phương pháp tìm kiếm kinh nghiệm hơn bằng cách sử dụng hàm đánh giá để hướng dẫn sự tìm kiếm bằng thuật toán Heuristic.

Khái quát: Áp dụng giải thuật cho bài toán Người đưa thư, Người đưa thư phải đi qua tất cả các điểm cần phát thư, đưa thư và trở về vị trí ban đầu với đường đi là ngắn nhất.

Lý do chọn chủ đề: đẩy nhanh quá trình tìm kiếm, tiết kiệm được nhiều thời gian, dễ dàng hoàn thành tốt công việc của người đưa thư.

MỤC LỤC

BÁO CÁO ĐỒ ÁN MÔN HỌC DIT0350	page.1
MỞ ĐẦU	page.3
CHƯƠNG I: CƠ SỞ LÝ THUYẾT THUẬT TOÁN HEURISTIC.	page.5
1.1. Định nghĩa thuật toán Heuristic.	page.5
1.2. Nguyên lý của thuật toán Heuristic.	page.5
1.3. Ví dụ về thuật toán tham lam.	page.6
1.4. Ứng dụng thực tế.	page.7
1.5. Kết luận chương	page.7
CHƯƠNG II. ÁP DỤNG THUẬT TOÁN HEURISTIC CHO BÀI TOÁN NGƯỜI ĐƯA THU.	page.8
2.1. Sơ lược về bài toán.	8
2.2. dataset và ví dụ dữ liệu cho sẵn.	page.8
2.3. Phương pháp.	page.9
2.4. Mã chương trình.	page.9
2.5. Kết quả bài toán.	page.11
CHƯƠNG III: TỔNG KẾT	page.12
Tài liệu tham khảo	page.12

Danh mục hình ảnh

Hình 0.1 sơ đồ vi du về phương pháp tham lam.....	6
Hình 0.2 sơ đồ kết quả về phương pháp tham lam.....	7
Hình 0.1 Vi du về dữ liệu cho sẵn.....	8
Hình 0.2 kết quả bài toán	11

CHƯƠNG I: CƠ SỞ LÝ THUYẾT THUẬT TOÁN HEURISTIC.

1.1. Định nghĩa thuật toán Heuristic.

Thuật toán Heuristic là một sự mở rộng khái niệm thuật toán, thể hiện cách giải bài toán với các đặc tính như: tìm được lời giải tốt nhưng không hẳn là tốt nhất và giải bài toán thường dễ dàng và nhanh chóng đưa ra kết quả hơn so với giải thuật tối ưu, vì vậy chi phí thấp hơn. Thuật toán thường thể hiện khá thực tế, gần gũi với cách suy nghĩ và hành động của con người.

1.2. Nguyên lý của thuật toán Heuristic.

Có nhiều phương pháp để xây dựng một thuật toán Heuristic, trong đó người ta thường dựa vào một số nguyên lý cơ sở như sau:

Nguyên lý vét cạn thông minh:

Trong một bài toán tìm kiếm nào đó, khi không gian tìm kiếm lớn, ta thường tìm cách giới hạn lại không gian tìm kiếm hoặc thực hiện một kiểu dò tìm đặc biệt dựa vào đặc thù của bài toán để nhanh chóng tìm ra mục tiêu.

Nguyên lý tham lam (Greedy):

Lấy tiêu chuẩn tối ưu (trên phạm vi toàn cục) của bài toán để làm tiêu chuẩn chọn lựa hành động cho phạm vi cục bộ của từng bước (hay từng giai đoạn) trong quá trình tìm kiếm lời giải.

Nguyên lý thứ tự:

Thực hiện hành động dựa trên một cấu trúc thứ tự hợp lý của không gian khảo sát nhằm nhanh chóng đạt được một lời giải tốt.

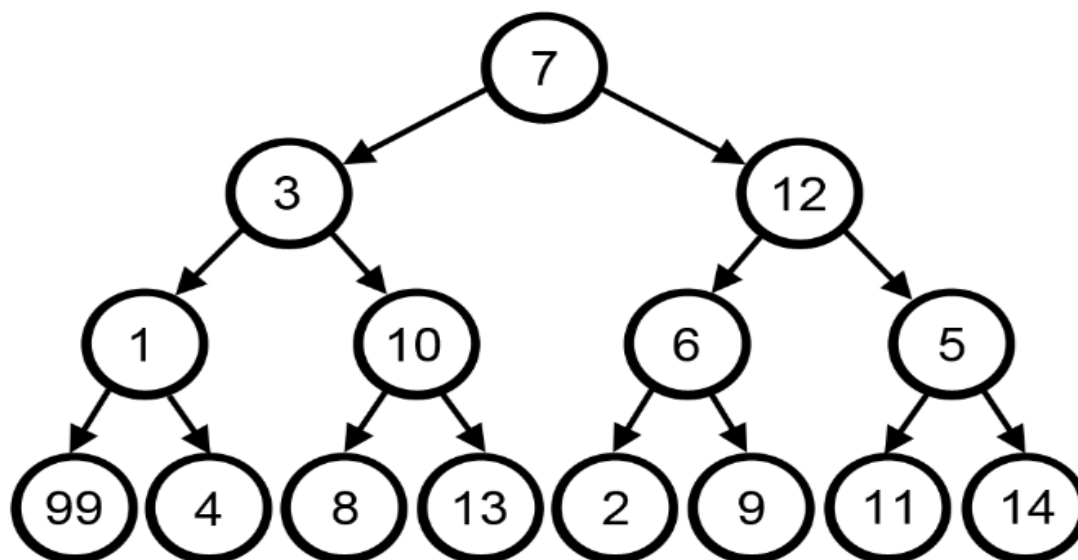
Hàm Heuristic:

Trong việc xây dựng các thuật giải Heuristic, người ta thường dùng các hàm Heuristic. Đó là các hàm đánh giá thô, giá trị của hàm phụ thuộc vào trạng thái hiện tại của bài toán

tại mỗi bước giải. Nhờ giá trị này, ta có thể chọn được cách hành động tương đối hợp lý trong từng bước của thuật giải.

1.3. Ví dụ về thuật toán tham lam.

Bài toán tìm đường đi ngắn nhất từ điểm 7 đến điểm 9.



Hình 0.1 sơ đồ ví dụ về phương pháp tham lam

Bước 1: Chúng ta có thể bắt đầu với điểm 7 và chúng ta có hai con đường để đi tiếp đỉnh 3 và điểm 12.

Bước 2: lấy điểm 3 đi tiếp chúng ta lại có hai con đường là điểm 1 và điểm 10.

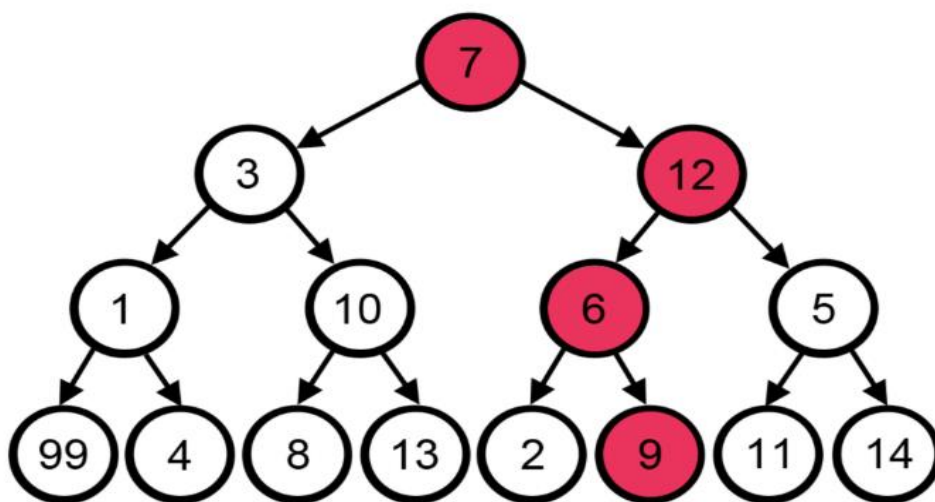
Bước 3: lấy điểm 1 đi tiếp chúng ta lại có hai con đường là điểm 99 và điểm 4 và hai điểm này cũng là hai điểm cuối chúng ta vẫn chưa tìm được kết quả.

Bước 4: chúng ta quay lại điểm 1 và điểm 3, từ điểm 3 tiếp tục đi từ đỉnh 10 chúng ta lại có hai con đường là điểm 8 và điểm 13 cũng là đỉnh cuối chúng ta vẫn chưa tìm được kết quả.

Bước 5: chúng ta quay lại điểm 10, điểm 3 và điểm 7, từ điểm 7 tiếp tục đi xuống điểm 12 chúng ta lại có hai con đường là điểm 6 và điểm 5.

Bước 6: lấy điểm 6 đi tiếp chúng ta lại có hai con đường là điểm 2 và điểm 9, điểm 9 là kết quả mà chúng ta muốn tìm.

Kết luận đường đi ngắn nhất là 7->12->6->9.



Hình 0.2 sơ đồ kết quả về phương pháp tham lam

1.4. Ứng dụng thực tế.

Bài toán tìm đường đi trong thành phố.

Bài toán người bán hàng.

Bài toán phân việc (ứng dụng của nguyên lý thứ tự),

Bài toán hành trình ngắn nhất (ứng dụng nguyên lý Greedy),

Phần mềm Antivirus.

1.5. Kết luận chương

Thuật toán Heuristic có độ phức tạp tốt hơn so với thuật toán tối ưu nhưng lại có những hạn chế như cho ra lời giải không chính xác 100%, dễ sử dụng và có thể áp dụng vào thực tế giúp tối ưu hóa nhiều thứ và có ích cho đời sống công việc hằng ngày.

CHƯƠNG II. ÁP DỤNG THUẬT TOÁN HEURISTIC CHO BÀI TOÁN NGƯỜI ĐƯA THƯ.

2.1. Sơ lược về bài toán.

Áp dụng trên không gian 2 chiều của Euclid, ứng với mỗi tọa độ là vị trí của mỗi địa điểm cần đưa thư trên bản đồ của một thành phố bất kỳ, trong bài toán ví dụ của chúng ta gồm 8 địa điểm cần đưa thư được tạo sẵn trong file dữ liệu csv tên dataset (có thể thay đổi, thêm hoặc bớt đi địa điểm trong dữ liệu bất kỳ lúc nào). Tìm đường đi ngắn nhất để đi qua tất cả các địa điểm cần giao thư, sau khi giao hoàn tất thì quay về vị trí bắt đầu với đoạn đường đi là ngắn nhất.

2.2. dataset và ví dụ dữ liệu cho sẵn.

	A	B	C
1	diadiem	x	y
2	1	37	52
3	2	49	49
4	3	52	64
5	4	20	26
6	5	40	30
7	6	21	47
8	7	17	63
9	8	31	62

Hình 0.1 Ví dụ về dữ liệu cho sẵn

2.3. Phương pháp.

Sử dụng phương pháp hoán đổi:

Ưu tiên tính khoảng cách giữa các địa điểm gần nhau theo cặp, cùng lúc có nhiều cặp địa điểm được tính toán, sau đó được so sánh để hoán đổi vị trí cho nhau để đưa ra lựa chọn tốt hơn, tiếp tục tính toán và hoán đổi đến khi khoảng cách giữa các địa điểm là tốt nhất.

2.4. Mã chương trình.

```
import numpy as np
```

```
import math as math
```

```
def objective_calculator(solution,dataset):
```

```
    cost = 0
```

```
    for i in range(len(solution)-2):
```

```
        cost += euclid_calculator(solution[i], solution[i+1],dataset)
```

```
    return cost
```

```
def euclid_calculator(city_1, city_2, dataset):
```

```
    return math.sqrt((dataset.loc[city_1-1,"x"]-dataset.loc[city_2-1,"x"])**2 +  
    (dataset.loc[city_1-1,"y"]-dataset.loc[city_2-1,"y"])**2)
```

```
def city_swap(city_1,city_2,current_solution,dataset):
```

```

tour_choice=current_solution.copy()
keeper=tour_choice[city_1].copy()
tour_choice[city_1]=tour_choice[city_2].copy()
tour_choice[city_2]=keeper

if objective_calculator(tour_choice,dataset) <
objective_calculator(current_solution,dataset):
    print("Khoảng cách: ",objective_calculator(tour_choice,dataset))
    current_solution=tour_choice
    print("của chuyến đi:", current_solution)
    print("-----")
return current_solution

def main(dataset):
    np.random.seed(28)
    partly_initial_solution= np.random.permutation(range(1,len(dataset)+1))
    initial_solution = np.append(partly_initial_solution, [partly_initial_solution[0]])

    current_solution = initial_solution
    for k in range(10):
        for i in range(1,len(dataset)-1):
            for j in range(i+1,len(dataset)):
                current_solution = city_swap(i,j,current_solution,dataset)

    print("Kết quả:")
    print("Khoảng cách tốt nhất là: ", objective_calculator(current_solution,dataset))

```

```
print("Đường đi tốt nhất là: ", current_solution)
return(current_solution)
```

2.5. Kết quả bài toán.

```
Khoảng cách: 193.9398250599695
của chuyến đi: [7 4 3 1 5 6 8 2 7]
-----
Khoảng cách: 177.8363018376516
của chuyến đi: [7 1 3 4 5 6 8 2 7]
-----
Khoảng cách: 159.83320846427236
của chuyến đi: [7 8 3 4 5 6 1 2 7]
-----
Khoảng cách: 141.11750166472936
của chuyến đi: [7 8 6 4 5 3 1 2 7]
-----
Khoảng cách: 130.19333388518936
của chuyến đi: [7 8 6 4 5 1 3 2 7]
-----
Khoảng cách: 129.01352661564346
của chuyến đi: [7 8 6 4 5 2 3 1 7]
-----
Khoảng cách: 126.0857849517181
của chuyến đi: [7 8 6 4 5 2 1 3 7]
-----
Khoảng cách: 123.35327804974378
của chuyến đi: [7 8 6 4 5 1 2 3 7]
-----
Kết quả:
Khoảng cách tốt nhất là: 123.35327804974378
Đường đi tốt nhất là: [7 8 6 4 5 1 2 3 7]
PS D:\demo bản 1> █
```

Hình 0.2 kết quả bài toán

Vậy đường đi ngắn nhất là từ địa điểm số:

$7 \mapsto 8 \mapsto 6 \mapsto 4 \mapsto 5 \mapsto 1 \mapsto 2 \mapsto 3$ rồi quay về 7.

2.6. Kết luận.

Tuy kết luận bài toán đưa ra chưa chắc chắn chính xác 100% nhưng độ tối ưu vẫn ở mức có thể chấp nhận và áp dụng được trong thực tế.

CHƯƠNG III: TỔNG KẾT

Áp dụng giải thuật cho bài toán Người đưa thư, đi qua tất cả các điểm cần đưa thư và trở về vị trí ban đầu với đường đi là ngắn nhất để tối ưu hóa thời gian, lộ trình, và độ hiệu quả của người đi.

Điểm sáng của giải thuật đó là phương pháp giải quyết vấn đề dựa trên kinh nghiệm và kiến thức thực tiễn. Cho phép bạn giải quyết vấn đề hoặc thu hẹp lựa chọn. Giúp bạn sáng tạo, khám phá các ý tưởng mới.

Lời bạt:

Nhập môn Trí tuệ nhân tạo là học phần quan trọng đối với người lập trình và phân tích dữ liệu. Là cơ sở vững chắc để giải quyết nhiều bài toán, đồng thời cung cấp cho ta hiểu biết về các giải thuật tác động lên dữ liệu, cũng như cách tổ chức dữ liệu hiệu quả để tối ưu bài toán.

Nhóm chúng em đã nghiên cứu và thực hiện đồ án này như là một cách để củng cố và mở rộng kiến thức. Thông qua quá trình thực hiện đồ án, chúng em nắm bắt được những kỹ thuật quan trọng của việc xây dựng cấu trúc dữ liệu và phân tích, thiết kế giải thuật sao cho tối ưu.

Bài toán Người đưa thư mà nhóm chúng em nghiên cứu và trình bày trong báo cáo này là một ví dụ.

Em xin chân thành cảm ơn thầy Phan Hồ Viết Trường đã bổ sung ý tưởng và giúp đỡ chúng em thực hiện đồ án.

Tài liệu tham khảo

<https://viblo.asia/p/thuat-toan-tham-lam-greedy-algorithm-XQZGxozlvwA>

http://www4.hcmut.edu.vn/~huynhqqlinh/TinhocDC/THDC13/Bai03_6.htm

<https://vi.wikipedia.org/wiki/Heuristic>

<https://ladigi.vn/heuristic-la-gi-chi-tiet-ve-heuristic-moi-nhat-2021>

