**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**Viện Công Nghệ Thông Tin Và Truyền Thông**

**---🙠**🕮**🙢---**



**Báo cáo Project III**

Đề tài: Lập lịch trình cho vận chuyển cho bài toán VRP sử dụng Giải thuật di truyền kết hợp thuật toán Split

**Sinh viên: Hoàng Ngọc Lâm - MSSV: 20194089**

***GVHD: Nguyễn Khánh Phương***

*Hà nội, ngày 5 tháng 10 năm 2022.*

Mục lục

[1. Giới thiệu 3](#_Toc116195657)

[2. Thuật toán 3](#_Toc116195658)

[2.1. Giải thuật di truyền 3](#_Toc116195659)

[2.1.1. Biểu diễn nhiễm sắc thể: 3](#_Toc116195660)

[2.1.2. Lai tạo cá thể: 4](#_Toc116195661)

[2.1.3. Đột biến: 4](#_Toc116195662)

[2.1.4. Xây dựng thế hệ mới: 4](#_Toc116195663)

[2.1.5. Cách khởi tạo quần thể ban đầu: 5](#_Toc116195664)

[2.2. Thuật toán Split 5](#_Toc116195665)

[2.3. Tối ưu lợi nhuận trên NST 6](#_Toc116195666)

[2.3.1. Tối ưu bằng cách lấp đầy trọng tải 6](#_Toc116195667)

[2.3.2. Tối ưu bằng cách tìm kiếm cục bộ(local search) 6](#_Toc116195668)

[3. Kết quả thực nghiệm 7](#_Toc116195669)

[4. Đánh giá và hưởng cải tiến 8](#_Toc116195670)

[4.1. Đánh giá 8](#_Toc116195671)

[4.2. Đề xuất cải tiến: 8](#_Toc116195672)

[5. Tổng kết 8](#_Toc116195673)

[Tài liệu tham khảo: 8](#_Toc116195674)

# Giới thiệu

Logistics là dịch vụ cung cấp, vận chuyển hàng hóa tối ưu nhất từ nơi sản xuất đến tay người tiêu dùng. Công việc của các công ty Logistics là lên kế hoạch cụ thể, kiểm soát sự di chuyển của hàng hóa hay thông tin về nguyên liệu từ điểm xuất phát đến điểm tiêu thụ theo yêu cầu khách hàng đặt ra. Để cạnh tranh hiệu quả trong ngành này, các công ty phải luôn cải tiến và chú trọng đến yếu tố số lượng, chất lượng, thời gian và giá cả dịch vụ. Sau đây là tổng quan về bài toán logistics:

**Bài toán lập lịch trình cho các xe tải và drone:**

* Có một kho hàng ở trung tâm, toạ độ (0, 0)
* Một tập toạ độ của n khách hàng, mỗi khách hàng i có yêu cầu lượng hàng thuộc [Li, Ui] và lợi nhuận mỗi đơn vị hàng là Wi
* Có K xe tải, M drone, trọng lượng tối đa lần lượt là Mt, Md. Tốc độ là Vt, Vd
* Thời gian bay tối đa của Drone trong một hành trình là d
* Tổng thời gian làm việc của mỗi phương tiện là D
* Mỗi xe tải thực hiện tối đa 1 hành trình duy nhất, thăm mỗi khách tối đa 1 lần
* Mỗi Drone có thể có nhiều hành trình, mỗi hành trình thăm mỗi khách hàng tối đa 1 lần

⇒ Cần tìm hành trình của xe tải và Drone, lượng hàng giao cho mỗi khách hàng của 2 loại phương tiện để tối đa hoá lợi nhuận

# Thuật toán

Hướng đi của thuật toán:

* Tìm 1 lời giải thông qua thuật toán tham lam
* Tạo quần thể đầu tiên để tiến hành giải thuật di truyền
* Áp dụng giải thuật di truyền và thuật toán split
* Đánh giá quần thể và đưa ra kết quả tối ưu đã đạt được

Thuật toán chính của bài toán là giải thuật di truyền(GA) và kết hợp với thuật toán Split

## 2.1. Giải thuật di truyền

### 2.1.1. Biểu diễn nhiễm sắc thể:

Nhiễm sắc thể (NST) để biểu diễn lời giải gồm một chuỗi các cặp

[tên khách hàng, số lượng hàng]

Ví dụ:

[[3, 500], [1, 50], [2, 500], [4, 25], [6, 125], [6, 40], [6, 35], [5, 35], [4, 25], [5, 15]]

[[6, 500], [4, 250], [10, 100], [9, 500], [1, 500], [7, 200], [8, 100], [2, 100], [3, 500], [5, 25]]

Các nhiễm sắc thế có thể chứa lặp lại các khách hàng, độ dài có thể thay đổi. Những nhiễm sắc thể này chưa phải là lời giải cuối của bài toán. Thông qua giải thuật di truyền để biến đổi các NST và mỗi NST thông qua thuật toán Split.

### 2.1.2. Lai tạo cá thể:

Chọn bố mẹ lai tạo:

* Chọn cha mẹ 1 nằm trong 25% cá thể tốt nhất của thế hệ trước
* Chọn cha mẹ 2 ngẫu nhiên trong cá thể trước(tỉ lệ 90%) và có tỉ lệ 10% chọn cá thể ở trong các cá thể infeasible

Thuật toán sử dụng lai chéo bình thường, chọn một vị trí nằm giữa 2 cá thể bố mẹ và tiến hành lai chéo

Ví dụ:

Bố mẹ 1: [[4, 50], [1, 200], [3, 50], [2, 500], [3, 50], [1, 50]]

Bố mẹ 2: [[2, 500], [1, 50], [4, 50], [1, 200], [2, 50], [3, 50]]

Chọn ví trí là 4:

Con 1: [[4, 50], [1, 200], [3, 50], [2, 500], [2, 50], [3, 50]]

Con 2: [[2, 500], [1, 50], [4, 50], [1, 200], [3, 50], [1, 50]]

### 2.1.3. Đột biến:

Đột biến trong bài toán chia làm 3 loại:

* Đột biến Gen:

Chọn 2 chuỗi con trong nhiễm sắc thể và tiến hành đổi chỗ:

NST cũ: [[4, 50], [1, 200], [3, 50], [2, 500], [3, 50], [1, 50]]

NST mới: [[2, 500], [3, 50], [3, 50], [4, 50], [1, 200], [1, 50]]

* Đột biến Weight:

Đột biến số lượng hàng giao cho mỗi khách hàng:

NST cũ: [[4, 50], [1, 200], [3, 50], [2, 500], [3, 50], [1, 50]]

NST mới: [[4, 50], [1, 200], [3, 75], [2, 500], [3, 50], [1, 50]]

* Đột biến Length:

Đột biến về độ dài của NST, chia khách hàng phân phối 2 lần

NST cũ: [[4, 50], [1, 200], [3, 50], [2, 500], [3, 50], [1, 50]]

NST mới: [[4, 50], [1, 200], [2, 300], [3, 50], [2, 200], [3, 50], [1, 50]]

Để đảm bảo độ dài của NST không xảy ra hiện tượng chỉ tăng lên về độ dài thì trong thuật toán Split có gộp lại các khách hàng giống ở trong một chu trình.

### 2.1.4. Xây dựng thế hệ mới:

Một thế hệ bao gồm 1 quần thể feasible(1000 cá thể) và 1 quần thể infeasible(200 cá thể)

Các cá thể trong quần thể feasible không cho phép được lặp lại.

Việc xây dựng thế hệ mới sẽ dựa trên tỉ lệ như sau:

- Giữ lại 25% cá thể ưu tú nhất của đợt trước

- Trong việc lựa chọn cặp cha mẹ đã chọn ra 250 cặp cha mẹ tương ứng với đó chúng ta nên sinh ra tỉ lệ các con là 1:3. Từ đó với mỗi cặp cha mẹ chúng ta tiến hành như sau:

- Lấy cha mẹ lai tạo chéo cho 2 con

- Lấy cha mẹ và 2 con đột biến lần 1 sinh ra 8 cá thể

- Tiếp tục đột biến 8 cá thể thành 16 cá thể

- Bỏ đi cha và mẹ ban đầu còn 14 cá thể

- Trong 14 cá thể này sẽ chọn ra 3 cá thể feasible tốt nhất và chọn ra 1 cá thể infeasible

- Trong quần thể feasible cá cá thể sẽ được sắp xếp theo lợi nhuận thu được, còn quần thể infeasible việc chọn lọc và đào thải sẽ ngẫu nhiên.

### 2.1.5. Cách khởi tạo quần thể ban đầu:

Sử dụng một thuật toán tham lam để tạo 1 lời giải ban đầu:

- Tạo lịch trình cho xe tải với lượng hàng tối thiểu dựa trên thuật toán tham lam (đi thăm các khách hàng gần nhất trước)

- Tạo lịch trình cho xe drone phân chia nốt cho cách khách hàng cho đủ lượng hàng tối thiểu

- Sau khi phân chia đủ lượng hàng tối thiểu thì tiếp tục phân chia hàng hóa cho cách khách hàng dựa trên thuật toán tham lam (theo chỉ số lợi nhuận / khoảng cách) để tối ưu lợi nhuận giao hàng mà vẫn giao được nhiều số lượng hàng.

- Tổng hợp các lịch trình trên ta thu được một lời giải tham lam

Sử dụng đột biến để tạo ra cách cá thể mới từ một cá thể khởi tạo ban đầu.

- Tạo ra một cá thể với NST ban đầu là chuỗi các khách hàng với số lượng hàng tối thiểu.

- NST khởi tạo ban đầu và lời giải tham lam thu được làm cá thể đầu tiên sau đó liên tục đột biến và lặp lại cho đến khi thu được đủ số lượng của quần thể feasible.

## 2.2. Thuật toán Split

Như đã trình bày ở phần trước khi chúng ta trả qua các cá thể mới thu được các NST là một chuỗi các khách hàng nhưng vẫn chưa phải là 1 lời giải của bài toán. Thuật toán Split này sẽ chia các NST đó thành một lời giải tối ưu nhất từ chính NST này.

NST:

[[2, 100], [5, 25], [6, 500], [1, 500], [8, 100], [3, 205], [4, 250], [3, 295], [7, 200], [9, 100]]

NST qua thuật toán Split (mặc định các chu trình đầu là chu trình của xe tải)

[[[2, 100], [5, 25], [6, 500], [1, 500], [8, 100], [3, 205]], [[4, 250]], [[3, 295]], [[7, 200], [9, 100]]]

**Bước 1:** Tạo ma trận các cạnh

Gọi ngắn gọn các khách hàng theo các NST là các khách hàng theo thứ tự trong NST luôn (Ví dụ khách hàng 1 là [2,100], khách hàng 2 là [5,25])

Tạo 2 ma trận(của xe tải và drone) các cạnh i-j là khoảng thời gian, số lượng hàng giao đi từ kho 0 tới khách hàng i+1 tới khách hàng i+2,… cho tới khách hàng j và quay lại kho 0 (không được quá thời gian và tải trọng)

time[i][j] = time[i][j-1] - time[0][j-1] + time[j-1][j] + time[j][0]

**Bước2:** Chạy thuật toán Split sẽ chọn ra các đường đi ngắn nhất lần lượt tới các đỉnh(lưu nhớ đỉnh trước đó), kết thúc khi đã đi đến được đỉnh cuối cùng

Ví dụ

NST ví dụ bên trên thì thuật toán sẽ đưa ra quãng đường đi là: [0,6,7,8,10]

Tương đương với các chu trình: [0-1-2-3-4-5-6-0] + [0-7-0] + [0-8-0] + [0-9-10-0]

Dựa vào số lượng xe tải là 1 thì lời giải là xe tải là chu trình đầu, 3 chu trình sau là của drone. Từ đây thay thế lại các cặp khách hàng thật và số lượng hàng để đưa ra lời giải cuối cùng.

## 2.3. Tối ưu lợi nhuận trên NST

### 2.3.1. Tối ưu bằng cách lấp đầy trọng tải

Khi sử dụng thuật toán split chia các hành trình từ drone thì kết hợp với việc tìm các hành trình vẫn còn chưa đầy trọng tải và cho thêm hàng hóa dựa theo lợi nhuận trên mỗi hàng hóa

### 2.3.2. Tối ưu bằng cách tìm kiếm cục bộ(local search)

Khi thu được thế hệ mới sau khi lai ghép và đột biến thì chọn ra một số lời giải tốt nhất để tìm kiếm cục bộ

Tìm kiếm cục bộ sẽ chuyển lượng hàng từ hàng hóa của cùng 1 khách hàng từ hành trình này qua hành trình khác, sau đó sử dụng thuật toán tối ưu bên trên để tìm kiếm lời giải tối ưu hơn lời giải tốt nhất hiện tại.

# Kết quả thực nghiệm

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tên bộ test | Lời giải thế hệ đầu | Lời giải tốt nhất  ở thế hệ gần nhất | Lợi nhuận tối đa | Thời gian giải(phút) | Thời gian 1 thres(s) | Độ tiến hóa của thuật toán |
| 6.20.1 | 19000 | Thế hệ 0:19000 | 19000 | 0.00005 | None | None |
| 10.20.1 | 37250 | Thế hệ 0:37250 | 37250 | 0.00006 | None | None |
| 20.20.1 | 72750 | Thế hệ 0:72750 | 72750 | 0.00022 | None | None |
| 50.10.1 | 114375 | Thế hệ 0:114375 | 114375 | 0.00062 | None | None |
| 50.10.2 | 99000 | Thế hệ 0:99000 | 99000 | 0.00066 | None | None |
| 50.10.3 | 132125 | Thế hệ 0:132125 | 132125 | 0.00055 | None | None |
| 50.10.4 | 123125 | Thế hệ 0:123125 | 123125 | 0.00047 | None | None |
| 50.20.1 | 104625 | Thế hệ 0:104625 | 104625 | 0.00055 | None | None |
| 50.20.2 | 134500 | Thế hệ 0:134500 | 134500 | 0.00059 | None | None |
| 50.20.3 | 86250 | Thế hệ 0:86250 | 86250 | 0.00058 | None | None |
| 50.20.4 | 121125 | Thế hệ 0:121125 | 121125 | 0.00078 | None | None |
| 50.30.1 | 115280 | Thế hệ 95:124205 | 142750 | 105.34 | 65 giây | 8925/142750=6.3% |
| 50.30.2 | 144500 | Thế hệ 0:144500 | 144500 | 0.00091 | None | None |
| 50.30.3 | 119125 | Thế hệ 0:119125 | 119125 | 0.00046 | None | None |
| 50.30.4 | 155125 | Thế hệ 0:155125 | 155125 | 0.00083 | None | None |
| 50.40.1 | 91040 | Thế hệ 93: 96750 | 98250 | 65.13 | 42 giây | 5710/98250=5.8% |
| 50.40.2 | 117500 | Thế hệ 99:120450 | 122250 | 129.16 | 77 giây | 2950/122250=2.4% |
| 50.40.3 | 96825 | Thể hệ 4: 100400 | 101375 | 9.53 | 114 giây | 3575/101375=3.5% |
| 50.40.4 | 78000 | Thế hệ 0: 78000 | 78000 | 0.00048 | None | None |

Nhận xét:

* Các bộ test đều cho ra lời giải (bộ 50.40.2 với time\_work = 70)
* Các bộ test nhỏ chỉ cần dùng thuật toán tham lam đã cho ra kết quả tối ưu vì tham số thời gian làm việc lớn và số lượng khách hàng còn ít
* Các bộ test lớn có đa số tìm được phương án tốt chấp nhận được:
  + Bộ test 50.40.3 bị dừng lại ở thế hệ 4 vì trải qua hơn 30 thế hệ không tăng lên được hàm giá trị lợi nhuận
  + Các bộ test 50.30.1, 50.40.1, 50.40.2 chỉ cần trải qua khoảng 10-20 thế hệ là có thể đặt tới kết quả tốt gần bằng với kết quả cuối cùng, từ các thế hệ > 20 lợi nhuận có tăng lên nhưng phải qua nhiều thế hệ (5-10) mới tăng được một lượng nhỏ lợi nhuận

# Đánh giá và hưởng cải tiến

## 4.1. Đánh giá

Giải thuật di truyền kết hợp với Split đã cơ bản giải được bài toán vận tải này. Trong trường hợp với số lượng khách hàng nhỏ (số lượng < 20) thì thuật toán có thể dễ dàng đi tới lời giải tốt(thậm chí là tối ưu) nhưng khi số lượng khách hàng tăng lên thì thuật toán rất nhiều thời gian và phải qua nhiều thế hệ mới đưa được lời giải tốt hoặc bị vướng vào một lời giải tối ưu cục bộ.

## 4.2. Đề xuất cải tiến:

- Xây dựng được lời giải tham lam tốt hơn giúp đi tham cách thành số ở xa hơn khi dùng xe tải

- Để hạn chế quần thể bị vướng ở điểm tối ưu cục bộ thì vần phải loại bỏ được các lời giải họ hàng với nhau => Thuật toán trên mới chỉ không cho phép cách NST giống nhau => Đề xuất loại bỏ cách lời giải họ hàng (Ví dụ: loại bỏ các lời giải họ hàng có các hành trình xe tải giống nhau)

- Đánh giá chỉ số fitness của NST theo nhiều chỉ số: lợi nhuận, lượng hàng giao được ở xa bởi xe tải => tối ưu được việc drone giao ở gần,…

- Đưa được các thuật toán tìm lời giải lân cận tốt hơn và nhanh hơn (local search)

# Tổng kết

Bài toán vận tải là một toán toán khó và hiện vẫn chưa có được lời giải với thời gian đa thức nên việc sử dụng các thuật toán tham lam, tìm kiếm cục bộ, heuristic, meta-heuristic được áp dụng phổ biến. Bài làm của em đã có thể tìm ra 1 lời giải cho bài toán nhưng vẫn còn nhiều điểm thiếu sót bên cạnh. Em cảm ơn cô Nguyễn Khánh Phương đã hỗ trợ em trong quá trình giải bài toán. Mô hình của em vẫn còn nhiều hạn chế và thiếu sót, em rất mong nhận được sự góp ý từ cô để trở nên hoàn thiện hơn.

### Tài liệu tham khảo:

* Order-first split-second methods for vehicle routing problems: A review - Christian Prins, Philippe Lacomme, Caroline Prodhon
* A simple and effective evolutionary algorithm for thevehicle routing problem - Christian Prins -LOSI, University of Technology of Troyes, BP 2060, 10010 Troyes Cedex, France
* An Effective Memetic Algorithm with Population Management for the Split Delivery Vehicle Routing Problem - Mourad Boudia, Christian Prins, and Mohamed Reghioui
* https://www.geeksforgeeks.org/traveling-salesman-problem-using-genetic-algorithm/
* https://github.com/polatbilek/Tabu-search-on-Travelling-Salesman-Problem