

BỘ GIÁO DỤC VÀ ĐÀO TẠO
ĐẠI HỌC KINH TẾ TP HỒ CHÍ MINH
TRƯỜNG CÔNG NGHỆ VÀ THIẾT KẾ



ĐỒ ÁN MÔN HỌC

ĐỀ TÀI:

GIẢI BÀI TOÁN KNAPSACK SỬ DỤNG
GIẢI THUẬT DI TRUYỀN

Học phần: Trí Tuệ Nhân Tạo

Nhóm Sinh Viên:

1. NGUYỄN NHẬT QUANG
2. TRẦN HOÀNG TRUNG ĐỨC
3. LÊ MINH TRIỀU
4. TRỊNH THỊ HẢI ÂU
5. NGÔ GIA BẢO

Chuyên Ngành: KHOA HỌC DỮ LIỆU

Khóa: K47

Giảng Viên: TS. Đặng Ngọc Hoàng Thành

TP. Hồ Chí Minh, Ngày 26 tháng 05 năm 2023

MỤC LỤC

MỤC LỤC	1
CHƯƠNG 1. TỔNG QUAN	5
1.1. Giới Thiệu Về Bài Toán Knapsack	5
1.2. Phát Biểu Bài Toán	6
1.3. Một Số Hướng Tiếp Cận Giải Quyết Bài Toán	8
CHƯƠNG 2. GIẢI THUẬT DI TRUYỀN	10
2.1. Giới Thiệu Về Giải Thuật Di Truyền.....	10
1.Chọn lọc tự nhiên:	10
2.Giải thuật di truyền.	12
3.Điểm tối ưu	17
2.2. Ứng Dụng Giải Thuật Di Truyền Cho Bài Toán Knapsack.....	18
1. Phát biểu bài toán:	18
2. Khởi tạo quần thể.....	20
3. Bước chọn (tournament selection).....	21
4. Lai tạo (One-point crossover).....	22
5. Đột biến (bit-flip mutation).....	22
6. Tiêu chí dừng	23
CHƯƠNG 3. CÁC KẾT QUẢ THỰC NGHIỆM.....	24
3.1. Các Tình Huống	24
1. Ảnh hưởng của các siêu tham số.....	24
2. Ảnh hưởng của các toán tử di truyền.	25
3. Bảo tồn trạng thái tốt.	26
3. Dữ liệu từ bài toán.	26
3.2. Phân Tích và Đánh Giá	Error! Bookmark not defined.
1. Vấn đề về các siêu tham số:	28
2. Các toán tử di truyền.....	35
3. Bảo toàn tính trạng tốt của quần thể.	42
4. Ảnh hưởng của dữ liệu.....	45
CHƯƠNG 4. KẾT LUẬN	50
4.1. Các Kết Quả Đạt Được:	50

4.2. Những Hạn Chế và Hướng Phát Triển:.....	52
4.3 Các ứng dụng khác trong thực tế:	53
TÀI LIỆU THAM KHẢO.....	55
PHỤ LỤC	58

BẢNG BIỂU

Hình 1 Tobias Dantzig	5
Hình 2 Bài toán knapsack	6
Hình 3 Vector x trong giải thuật di truyền.	7
Hình 4 Giải thuật tiến hóa	10
Hình 5 Quá trình chọn lọc tự nhiên	10
Hình 6 Charles Darwin.....	11
Hình 7 Quá trình thực hiện giải thuật di truyền	14
Hình 8 Các toán tử di truyền kết hợp với nhau	15
Hình 9 Pseudocode của giải thuật di truyền	17
Hình 10 Điểm tối ưu.....	17
Hình 11 Các vật phẩm.....	18
Hình 12 Minh họa cho giá trị hàm fitness dựa trên tổ hợp các vật phẩm thể hiện dưới dạng vector nhị phân.....	19
Hình 13 Một trường hợp khác với giá trị fitness bằng 0.....	20
Hình 14 Quần thể khởi tạo (nhiễm sắc thể bên trái và giá trị fitness bên phải)	20
Hình 15 Đấu trường của 2 cá thể với độ phù hợp được thể hiện trên yên ngựa.	21
Hình 16 Cá thể với độ phù hợp cao hơn sẽ được chọn cho việc tái sản xuất.....	21
Hình 17 Làm tương tự với các vòng đấu khác để chọn ra các cá thể phù hợp cho việc tái sản xuất.	22
Hình 18 Sử dụng lai tạo 1 điểm từ bước 3.	22
Hình 19 Nhiễm sắc thể hậu biến đổi gen	23
Hình 20 Benchmark.....	27
Hình 21 Sử dụng các tham số khuyến nghị.....	27
Hình 22 Hành vi của cá thể phù hợp nhất trong mỗi thế hệ trong giải thuật di truyền với tham số mặc định.....	27
Hình 23 Hành vi của độ phù hợp toàn quần thể trong mỗi thế hệ trong giải thuật di truyền với tham số mặc định.....	28
Hình 24 Hàm cho thấy tốc độ hội tụ tới một kết quả chỉ định là nhanh hay chậm.	28
Hình 25 Khi kích thước quần thể là 1000	29
Hình 26 Trung bình độ phù hợp khi kích thước quần thể là 1000	29
Hình 27 Tốc độ hội tụ của thuật toán trên một không gian rộng hơn.	29
Hình 28 Khi tăng số lượng thế hệ lên 1000.....	30
Hình 29 Hành vi thuật toán khi nhiều thế hệ hơn	31
Hình 30 Tốc độ hội tụ của thuật toán khi nhiều thế hệ hơn.	31
Hình 31 Khi thay đổi tỉ lệ lai tạo lên cao	32
Hình 32 Trung bình độ phù hợp quần thể khi tăng tỉ lệ lai tạo lên cao.....	32
Hình 33 Tốc độ đạt tới kết quả tốt khi tỉ lệ lai tạo tăng lên.....	32
Hình 34 Khi thay đổi tỉ lệ đột biến lên cao.....	33
Hình 35 Hành vi của độ phù hợp trung bình quần thể khi tỉ lệ lai tạo cao	34
Hình 36 Khi tỉ lệ lai tạo quá cao, giải thuật di truyền không đạt được tới kết quả tốt.....	34

Hình 37 Bảng tổng kết cho sự thay đổi các siêu tham số.....	35
Hình 38 Sử dụng toán tử chọn lọc mới: vòng xe	36
Hình 39 Độ phù hợp trung bình của quần thể khi sử dụng toán tử chọn lọc mới: vòng xe	36
Hình 40 Khi sử dụng toán tử chọn lọc xếp hạng.....	37
Hình 41 Trung bình độ phù hợp của toàn bộ quần thể khi sử dụng toán tử chọn lọc xếp hạng	37
Hình 42 Tổng kết thí nghiệm 2.a.....	38
Hình 43 Sử dụng toán tử lai tạo 2 điểm	39
Hình 44 Trung bình độ phù hợp của quần thể khi sử dụng lai tạo 2 điểm.	39
Hình 45 Tốc độ đạt tới điểm mong muốn của giải thuật khi sử dụng lai tạo 2 điểm.....	39
Hình 46 Tổng kết thí nghiệm 2.b	40
Hình 47 Sử dụng toán tử đột biến kiểu scramble	40
Hình 48 Trung bình độ phù hợp khi sử dụng đột biến kiểu scramble.....	41
Hình 49 Tốc độ đạt tới độ phù hợp tốt khi sử dụng đột biến kiểu scramble.....	41
Hình 50 Tổng kết thí nghiệm 2.c.....	41
Hình 51 Sử dụng hướng tiếp cận elitism để hạn chế sự thoái hóa.	42
Hình 52 Độ phù hợp trung bình qua các thế hệ khi sử dụng hướng tiếp cận elitism.....	43
Hình 53 Tốc độ đạt tới một kết quả tốt của giải thuật sử dụng hướng tiếp cận elitism	43
Hình 54 Áp dụng tiêu chí dừng delta cho giải thuật di truyền	44
Hình 55 Độ phù hợp trung bình khi sử dụng hướng tiếp cận delta.....	44
Hình 56 Khi sử dụng hướng tiếp cận delta có khả năng khiến cho kết quả không đạt được như mong muốn.....	44
Hình 57 Dữ liệu cho bài toán dễ.....	45
Hình 58 Kết quả chính xác của bài toán dễ.....	45
Hình 59 Kết quả của giải thuật di truyền để giải quyết các vấn đề dễ.	46
Hình 60 Trung bình độ phù hợp của giải thuật di truyền đối với các bài toán dễ.....	46
Hình 61 Dữ liệu của bài toán khó giải quyết.....	47
Hình 62 Kết quả chính xác của bài toán khó giải quyết.....	47
Hình 63 Kết quả của giải thuật di truyền đối với bài toán khó giải quyết.	48
Hình 64 Độ phù hợp trung bình của giải thuật di truyền với bài toán khó giải quyết.	48
Hình 65 Tổng kết thí nghiệm 4.b	49
Hình 66 So sánh độ phức tạp về thời gian	50
Hình 67 So sánh các thuật toán với số lượng vật phẩm khác nhau.....	51
Hình 68 Thời gian thực thi chương trình của từng thuật toán.....	51
Hình 69 So sánh các kết quả của từng thuật toán.....	52

CHƯƠNG 1. TỔNG QUAN

1.1. Giới Thiệu Về Bài Toán Knapsack

Bài toán knapsack là một bài toán thuộc vấn đề tối ưu hỗn hợp: Cho một tập các vật phẩm và mỗi vật phẩm có chứa giá trị và cân nặng riêng, xác định xem vật phẩm nào nên được bỏ vào ba lô sao cho tổng khối lượng của toàn bộ vật phẩm là nhỏ hơn hoặc bằng một giới hạn cho trước (giới hạn này thường là khả năng chứa của cái ba lô(capacity)), sao cho tổng giá trị của các vật phẩm trong cái knapsack phải lớn nhất có thể.

Bài toán có cái tên này là dựa trên một vấn đề: Một người nào đó đang có một cái ba lô có sức chứa cố định và người này phải cho vào ba lô toàn bộ vật phẩm có giá trị nhất (nhưng các vật phẩm này vẫn nằm trong sức chứa của cái ba lô). Việc phân bổ sao cho hợp lý cũng đại diện cho các vấn đề thường phát sinh trong việc phân bổ nguồn lực mà ở đó người ra quyết định phải lựa chọn từ một tập hợp các dự án hoặc nhiệm vụ dưới một mức chi phí cố định, hoặc là ràng buộc về thời gian (ứng dụng trong kinh doanh).

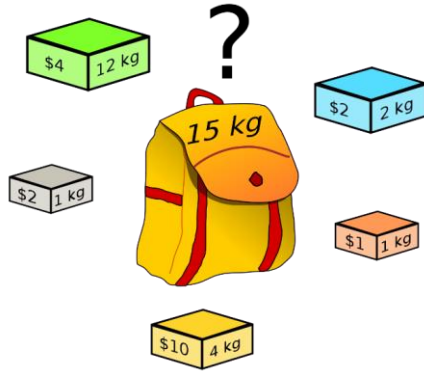
Về lịch sử hình thành, bài toán này đã được nghiên cứu hơn 1 thế kỷ với những bài nghiên cứu đầu tiên từ năm 1897 trong một tạp chí về toán (Mathews, 1896). Và bài toán này bắt nguồn từ những công trình đầu tiên của nhà toán học Tobias Dantzig (Dantzig)



Hình 1 Tobias Dantzig

1.2. Phát Biểu Bài Toán

Như đã đề cập ở trên, Knapsack là một vấn đề quen thuộc trong lĩnh vực tối ưu hóa, đặc biệt là trong lĩnh vực tối ưu hóa tổ hợp. Vấn đề này đặt ra câu hỏi về cách xếp các món đồ vào một cái balô với khối lượng tối đa đã cho trước, sao cho tổng giá trị của các món đồ này là lớn nhất có thể.



Hình 2 Bài toán knapsack

Vấn đề Knapsack là một trong những vấn đề tối ưu hóa tổ hợp quan trọng nhất trong lĩnh vực khoa học máy tính. Hướng tiếp cận để giải bài toán knapsack cũng có thể được sử dụng để giải quyết một số bài toán thực tiễn như:

- Đóng hàng: Khi đóng hàng, chúng ta phải quyết định xem vật phẩm nào cần được mang và bao nhiêu vật phẩm được mang. Mục tiêu là để tối đa hóa số lượng vật phẩm hữu dụng, đồng thời cũng tối thiểu hóa cân nặng của balo.
- Lựa chọn cổ phiếu: Khi lựa chọn cổ phiếu để đầu tư, chúng ta cần phải quyết định xem cổ phiếu nào nên mua và mua bao nhiêu. Mục đích là để tối ưu hóa doanh thu mà chúng ta đạt được trong khi tối thiểu rủi ro.
- Xếp lịch: khi thực hiện nhiệm vụ xếp lịch, chúng ta phải quyết định xem công việc nào nên làm và khi nào làm công việc đó. Mục đích là để tối thiểu thời gian cần phải bỏ ra để hoàn thành tất cả công việc trong khi tối ưu lợi nhuận mà ta làm ra.

Bài toán knapsack là một bài toán NP-problem* tức Nondeterministic Polynomial time. Điều này đã được chứng minh trong một tạp chí về vấn đề knapsack (Kellerer, H., Pferschy, U., Pisinger, D., 2004) và trong bài giảng của đại học Cornell (P.W, 2014) cũng chứng minh rằng bài toán knapsack không phải là NP-complete nếu đầu vào của nó là đơn nguyên (unary), nhưng sẽ là NP-complete nếu như đầu vào của nó là nhị phân (binary).

* NP-problem: một bài toán quyết định (decision problem) được gọi là thuộc lớp NP nếu tồn tại một bằng chứng có thể được kiểm tra bởi một máy Turing trong thời gian đa thức.

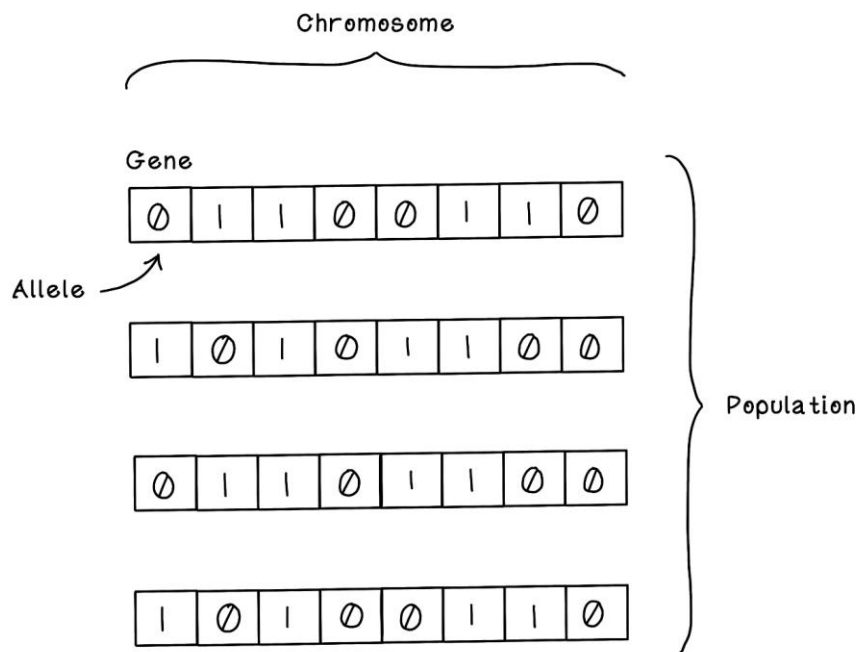
Bài toán knapsack phổ biến nhất là bài toán knapsack dạng 0-1, bài toán dạng này thường sử dụng một vector trong đó hạn chế số x_i bản sao của mỗi loại mục là 0 hoặc 1. Cho một tập gồm n vật phẩm và được đánh số từ 1 đến n , mỗi vật phẩm thứ i đều sẽ có một cân nặng w_i và giá trị v_i tương ứng, theo sau đó là một giá trị giới hạn W của cái cặp, lúc này mục tiêu của bài toán tối ưu là tối đa giá trị:

$$\sum_{i=1}^n v_i x_i$$

Và đồng thời phải đáp ứng điều kiện:

$$\sum_{i=1}^n w_i x_i \leq W \text{ và } x_i \in \{0,1\}$$

Với \mathbf{x} là một vector nhị phân, mỗi phần tử 0 đại diện cho việc phần tử không được chọn và phần tử 1 đại diện cho việc phần tử được chọn để bỏ vào balo. Một vector \mathbf{x} được sử dụng trong giải thuật di truyền có cấu tạo như sau:



Hình 3 Vector \mathbf{x} trong giải thuật di truyền.

Một vài biến thể nổi tiếng của bài toán knapsack hầu hết xuất hiện là do nó thường gắn với các ví dụ trong thực tiễn và người ta muốn giải quyết các vấn đề này bằng hướng tiếp cận khác nhau, ví dụ như:

- Vấn đề knapsack không ràng buộc (Unbounded knapsack problem): Trong biến thể này, số lượng của mỗi vật phẩm là không bị giới hạn.
- Vấn đề nhiều knapsack (Multiple knapsack problem): Trong biến thể này, chúng ta có nhiều cái knapsacks và mỗi cái thường có các sức chứa khác nhau.
- Vấn đề knapsack tối ưu nhiều mục tiêu (Multi-objective knapsack problem): Trong biến thể này, chúng ta có nhiều mục tiêu cần được tối ưu, chẳng hạn như tối ưu hóa lợi nhuận trong khi tối thiểu hóa cân nặng, đồng thời cũng có các vật phẩm không được xuất hiện đồng thời.
- Vấn đề knapsack phi tuyến tính (Nonlinear knapsack problem): Trong biến thể này, lợi nhuận hoặc là cân nặng của một vật phẩm không tuyến tính trong số lượng của nó.
- Vấn đề knapsack mạnh (Robust knapsack problem): Trong biến thể này, cân nặng của mỗi vật phẩm là không chắc chắn.

1.3. Một Số Hướng Tiếp Cận Giải Quyết Bài Toán

Là một bài toán đã được nghiên cứu suốt hơn 1 thế kỷ, bài toán knapsack được các nhà toán học cũng như các nhà khoa học máy tính nghiên cứu và cho ra nhiều giải pháp khác nhau để có thể giải quyết vấn đề này một cách tốt nhất có thể. Một vài giải pháp phổ biến được mọi người sử dụng có thể kể đến như:

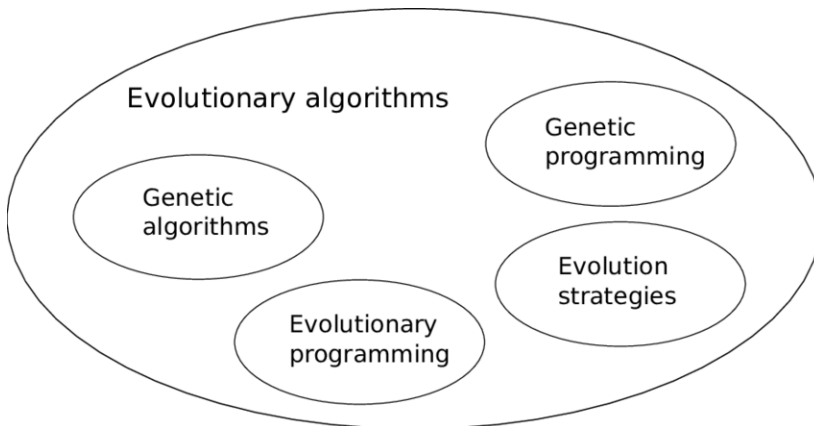
- Brute force (vét cạn): Đây là giải pháp đơn giản nhất, về cơ bản thuật toán này sẽ duyệt qua tất cả tổ hợp vật phẩm có thể có. Và như vậy, với chi phí tính toán cao, thuật toán này là không phù hợp trong hầu hết các tình huống trong thực tế mà chỉ phù hợp với các tập dữ liệu và ở đó có kích cỡ nhỏ.
- Dynamic programming (quy hoạch động): Thuật toán này sử dụng bằng cách chia nhỏ vấn đề đang gặp thành nhiều vấn đề nhỏ hơn. Và rồi kết quả của bài toán lớn lúc này sẽ được gộp các kết quả của bài toán nhỏ lại với nhau. Thuật toán này thì thích hợp hơn thuật toán brute force được đề cập ở trên, hơn nữa nó cũng có thể được sử dụng để giải quyết các bài toán mà có kích cỡ lớn.
- Greedy algorithm (giải thuật tham lam): Giải thuật này hoạt động bằng cách thêm các vật phẩm một cách tuần tự các vật phẩm có giá trị cao nhất vào trong balo, miễn sao cái cặp vẫn chứa được. Tuy phương pháp này không đảm bảo sẽ cho ra một kết quả là tối ưu toàn cục, nhưng sẽ lại rất gần với tối ưu toàn cục.
- Branch and bound (nhánh và cận): Thuật toán này hoạt động bằng cách khám phá đệ quy không gian kết quả. Thuật toán sẽ bắt đầu với một tập kết quả khả thi, và sau đó phân nhánh để tìm khám phá các kết quả khả thi khác (các kết quả khả thi ở đây là tổ hợp các vật phẩm đáp ứng được điều kiện tổng cân nặng nhỏ hơn sức chứa W của balo).

- Minimax: Thuật toán này là một thuật toán heuristics mà nó sẽ tìm ra một đáp án mà ở đó tối thiểu cái “regret” tối đa. “regret” được định nghĩa là sự khác biệt giữa giá trị của tối ưu và kết quả được tìm thấy bởi thuật toán.
- Genetic algortihm (Giải thuật di truyền): Giải thuật di truyền là một dạng tìm kiếm heuristics mà ở đó sao chép quá trình chọn lọc tự nhiên. Nó được sử dụng để tìm kiếm các điểm tối ưu hoặc các điểm gần tối ưu cho nhiều bài toán khác nhau. Bao gồm cả bài toán knapsack.

CHƯƠNG 2. GIẢI THUẬT DI TRUYỀN

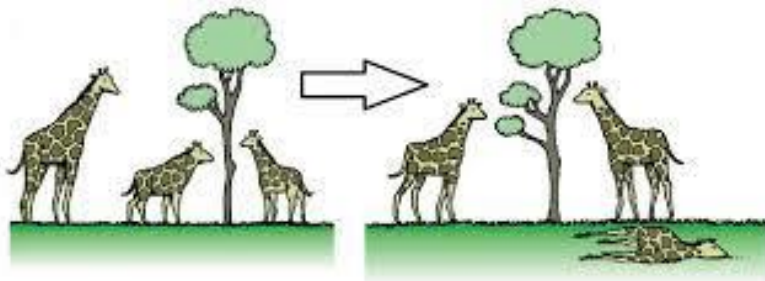
2.1. Giới Thiệu Về Giải Thuật Di Truyền

Như đã đề cập ở trên, giải thuật di truyền (genetic algorithm) là một giải thuật sao chép ý tưởng từ quá trình chọn lọc tự nhiên, bên cạnh đó giải thuật di truyền còn là một phần của giải thuật tiến hóa (evolutionary algorithms).



Hình 4 Giải thuật tiến hóa

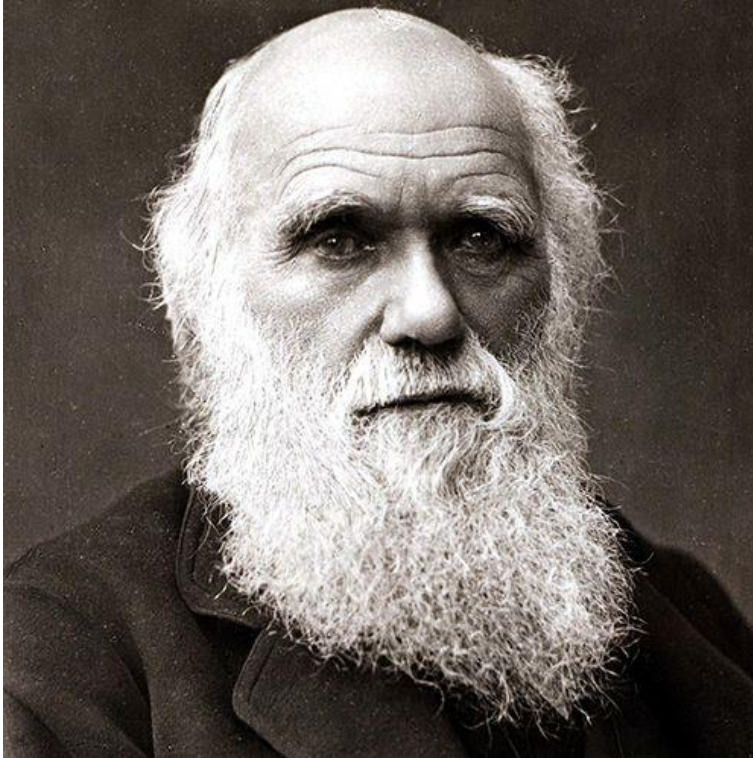
1. Chọn lọc tự nhiên:



Natural Selection in action

Hình 5 Quá trình chọn lọc tự nhiên

Quá trình chọn lọc tự nhiên đã được Charles Darwin giới thiệu và sáng tạo vào thế kỷ 19, sau chuyến du ngoạn 5 năm để học về thực vật, động vật và hóa thạch ở Nam Mỹ. Năm 1859, ông mang ý tưởng về quá trình chọn lọc tự nhiên và có được sự chú ý của công chúng toàn thế giới và trình bày nó trong cuốn “On the original Species” (Darwin, 1859).



Hình 6 Charles Darwin

Quá trình chọn lọc tự nhiên chính là cách hiểu của Darwin về quá trình tiến hóa và được mô tả trong sách “*On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life* (1859)”, và như tên gọi, quá trình tiến hóa qua chọn lọc tự nhiên có thể được hiểu ý tưởng là mỗi cá thể với đặc tính phù hợp với môi trường của chúng thì hầu hết sẽ có khả năng sống sót và tái sản xuất, chuyển tải gens của chúng cho thế hệ kế tiếp. Quá trình này, qua nhiều thế hệ, có thể dẫn tới sự tiến hóa của loài mới.

Dưới đây là giải thích đơn giản về quá trình làm sao mà chọn lọc tự nhiên hoạt động:

- Các biến thể của mỗi loài: Tất cả các loài đều tồn tại biến thể bên trong chính quần thể của chúng. Sự khác biệt này có thể là do sự khác biệt về gen, các yếu tố môi trường, đôi khi là sự kết hợp của cả hai.
- Tính di truyền: Các đặc tính có thể bị truyền từ các thế hệ cha mẹ (parents) sang các thế hệ con (offspring). Điều này cũng có nghĩa là các thế hệ con thường sẽ có các đặc điểm hệt như thế hệ bố mẹ.

- **Sống sót khác biệt và tái sản xuất:** Mỗi cá thể với đặc điểm phù hợp với môi trường nó sống thì thường có khả năng tồn tại và tái sản xuất cao hơn. Điều này cũng mang ý nghĩa các cá thể này sẽ có khả năng truyền tiếp gen của nó lên các thế hệ sau.

Lấy ví dụ như hình ở trên, giả sử có một quần thể hươu cao cổ sống ở cao nguyên. Một vài con có cổ cao hơn những con còn lại. Sự khác biệt này có thể xuất hiện là do khác biệt về gen. Những con cổ cao có khả năng sống sót cao hơn những con cổ không cao do chúng có thể với tới nguồn thực phẩm ở trên cây. Vì lý do này mà các con hươu với cổ cao thường có khả năng sống sót và tái sản xuất cao hơn. Qua thời gian, quần thể hươu cao cổ này sẽ có ngày càng nhiều các con hươu cổ cao. Điều này là do các con cổ thấp thì thường chết đói, do đó không thể truyền gen của nó qua các thế hệ sau.

Quá trình chọn lọc tự nhiên cũng có thể dẫn đến tiến hóa của các loài mới. Điều này xảy ra là do khi một quần thể sinh vật bị cô lập khỏi các quần thể khác và bắt đầu tiến hóa một cách độc lập, khi này chúng sẽ có các môi trường mới, và do đó phải thích nghi và tiến hóa sao cho phù hợp hơn với môi trường sống mới này. Những đặc điểm mới này có thể giúp quần thể tồn tại ở môi trường của chúng, điều này dẫn đến sự hình thành các loài mới.

2. Giải thuật di truyền.

Giải thuật di truyền như được đề cập ở trên là nhằm mục đích sử dụng chọn lọc tự nhiên để tạo ra một thế hệ mới phù hợp với mục tiêu của bài toán cần giải quyết. Ý tưởng về ứng dụng giải thuật di truyền được tạo ra bởi John Holland vào năm 1975 trong cuốn sách “Adaption in Natural and Artificial Systems”. Trong cuốn sách này, Holland định nghĩa giải thuật như là một tìm kiếm heuristic dựa trên cơ chế chọn lọc tự nhiên (Holland, 1975).

Một giai đoạn quan trọng trong giải thuật di truyền đó chính là bước khởi tạo quần thể (quần thể được khởi tạo này chính là quần thể thuộc thế hệ đầu tiên). Một vài nghiên cứu đã chỉ ra rằng nếu quần thể khởi tạo để giải bài toán về giải thuật di truyền là tốt thì kết quả đạt được của thuật toán cũng sẽ tốt (kết quả của thuật toán là một lời giải tốt cho bài toán ta đang giải)(Burke, 2006)(Zitzler, 2000). Và ngược lại, cũng có nghiên cứu chỉ ra nếu quần thể khởi tạo không đủ tốt thì thuật toán sẽ khó tìm ra một kết quả thật sự tốt(Fernando G. Lobo & Cláudio F. Lima, 2005). Và đôi khi, nếu bài toán đang cần được áp dụng GA là một bài toán khó thì ta cũng có thể cung cấp thêm một vài thông tin vào trong quần thể khởi tạo. Bên cạnh đó, đối với quần thể được khởi tạo, quần thể này càng phân bố rộng thì giải thuật di truyền càng dễ tìm được một kết quả tốt (và điều này cũng cần một số lượng cá thể phù hợp), hơn nữa một vài nghiên cứu cũng chỉ ra cách sử dụng sự phân bố này để tránh hội tụ sớm(William Frederick, Robert Sedlmeyer, & Curt White, 1993; Leung Yee, Gao Yong, & Xu Zong-Ben, 1997). Ngoài ra các nghiên cứu khác cũng dựa vào các phương pháp chọn và hàm fitness để cải thiện quần thể khởi tạo(Colin Reeves, 1998).

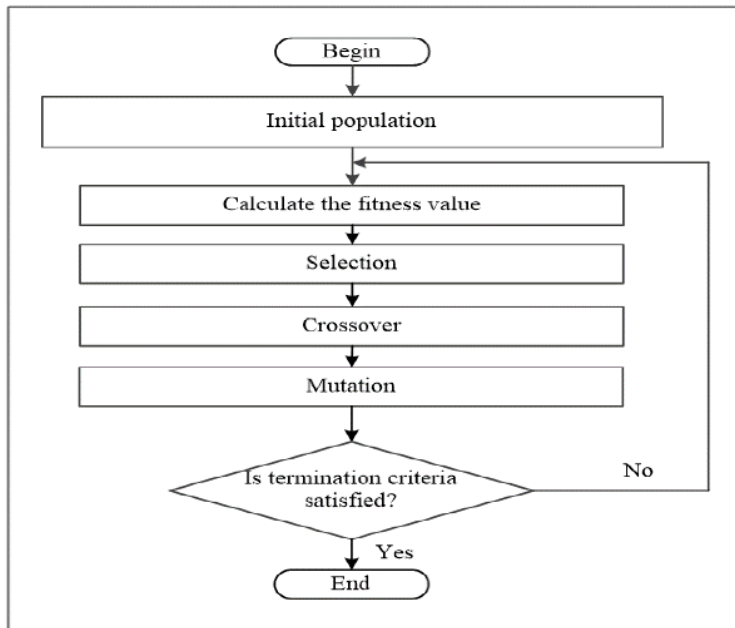
Giải thuật di truyền hoạt động bằng cách duyệt tuần tự qua một quần thể kết quả (population of solutions). Mỗi một kết quả có thể được biểu diễn dưới dạng một nhiễm sắc

thể (chromosome) tức một chuỗi gen. Mỗi gen trong một nhiễm sắc thể có thể được biểu diễn dưới dạng nhị phân hoặc một số thực. Độ phù hợp (fitness) của mỗi nhiễm sắc thể là nhằm đo lường xem nhiễm sắc thể đó tốt thế nào, Độ tốt này được tính bằng cách đánh giá nhiễm sắc thể đó thông qua một hàm số. Đối với mỗi bài toán khác nhau, người thiết kế có thể tạo ra các hàm fitness khác nhau sao cho phù hợp với vấn đề đang giải quyết, từ đó người thiết kế cũng có thể định hướng kết quả của giải thuật di truyền thông qua cách thiết lập hàm fitness này. Do đó có thể thấy rằng hàm fitness là một hàm quan trọng cần được thiết kế cẩn thận để có thể hướng bài toán tới một kết quả mong muốn.

Giải thuật di truyền cần có các phép toán, các phép toán này được gọi là các toán tử di truyền (genetic operator). Các phép toán này chính là cơ chế được sử dụng trong giải thuật di truyền để tạo ra các nhiễm sắc thể mới được điều chỉnh từ các nhiễm sắc thể cũ. Có 3 toán tử chính, tuy nhiên chỉ có 2 trong số đó là có khả năng tạo nên sự thay đổi của các nhiễm sắc thể. Các toán tử đó chính là:

- Đột biến (mutation): Sự đột biến chính là việc thay đổi ngẫu nhiên trong một nhiễm sắc thể. Sự đột biến này có thể nhỏ, như là thay đổi đơn lẻ một bit, hoặc là thay đổi nhiều hơn như là thêm vào hoặc cắt bớt/ thay thế một đoạn gen. Phép đột biến này cho phép giới thiệu các cá thể mới vào trong quần thể. Điều này khiến cho quần thể trở nên đa dạng hơn, giúp cho chúng ta khám phá rộng hơn không gian tìm kiếm, nhờ vậy tăng khả năng tìm kiếm được kết quả tốt hơn.
- Lai tạo (crossover): Lai tạo là toán tử kết hợp các nhiễm sắc thể lại với nhau để tạo nên một nhiễm sắc thể mới. Lai tạo hiểu đơn giản là bằng cách trao đổi các gen giữa các nhiễm sắc thể. Nhiễm sắc thể mới này được kì vọng sẽ bảo tồn các gen tốt từ các nhiễm sắc thể tạo nên nó, hơn nữa nó việc lai tạo này cũng được tạo ra các tổ hợp gen mới có thể dẫn đến kết quả tốt hơn.
- Chọn lọc (selection): Trong 3 toán tử di truyền chính, chỉ có toán tử chọn lọc là không tạo ra sự thay đổi của các nhiễm sắc thể. Đây là quá trình chọn ra các nhiễm sắc thể sẽ được dùng để tạo ra thế hệ kế tiếp. Các phương thức chọn lọc được sử dụng thường xuyên nhất chính là phương thức bánh xe roulette, phương thức đấu trường, ngoài ra còn có nhiều phương thức khác nữa. Nhìn chung quá trình chọn lọc được sử dụng nhằm đảm bảo rằng các nhiễm sắc thể tốt nhất sẽ được truyền sang các thế hệ kế tiếp, điều này giúp thuật toán đạt tới một kết quả tốt.

Các toán tử di truyền được sử dụng liên kết với nhau để hướng dẫn giải thuật tiến tới một kết quả tốt nhất có thể. Đầu tiên, thuật toán sẽ tạo ra một quần thể các nhiễm sắc thể một cách ngẫu nhiên. Các nhiễm sắc thể này sẽ được đánh giá bằng một hàm tính độ phù hợp, và tùy vào phương thức chọn lọc, sau quá trình chọn lọc, ta có các nhiễm sắc thể với độ phù hợp cao nhất được chọn để thực hiện lai tạo và đột biến để cho ra thế hệ kế tiếp. Và chúng ta sẽ làm đi làm lại quá trình này cho tới khi tìm được một kết quả như ý (hoặc là thỏa tiêu chí dừng).



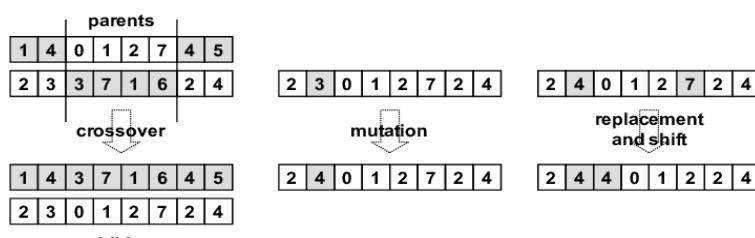
Hình 7 Quá trình thực hiện giải thuật di truyền

Với các ý tưởng chung về toán tử di truyền chung như vậy, có nhiều nghiên cứu khác tìm cách cải thiện các toán tử di truyền đó, cụ thể như sau:

- Đột biến:
 - Đột biến bit (bit-flip mutation): Ngẫu nhiên lật giá trị một bit trong một nhiễm sắc thể. Loại đột biến này thường được sử dụng cho các nhiễm sắc thể được biểu diễn dưới dạng chuỗi nhị phân.
 - Đột biến đảo (inversion mutation): Trong đột biến đảo, một đoạn ngẫu nhiên của nhiễm sắc thể bị đảo ngược lại.
 - V.v...
- Lai tạo:
 - Lai tạo một điểm (one-point crossover): Một điểm chéo được chọn ngẫu nhiên và đoạn gen của một phía tính từ điểm chéo đó sẽ được trao đổi giữa các cá thể phụ huynh.
 - Lai tạo hai điểm (two-point crossover): Giống như lai tạo một điểm, nhưng ở đây tồn tại hai điểm chéo.
 - Lai tạo số học (arithmetic crossover): Trong lai tạo số học, mỗi gen của cá thể con được tính là trung bình trọng số của các gen của các cá thể bố và mẹ.
 - V.v...

- Lựa chọn:
 - Bánh xe roulette (Roulette wheel selection): Trong cách chọn này, mỗi một cá thể trong quần thể được gán một xác suất được chọn, xác suất này tỷ lệ thuận với độ phù hợp của nó.
 - Đấu trường (Tournament selection): Một số lượng nhỏ các cá thể được chọn ngẫu nhiên từ quần thể. Cá thể mang độ phù hợp cao nhất trong toàn bộ đấu trường được chọn.
 - Xếp hạng (Rank selection): Với cách chọn này, các cá thể được xếp hạng dựa trên độ phù hợp. Các cá thể với hạng cao sẽ được chọn.
 - Tinh hoa (Elitist selection): Các cá thể tốt nhất (có điểm phù hợp cao nhất) được chọn để cho vào thế hệ kế tiếp.
 - V.v...

Hình dưới đây thể hiện việc các toán tử chọn lọc và toán tử đột biến tương tác với nhau:



Hình 8 Các toán tử di truyền kết hợp với nhau

Ngoài các toán tử di truyền vừa được đề cập, các tiêu chí dừng cũng rất quan trọng. Việc ta thiết lập một tiêu chí dừng phù hợp nhằm giảm giá trị tính toán và giữ lại các thế hệ tốt nhất, các tiêu chí dừng phổ biến có thể kể đến như:

- Số lượng thế hệ tối đa (Maximum number of generations): giải thuật di truyền sẽ tự động dừng lại sau khi đạt tới số thế hệ cho trước.
- Độ phù hợp tối đa (Maximum fitness value): tương tự như số lượng thế hệ tối đa, nhưng lần này giải thuật sẽ dừng lại khi đạt tới một số điểm phù hợp cho trước.
- Không có sự cải thiện trong độ phù hợp (No improvement in fitness): Giải thuật di truyền sẽ dừng lại khi không có sự thiện nào về độ phù hợp qua các thế hệ.
- Tiêu chí người dùng (User-defined stopping criterion): Giải thuật sẽ tự động dừng thi hành khi người dùng chỉ định một tiêu chí dừng cụ thể, phù hợp với bài toán đang giải quyết.

Tiêu chí dừng là một tiêu chí quan trọng trong thiết kế ra quyết định của giải thuật di truyền, vì vậy ta nên thiết lập tiêu chí dừng sao cho giải thuật di truyền hội tụ đến một kết quả tốt mà không tốn quá nhiều thời gian. Để phục vụ mục tiêu này, ta cần một vài gợi ý để thiết kế tiêu chí dừng hợp lý:

- Tiêu chí dừng phải nhất quán với mục tiêu của bài toán. Ví dụ nếu bài toán của chúng ta là tìm điểm ngắn nhất của hai điểm thì tiêu chí dừng phải dựa trên độ dài của đường đi. Còn trong bài toán knapsack, nếu như chúng ta quan tâm tới giá trị của balo thì tiêu chí dừng cũng phải liên quan tới giá trị của balo.
- Tiêu chí dừng phải thực tế. Lấy ví dụ nếu giải thuật di truyền của chúng ta được kỳ vọng sẽ tốn một thời gian dài để hội tụ, thì tiêu chí dừng không nên được đáp ứng quá sớm.
- Tiêu chí dừng phải linh hoạt (có thể thay đổi tiêu chí dừng nếu như giải thuật di truyền của chúng ta không được hội tụ giống như kỳ vọng của ta).

Bên cạnh những yếu tố tác động đến giải thuật di truyền được đề cập ở trên, việc chúng ta xác định các siêu tham số (hyperparameter) cũng rất quan trọng để kết quả đạt đến điểm tối ưu toàn cục hoặc các điểm gần điểm tối ưu toàn cục. Nếu xét các tổ hợp cơ bản nhất của giải thuật di truyền thì sau đây là các tham số/phương thức chính mà chúng ta chú ý tới:

- Kích thước quần thể (population size): Số lượng cá thể (nhiễm sắc thể) ở trong quần thể.
- Số lượng thế hệ (generations): Số lượng thế hệ mà thuật toán sẽ tạo ra (trong trường hợp không bị các tiêu chí dừng làm dừng lại trước đó).
- Tỷ lệ lai tạo (mutation rate): Xác suất mà một cá thể sẽ được đột biến.
- Phương thức chọn (selection method): Các cách thức được đề cập ở trên (thường mỗi cách thức sẽ yêu cầu một siêu tham số khác nhau).
- Phương thức lai tạo (crossover method): Các cách thức được đề cập ở trên (thường mỗi cách thức lai tạo khác nhau yêu cầu các siêu tham số khác nhau).
- Phương thức đột biến (mutation method): Các cách thức được đề cập ở trên (thường mỗi cách thức đột biến khác nhau cũng yêu cầu các siêu tham số khác nhau).
- Tiêu chí dừng (Stopping criteria): Như đã đề cập ở trên, tiêu chí dừng cũng quan trọng, đặc biệt là khi chúng ta không muốn tốn quá nhiều tài nguyên tính toán hoặc là chúng ta muốn bảo toàn trạng thái tốt của kết quả.

Bên cạnh các công dụng của việc tìm kiếm bộ siêu tham số phù hợp như đề cập ở trên, trong giải thuật di truyền, việc sử dụng siêu tham số cũng tồn tại một số nhược điểm như tốn thời gian để tìm ra một bộ siêu tham số phù hợp, khó để tìm ra bộ tham số tốt nhất cho toàn bộ bài toán và điều này cũng đồng nghĩa với việc bộ siêu tham số tốt nhất cũng phụ thuộc vào dữ liệu của bài toán (điều này sẽ được bàn luận kỹ hơn ở phần thí nghiệm).

Dưới đây là mô tả toàn bộ quá trình một giải thuật di truyền cơ bản sẽ thực hiện.

Genetic Algorithm Pseudocode

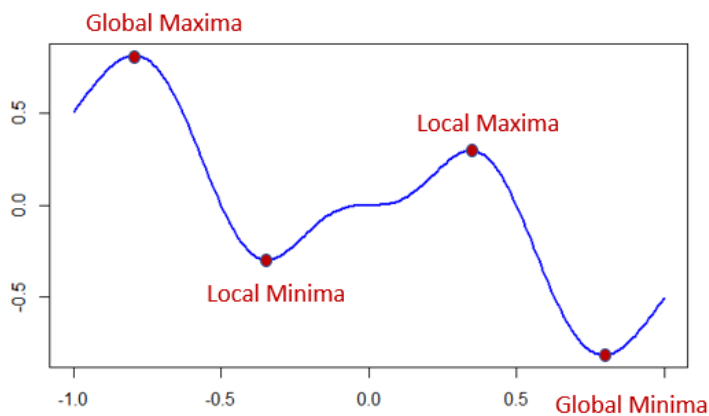
- Let $P(t) = \{x_1^t, x_2^t, \dots, x_n^t\}$ be population at time t
- Procedure GA
 - Begin
 - Set time $t = 0$
 - Initialize $P(t)$
 - While not terminate
 - Begin
 - » Evaluate fitness of each population member $P(t)$
 - » Select members of $P(t)$ based on fitness
 - » Produce offspring of these pairs using genetic operators of crossover and mutation
 - » Replace individuals of $P(t)$ based on fitness with these offspring
 - » Increment time t
 - End
 - End

Hình 9 Pseudocode của giải thuật di truyền

3. Điểm tối ưu

Điểm tối ưu là một thành phần mà ta quan tâm khi thực hiện bất cứ bài toán nào có liên quan đến việc tìm kiếm các kết quả tốt nhất. Khái niệm này tồn tại ở trong nhiều ngành nghề, nhiều môn học khác nhau bởi vì khái niệm tối ưu là một khái niệm rộng và được ứng dụng liên ngành.

Trong bài toán giải thuật di truyền, một giá trị tối ưu là kết quả tốt nhất đáp ứng được yêu cầu bài toán cần được giải quyết, vì vậy nó cũng là kết quả có độ phù hợp cao nhất (nhớ lại độ phù hợp đo lường xem kết quả tốt như thế nào và hàm đo lường này được thiết kế dựa vào vấn đề cần được giải quyết).



Hình 10 Điểm tối ưu

Như hình trên là một đồ thị mô tả các điểm cực đại, cực tiểu trong hàm số. Trong các bài toán về học máy sử dụng cập nhật tham số theo đạo hàm thì không khó để bắt gặp việc thuật toán bị hội tụ ở một điểm tối ưu cục bộ (và không thoát ra được do tại đó thì đạo hàm bằng không). Nhưng điểm mạnh của giải thuật di truyền có thể khắc phục nhược điểm này, các toán tử di truyền cho phép tạo ra sự thay đổi, từ đó thoát khỏi các vùng tối ưu cục bộ. Tuy nhiên đây cũng là hạn chế của giải thuật di truyền, bởi vì giải thuật di truyền áp dụng phương pháp tìm kiếm ngẫu nhiên (stochastic search) nên ở nhiều trường hợp, nếu mục tiêu của chúng ta là để tìm kết quả chính xác cho bài toán thì giải thuật di truyền có thể không phù hợp (do vậy brute force cũng là hướng tiếp cận hợp lý nếu chúng ta thật sự muốn tìm ra kết quả chính xác). Dù vậy, giải thuật di truyền cũng hợp lý nếu chúng ta muốn tìm ra kết quả tốt (xấp xỉ kết quả chính xác) *.

2.2. Ứng Dụng Giải Thuật Di Truyền Cho Bài Toán Knapsack

Giải thuật di truyền là một loại thuật toán tìm kiếm ngẫu nhiên có thể được sử dụng để giải quyết bài toán độ khó NP (như đề cập ở trên, các bài toán độ khó NP thường rất khó được đáp ứng nếu sử dụng các phương pháp truyền thống). Hơn nữa giải thuật di truyền cũng đã được sử dụng để giải quyết rất nhiều bài toán có độ khó NP bao gồm bài toán người đi du lịch, bài toán knapsack, bài toán xếp lịch. Sau đây là mô tả cụ thể cách thức sử dụng giải thuật di truyền cho bài toán knapsack.

Để cho đơn giản và dễ trực quan, nhóm sẽ thực hiện trên một tập nhỏ các vật phẩm và thực hiện trên một nhóm số lượng lớn các vật phẩm ở chương 3.

1. Phát biểu bài toán:

Chúng ta muốn bỏ 4 vật phẩm với các khối lượng và giá trị khác nhau vào trong 1 cái balo có cân nặng nhất định, giá trị và cân nặng của các vật phẩm như hình dưới đây:



Hình 11 Các vật phẩm.

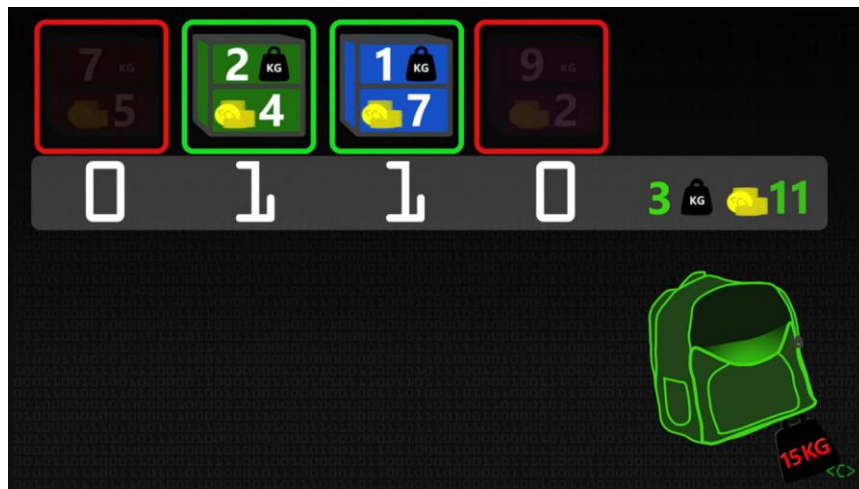
* Nhóm sử dụng từ “chính xác” nếu kết quả cho ra độ phù hợp như kết quả được tạo ra bởi hướng tiếp cận brute force và nhóm sử dụng từ “tốt” nếu kết quả cho ra có độ phù hợp có thể xấp xỉ với độ phù hợp của kết quả có được theo hướng tiếp cận brute force.

Vấn đề của bài toán này là chúng ta muốn cho các vật phẩm vào balo sao cho giá trị của cái balo là cao nhất và bên cạnh đó số cân nặng tổng thể của các vật phẩm có trong balo không được vượt quá 15kg. Và như trong hình trên, các vật phẩm bao gồm 4 vật phẩm với cân nặng và giá trị lần lượt là (7,5); (2,4); (1,7); (9,2).

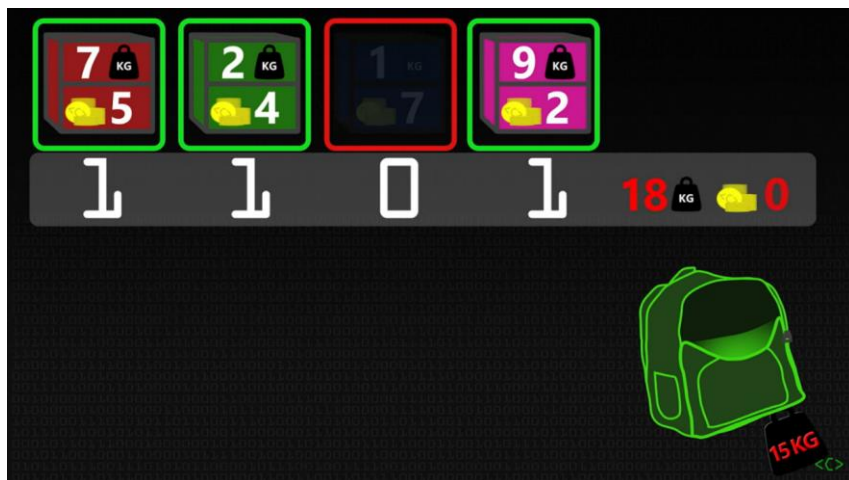
Để phục vụ cho việc áp dụng giải thuật di truyền trong bài toán knapsack, ta cần thể hiện cách chọn các vật phẩm sao cho chương trình máy tính có thể hiểu được. Ta đạt được điều này bằng cách sử dụng mã hóa nhị phân, mỗi gen được biểu diễn bởi một bit và mỗi chromosome (nhiễm sắc thể) là một chuỗi các bit biểu diễn một cách chọn các vật phẩm. Và chúng ta thiết kế hàm fitness chính là tổng giá trị của các vật phẩm được cho vào balo, nếu cân nặng của các món hàng này lớn hơn khả năng chứa của balo (trong trường hợp này là 15kg) thì chúng ta đặt fitness của tổ hợp vật phẩm đó bằng 0.

$$f(x) = \begin{cases} \sum_{i=1}^n x_i \times v_i, & \sum_{i=1}^n x_i \times w_i \leq W \\ 0, & \sum_{i=1}^n x_i \times w_i > W \end{cases}$$

Trong đó: x, v, w là các vector có độ dài bằng nhau và bằng n với n là tổng số vật phẩm (n trong trường hợp bài toán này là 4), và W là khả năng mà cái cặp chứa được (trong trường hợp bài toán này thì $W = 15$). Và với vector x được biểu diễn dưới dạng vector nhị phân, còn 2 vector v, w là 2 vector có độ dài bằng n với các phần tử trong đó chứa các số thực lớn hơn 0.



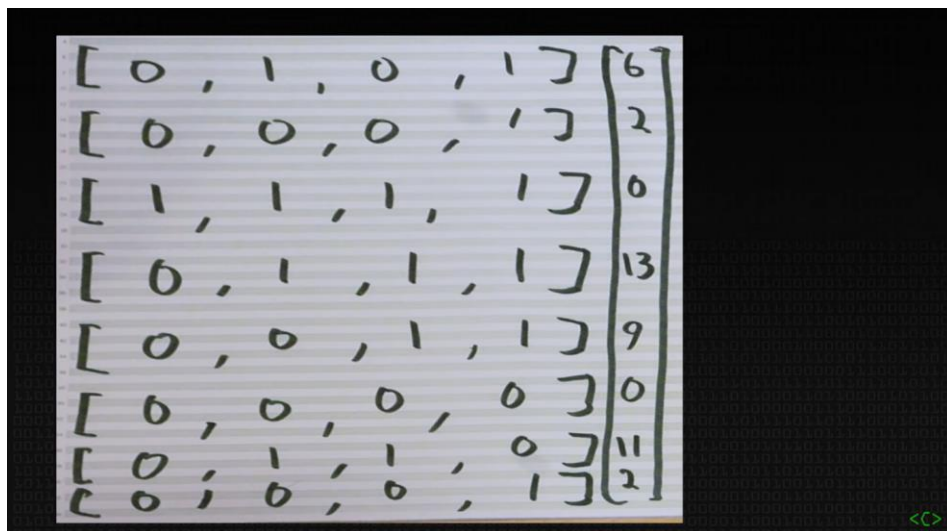
Hình 12 Minh họa cho giá trị hàm fitness dựa trên tổ hợp các vật phẩm thể hiện dưới dạng vector nhị phân.



Hình 13 Một trường hợp khác với giá trị fitness bằng 0

2. Khởi tạo quần thể.

Như đã đề cập ở trên về tầm quan trọng của quần thể khởi tạo đến kết quả của bài toán, tuy nhiên để đơn giản, ta sẽ khởi tạo quần thể này một cách ngẫu nhiên và không có chủ ý.

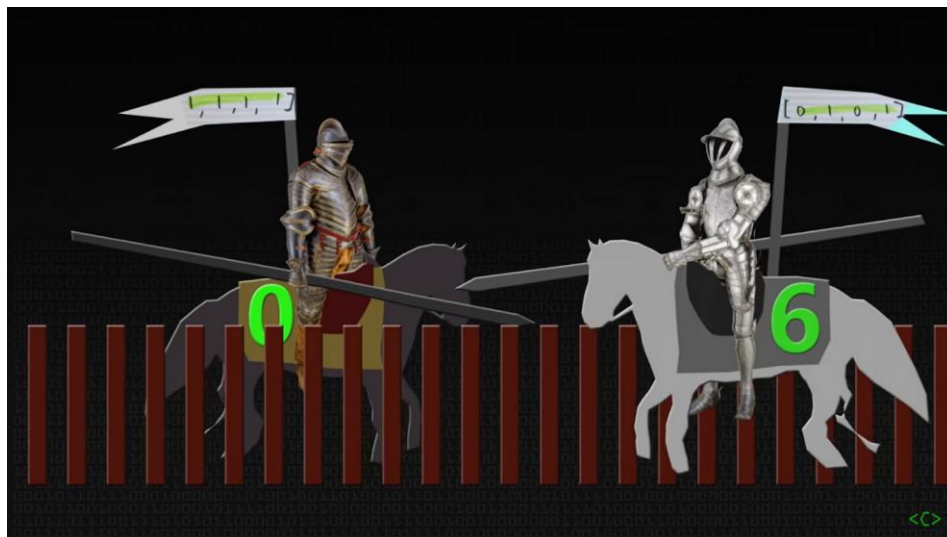


Hình 14 Quần thể khởi tạo (nhiễm sắc thể bên trái và giá trị fitness bên phải)

Để dễ trực quan nên quần thể này sẽ có 8 cá thể (trong thực tế thường có nhiều hơn như cỡ 100 hoặc 1000).

3. *Bước chọn (tournament selection).*

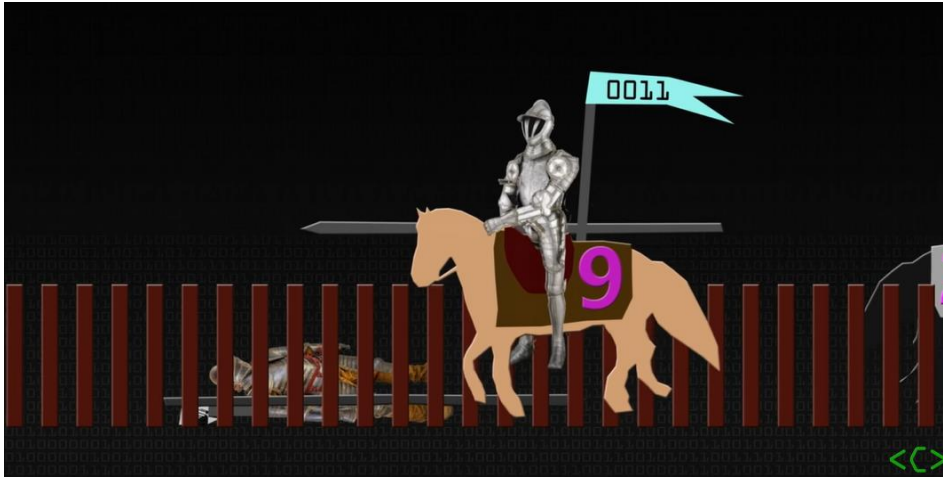
Nhóm sử dụng chọn theo kiểu đấu trường để chọn ra các cá thể phù hợp cho việc tái sản xuất.



Hình 15 Đấu trường của 2 cá thể với độ phù hợp được thể hiện trên yên ngựa.



Hình 16 Cá thể với độ phù hợp cao hơn sẽ được chọn cho việc tái sản xuất.



Hình 17 Làm tương tự với các vòng đấu khác để chọn ra các cá thể phù hợp cho việc tái sản xuất.

Và cứ sau mỗi vòng đấu, chúng ta lại chọn ra một phụ huynh để thực hiện tái sản xuất, và ta cứ liên tục làm như vậy tới khi số lượng cá thể của quần thể trong thế hệ kế tiếp bằng với số cá thể trong quần thể của thế hệ hiện tại.

Lưu ý ở đây nhóm chọn kích thước đấu trường này là 2, trên thực tế để cho bước chọn lọc khách quan hơn, chúng ta có thể chọn kích thước đấu trường lớn hơn, độ lớn có thể có kích cỡ 2^n với n không quá lớn và không quá nhỏ, lấy ví dụ ($n \in [1,4]$)

4. Lai tạo (One-point crossover).

Nhóm sử dụng lai tạo một điểm, sử dụng một đoạn gen của cá thể A và một đoạn gen của cá thể B của thế hệ cũ, ghép lại với nhau để tạo ra một nhiễm sắc thể mới của thế hệ kế tiếp, lấy ví dụ 2 phụ huynh A và phụ huynh B là các nhiễm sắc thể được chọn ở bước chọn lọc.

Phụ huynh A: $[0, 1, 0, 1]$ \Rightarrow Con1 $[0, 1, 1, 1]$

Phụ huynh B: $[0, 0, 1, 1]$ \Rightarrow Con2 $[0, 0, 0, 1]$

Hình 18 Sử dụng lai tạo 1 điểm từ bước 3.

5. Đột biến (bit-flip mutation).

Nhóm sử dụng đột biến gen ở một vị trí ngẫu nhiên với tỉ lệ đột biến cho trước. Lúc này nhiễm sắc thể “Con1” sẽ trở thành một nhiễm sắc thể mới như hình:

Con1 [0,1,1,0]

Hình 19 Nhiễm sắc thể hậu biến đổi gen

6. Tiêu chí dừng.

Tùy vào bài toán mà sẽ có các tiêu chí dừng khác nhau. Để tổng quan nhất thì nhóm sẽ dừng sau khi đạt số thế hệ tối đa. Và như đã đề cập, thế hệ cuối cùng này được kỳ vọng mỗi cá thể sẽ có độ phù hợp cao hơn so với các thế hệ trước đó.

CHƯƠNG 3. CÁC KẾT QUẢ THỰC NGHIỆM

3.1. Các Tình Huống

Trong quá trình thực hiện ứng dụng giải thuật di truyền cho bài toán knapsack, nhóm nhận thấy các siêu tham số như số lượng cá thể thuộc quần thể, tổng số thế hệ, tỉ lệ lai tạo và tỉ lệ đột biến cũng như các cách thức chọn lọc, lai tạo, đột biến hoặc là các tiêu chí dừng và quần thể khởi tạo, độ khó bài toán, v.v... đều ảnh hưởng lớn đến kết quả bài toán.

1. Ảnh hưởng của các siêu tham số.

- a. Kích thước quần thể: Đây là một trong các siêu tham số quan trọng quyết định kết quả của giải thuật là tốt hay xấu. Thông thường chúng ta không muốn một quần thể quá lớn (làm tăng không gian tìm kiếm dẫn đến tăng thời gian tính toán), nhưng đồng thời chúng ta cũng không muốn một kích thước quần thể quá nhỏ (một quần thể quá nhỏ làm hạn chế không gian tìm kiếm, do đó làm cho kết quả của thuật toán không thật sự tốt). Ở một vài nghiên cứu đã chỉ ra việc tăng kích thước quần thể thường dẫn tới một kết quả tốt hơn của giải thuật di truyền (Stanley Gotshall & Bart %J Population Rylander, 2002). Trong nghiên cứu này cũng đã chỉ ra các lợi ích của việc sử dụng một kích thước quần thể lớn đó chính là:

- Tăng kích thước quần thể cũng làm tăng khả năng quần thể ở giai đoạn khởi tạo có chứa nhiễm sắc thể với độ phù hợp cao.
- Việc tăng kích thước quần thể cũng dẫn đến việc số thế hệ cần thiết để hội tụ giảm đi (cần ít thế hệ hơn để hội tụ).
- Kích thước quần thể tối ưu cho một vấn đề là điểm mà ở đó chỉ cần ít thế hệ để hội tụ và cân bằng với lợi ích của độ chính xác gia tăng khi kích thước quần thể tăng lên.

Tuy nhiên vẫn có những nghiên cứu chỉ ra rằng việc tăng kích thước của quần thể đôi khi cũng dẫn tới một kết quả tệ hơn khi áp dụng giải thuật di truyền (Tianshi Chen, Ke Tang, Guoliang Chen, & Xin Yao, 2012).

- b. Số lượng thế hệ: Đây cũng là một trong các siêu tham số quan trọng cần được xác định để giúp cải thiện kết quả của giải thuật di truyền. Ta thường muốn chọn các cặp siêu tham số kích thước quần thể và số lượng thế hệ một cách hợp lý để không cần phải tốn quá nhiều chi phí tính toán để đạt được một kết quả tốt, một bài nghiên cứu đã chỉ ra điều này (Dana Vrajitoru, 2000). Trong bài nghiên cứu cũng chỉ ra rằng việc tăng kích thước quần thể dẫn tới kết quả tốt hơn so với việc tăng số lượng thế hệ (miễn bộ nhớ cho phép). Bên cạnh đó bài nghiên cứu cũng đã thực nghiệm với số lượng thế hệ rất nhỏ cũng dẫn đến các kết quả khác.
- c. Tỉ lệ lai tạo: Đây là một tham số quan trọng điều khiển khả năng có sự lai tạo giữa các cá thể phụ huynh. Tham số này đặc biệt ở chỗ nếu chúng ta đặt tỉ lệ lai tạo quá cao, dẫn đến việc lai tạo gần như sẽ xảy ra với các cặp phụ huynh được lựa chọn cho quá trình tái sản xuất, điều này sẽ dẫn đến sự đa dạng trong quần thể, còn nếu như tỉ lệ lai tạo thấp thì ngược lại. Thông qua kiểm chứng thực nghiệm, một nghiên

Commented [NQ1]: Nếu không có hàm selection thì sao?

Commented [NQ2R1]: Nếu không có hàm crossover thì sao?

cứu đã chỉ ra các khoảng cho các siêu tham số của giải thuật di truyền mà ở đó khi áp dụng ta thường có một kết quả tốt(KA %J Computer DeJong & University of Michigan Communication Sciences, Ann Arbor, 1975). Bên cạnh đó cũng có các bài nghiên cứu đã ứng dụng các tỉ lệ thích nghi (adaptive crossover rate) để cải thiện sự chính xác của thuật toán, điểm mạnh của việc sử dụng các tỉ lệ thích nghi đó là chúng ta không cần quá quan tâm tới việc phải tinh chỉnh các siêu tham số một cách thủ công(Qi Zhu, 2018).

- d. Tỉ lệ đột biến: Đây cũng là một siêu tham số quan trọng trong việc giải các bài toán giải thuật di truyền. Việc đột biến có thể dùng để ngăn cản vấn đề hội tụ sớm, nhưng nếu như tỉ lệ đột biến quá cao sẽ khiến cho giải thuật di truyền trở thành tìm kiếm ngẫu nhiên, do đó mà tỉ lệ đột biến thường không nên quá cao, các tham số được khuyến nghị cũng được đề cập trong một nghiên cứu đã được thực hiện trước đó(M. Srinivas & L. M. Patnaik, 1994).
- e. Các siêu tham số khác: Bên cạnh các siêu tham số được đề cập ở trên, đối với mỗi toán tử di truyền khác nhau cũng sẽ thường yêu cầu thêm các siêu tham số khác, ví dụ như toán tử tái sản xuất (chọn lọc), ta có thể sử dụng chọn lọc theo kiểu đấu trường và lúc này siêu tham số mới chính là số cá thể được cho vào đấu trường. Tuy nhiên các siêu tham số này thường không quan trọng bằng các siêu tham số vừa được đề cập nên sẽ được cho theo như các giá trị mặc định/khuyến nghị của các chuyên gia.

2. Ảnh hưởng của các toán tử di truyền.

Như đã đề cập ở trên, các toán tử di truyền đóng một vai trò rất quan trọng trong việc cải thiện kết quả của giải thuật di truyền. Một vài nghiên cứu đã được thực hiện bằng cách thử nghiệm các tổ hợp toán tử di truyền khác nhau để cho ra các kết quả tốt, ví dụ như có một bài nghiên cứu gợi ý rằng chúng ta nên lai tạo “tham lam” (“greedy” crossover) và tìm kiếm “khó” (“hard” selection) cùng với tỉ lệ lai tạo thấp thường cải thiện hiệu quả của thuật toán (Xin Yao, 1993) .

- a. Thay đổi toán tử chọn lọc: Trong một bài nghiên cứu thực nghiệm được thực hành vào năm 2017 đã chỉ ra rằng các toán tử chọn lọc khác nhau cũng ảnh hưởng tới hành vi (ví dụ như độ phù hợp biến động hơn) của giải thuật trong các thế hệ cho tới khi đạt tiêu chí dừng(Saneh Lata, Saneh Yadav, & Asha Sohal, 2017) .
- b. Thay đổi toán tử lai tạo: Tương tự như với toán tử chọn lọc, rất nhiều nghiên cứu thực nghiệm cũng được thực hiện cho việc lai tạo(Anant J Umbarkar & Pranali D %J ICTACT journal on soft computing Sheth, 2015) và trong bài nghiên cứu này tác giả nhận định rằng việc xác định toán tử lai tạo cũng phụ thuộc vào việc chúng ta mã hóa thông tin như thế nào (ví dụ như với bài toán knapsack thì nhóm thực hiện mã hóa theo kiểu mã hóa nhị phân). Bài nghiên cứu còn chỉ ra rằng chúng ta nên tham khảo trước không gian tìm kiếm trước khi chọn ra toán tử lai tạo phù hợp.

- c. Thay đổi toán tử đột biến: Tương tự như hai toán tử ở trên, những người nghiên cứu cũng quan tâm đến các toán tử đột biến ảnh hưởng đến hành vi của giải thuật như thế nào (Alan Piszcz & Terence Soule, 2006).

3. Bảo tồn trạng thái tốt.

Trong quá trình thực nghiệm, nhóm nhận thấy rằng khi áp dụng giải thuật di truyền, các thế hệ sau không phải lúc nào cũng tốt hơn các thế hệ trước đó, nguyên nhân có thể là do việc hội tụ sớm ở một thế hệ trước đó, điều này kết hợp với việc tiêu chí dừng chưa thỏa mãn sẽ dẫn tới việc hình thành các quần thể mới của các thế hệ tiếp theo. Và cũng vì sự trao đổi gen, đột biến của các cá thể trong quần thể mà làm mất đi kết quả vốn đã hội tụ tới một điểm phù hợp. Do vậy mà chúng ta quan tâm tới việc bảo tồn các trạng thái tốt này. Có nhiều cách tiếp cận khác nhau để có thể bảo tồn các trạng thái tốt, nhưng cũng có một vài phương pháp mà cần sự góp ý của chuyên gia để điều kiện dừng thêm hợp lý.

- a. Hướng tiếp cận tinh hoa (elitism).
- b. Hướng tiếp cận theo tiêu chí dừng delta.

Ngoài 2 hướng tiếp cận trên, một vài nghiên cứu cũng đã chỉ ra một vài hướng tiếp cận khác cho tiêu chí dừng, ví dụ như trong một bài nghiên cứu về tiêu chí dừng của giải thuật di truyền đã đề xuất một phương hướng mới đó là sẽ dừng khi đạt tới một số lượng thế hệ cần thiết để đạt được khoảng tin cậy để đảm bảo rằng giải thuật di truyền đã thấy qua tất cả nhiễm sắc thể bên trong nó (bao gồm điểm tối ưu)(Haldun Aytug & Gary J. Koehler, 2000).

4. Dữ liệu từ bài toán.

Như đã đề cập, việc giải thuật di truyền có ứng dụng tốt được hay không còn phụ thuộc vào chính vấn đề của bài toán. Đối với các bài toán có yêu cầu dễ đáp ứng (ví dụ bài toán knapsack với các vật phẩm như nhau (cả về cân nặng lẫn giá trị và một cái balo có sức chứa lớn (ví dụ như bằng tổng tất cả các vật thể)) thì kết quả sẽ dễ dàng hội tụ, và sẽ không trải qua hiện tượng độ phù hợp bị sụt giảm qua các thế hệ. Ngược lại, nếu bài toán đó là bài toán khó đáp ứng (ví dụ như các vật phẩm có phân bố về cân nặng và giá trị rất biến động, kết hợp với sức chứa của balo là rất nhỏ) thì kết quả sẽ khó đạt tới một kết quả tốt.

3.2 Phân tích và đánh giá.

Trong các phân tích của nhóm về hành vi của giải thuật, bên cạnh các tình huống được nêu ở mục 3.1. Bên cạnh đó chúng ta cũng cần một benchmark để xác định xem giải thuật của chúng ta có thực sự hoạt động tốt hay không. Và với ví dụ về bài toán knapsack, nhóm sử dụng hướng tiếp cận theo kiểu brute-force để tìm ra kết quả chính xác của bài toán và sẽ sử dụng kết quả này làm thước đo.

```

❏ values: [82, 15, 4, 95, 36, 32, 29, 18, 95, 14, 87, 95, 70, 12, 76, 55, 5, 4, 12, 28]
weights: [8, 17, 20, 1, 18, 7, 18, 14, 8, 15, 19, 9, 1, 6, 14, 11, 9, 5, 7, 11]
knapsack's capacity: 130
Best Selection: [1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1]
Best Value: 784
Total Weight: 130

```

Hình 20 Benchmark.

Trong hình trên, nhóm sử dụng các hàm random (có sử dụng seed để các số không bị thay đổi sau mỗi lần chạy lại cell) để tạo ra weights và values, bên cạnh đó sức chứa của knapsack là 0.6 lần tổng khối lượng của tất cả vật thể. 3 dòng cuối là dùng hướng tiếp cận bruteforce để tìm ra kết quả chính xác nhất của bài toán để làm đối chiếu.

Chú thích: Các thực nghiệm của nhóm đều đã cố định quần thể khởi tạo bằng hàm trong python (random.seed(42)) để các thực nghiệm trở nên công bằng hơn.

Nhóm thực hiện so sánh kết quả ở trên với giải thuật di truyền sử dụng các tham số được khuyến nghị với các toán tử di truyền (chọn lọc: đấu trường, lai tạo: 1 điểm, đột biến: bit flip) với các tham số (kích thước quần thể = 100, số lượng thế hệ = 100, tỉ lệ lai tạo = 0.5, tỉ lệ đột biến = 0.001).

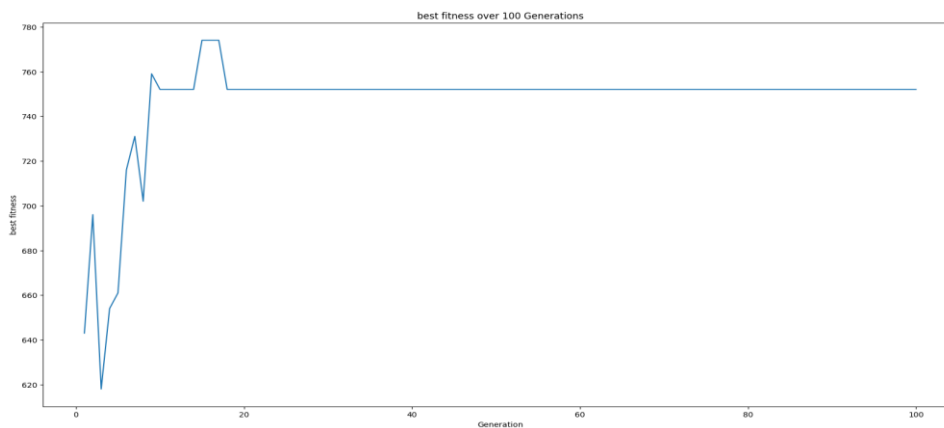
```

❏ =====
The exact solutions for knapsack is: [1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1]
Maximum values of knapsack is: 784
Maximum weight of knapsack is: 130
=====
Best individual in population in final generation: [1, 0, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0]
Best fitness in population in final generation: 752
Weights of the best individual: 130
=====

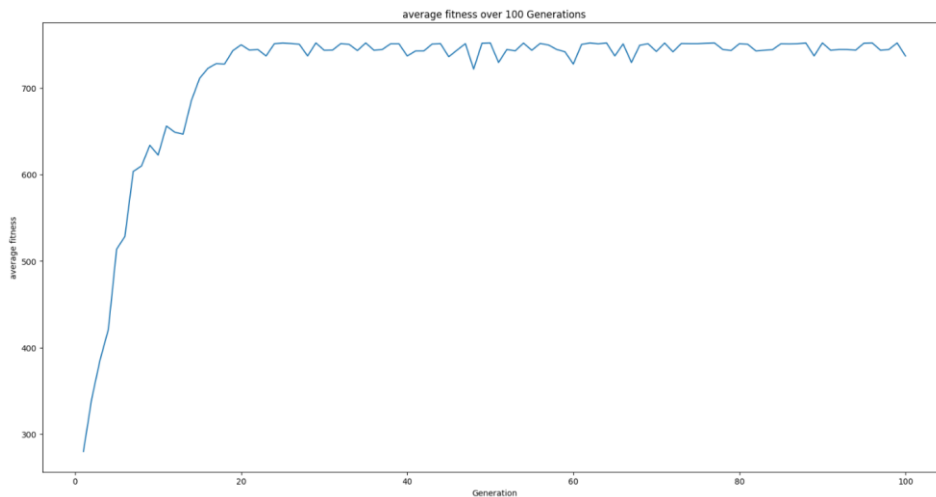
```

Hình 21 Sử dụng các tham số khuyến nghị

Nhóm quan sát hành vi của giải thuật qua từng thế hệ (chỉ lấy cá thể có độ phù hợp cao nhất của mỗi thế hệ cho vào hình vẽ).



Hình 22 Hành vi của cá thể phù hợp nhất trong mỗi thế hệ trong giải thuật di truyền với tham số mặc định



Hình 23 Hành vi của độ phù hợp toàn quần thể trong mỗi thế hệ trong giải thuật di truyền với tham số mặc định.

Ngoài ra nhóm còn sử dụng một threshold và xem đây như là thước đo cho sự phù hợp của kết quả.

```
[19] 1 convergence_rate(best_fitness_standard, 699.5)

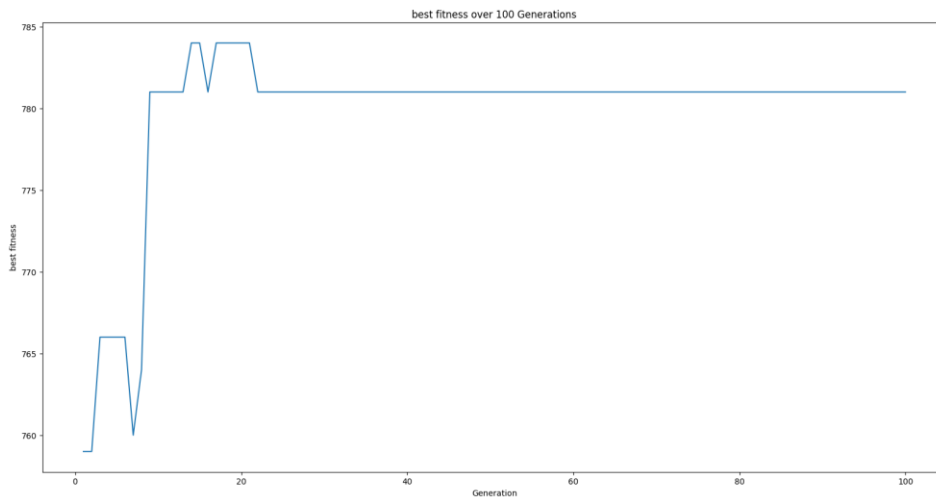
After 6 generations, we reached fitness's threshold (699.5)
```

Hình 24 Hàm cho thấy tốc độ hội tụ tới một kết quả chỉ định là nhanh hay chậm.

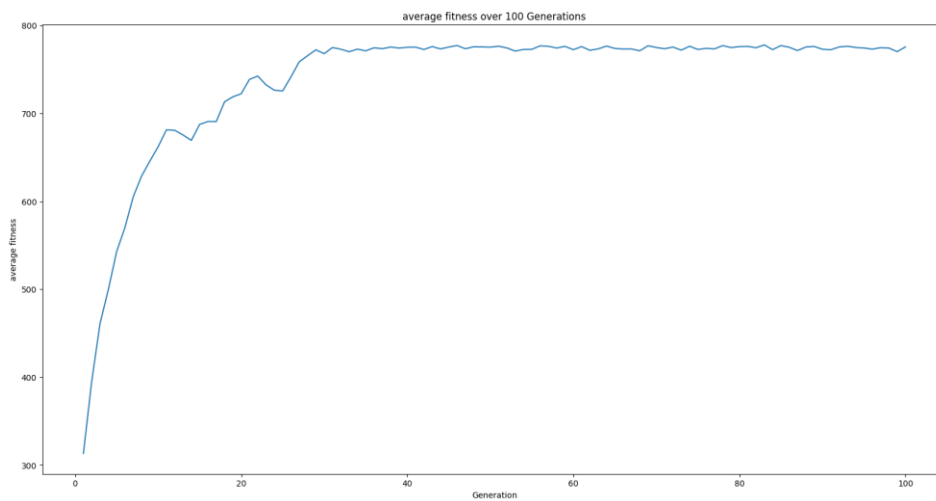
1. Vấn đề về các siêu tham số:

a. Tăng kích thước quần thể.

Nhóm thực hiện việc tăng kích thước của quần thể lên 1000 thay vì 100 như trước và quan sát hành vi của giải thuật (về cá thể phù hợp nhất và trung bình quần thể) trong mỗi thế hệ. Và như đã kì vọng ở trên, việc tăng số lượng cá thể trong quần thể cũng đồng thời tăng không gian tìm kiếm. Từ đó giúp thuật toán mau chóng tìm ra các kết quả tốt hơn, thể hiện điều này rõ nhất ta có thể sử dụng hàm `convergence_rate` được tạo ra để củng cố nhận định.



Hình 25 Khi kích thước quần thể là 1000



Hình 26 Trung bình độ phù hợp khi kích thước quần thể là 1000

```
6 convergence_rate(best_fitness_standard,750)
7 plot_best_fit(best_fitness_standard, num_generations)
8 plot_avg_fit(avg_fitness_standard, num_generations)

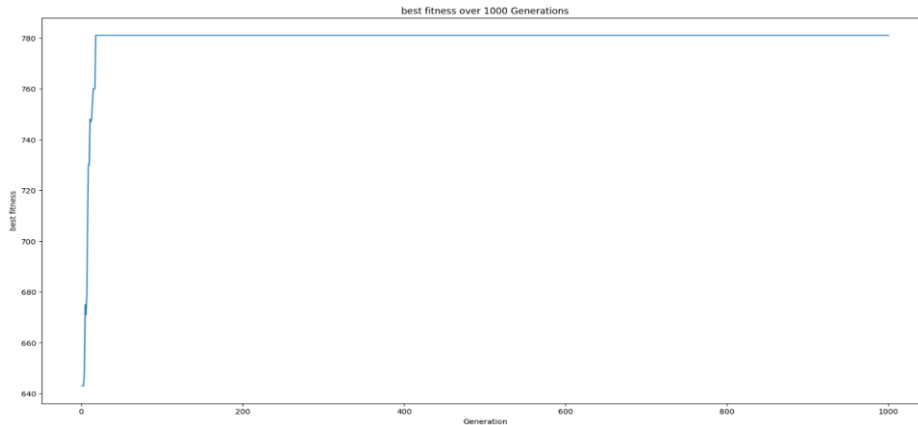
=====
After 1 generations, we reached fitness's threshold (750)
```

Hình 27 Tốc độ hội tụ của thuật toán trên một không gian rộng hơn.

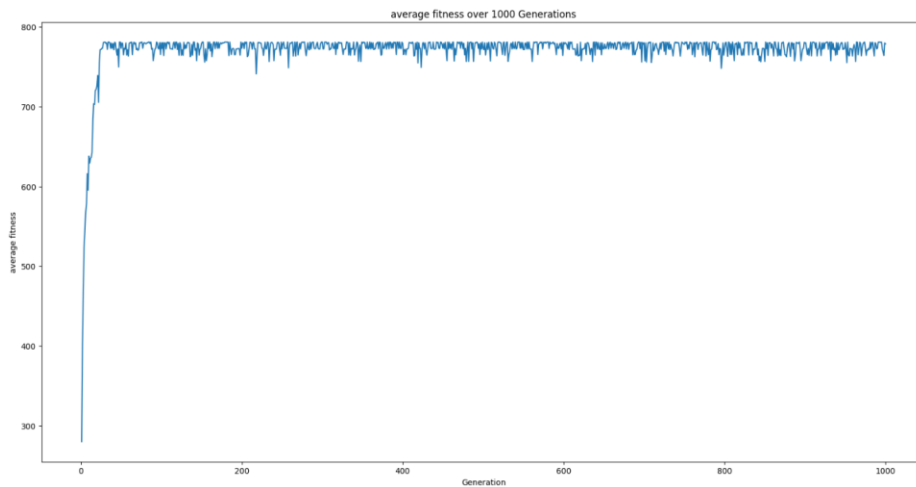
Nhận xét: Khi chúng ta có một không gian tìm kiếm rộng hơn, giải thuật di truyền sẽ cho ra các kết quả tốt hơn và đạt tới một kết quả tốt nhanh hơn. Như ví dụ ở trên, ta thấy đối với các tham số mặc định và tham số đã điều chỉnh kích thước quần thể, GA với kích thước quần thể lớn hơn đạt được tốc độ tiến tới một kết quả tốt nhanh hơn là với các tham số mặc định (như các nghiên cứu đã chỉ ra ở trên). Tuy vậy, để phục vụ cho việc gia tăng số lượng cá thể trong quần thể yêu cầu nhiều chi phí tính toán hơn, do đó sẽ không phù hợp trong thực tế nếu như bộ nhớ là một hạn chế. Bên cạnh đó khi nhìn vào đồ thị về giá trị phù hợp trung bình của quần thể trong mỗi thế hệ, ta nhận thấy rằng ở các thế hệ sau, độ phù hợp trung bình của quần thể ở các thế hệ sau có xu hướng hội tụ tới một giá trị phù hợp nhất định, sự hội tụ này cũng diễn ra nhanh chóng, trong khi với các tham số mặc định, các quần thể sẽ cần nhiều thế hệ đơn để hội tụ (giá trị ở phần đuôi dao động mạnh hơn giá trị ở phần đầu sau khi điều chỉnh tham số).

b. Tăng số lượng thế hệ.

Để kiểm chứng việc số lượng thế hệ có ảnh hưởng đến kết quả của bài toán, nhóm thực hiện việc thay đổi số lượng thế hệ từ 100 lên thành 1000 và quan sát hành vi của giải thuật với số lượng thế hệ mới này.



Hình 28 Khi tăng số lượng thế hệ lên 1000



Hình 29 Hành vi thuật toán khi nhiều thể hệ hơn

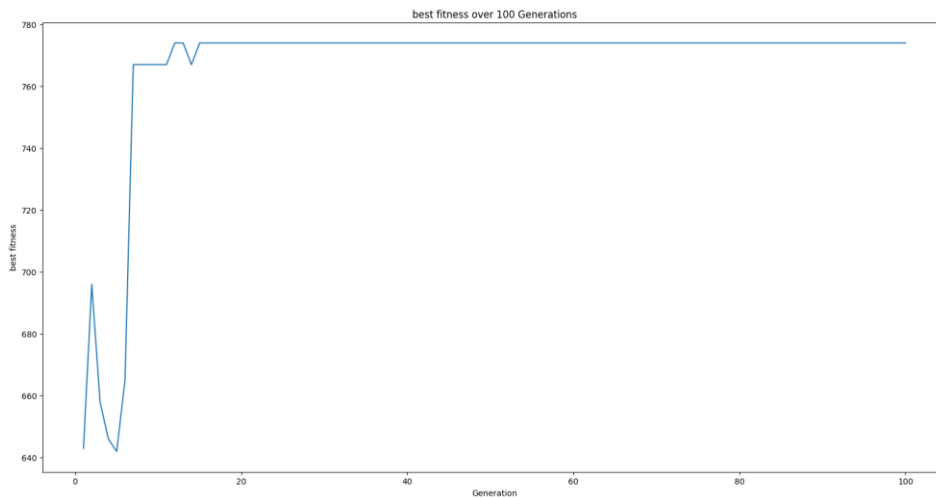
After 15 generations, we reached fitness's threshold (760)

Hình 30 Tốc độ hội tụ của thuật toán khi nhiều thể hệ hơn.

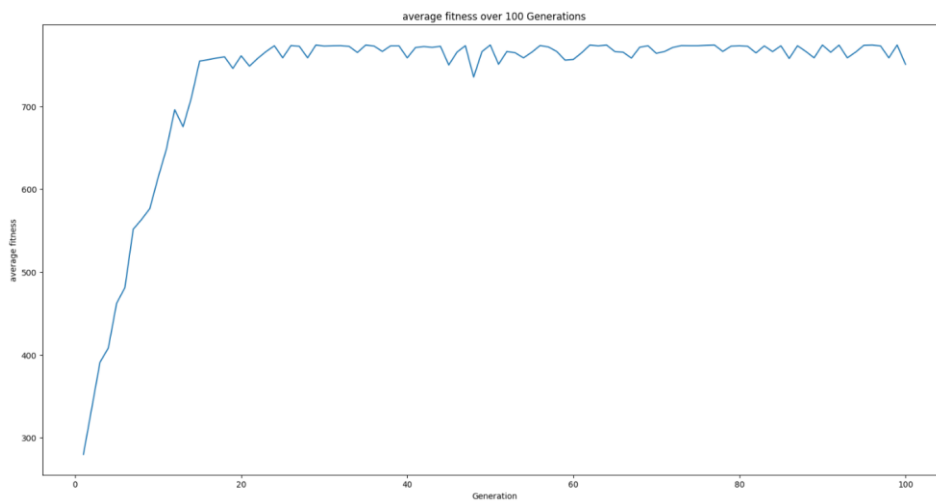
Nhân xét: Khi chúng ta sử dụng giải thuật di truyền sử dụng nhiều thể hệ hơn, điều này làm tăng thời gian tính toán, khiến cho thời gian xử lý dài hơn, đặc biệt là khi kích cỡ bài toán hoặc là độ phức tạp của bài toán là cao. Ngoài ra chúng ta cũng bắt gặp hiện tượng giảm điểm đầu ra (diminishing return) bởi vì sau một số thể hệ nhất định, giải thuật của chúng ta có thể đạt tới điểm này, mà ở đó những vòng lặp qua các thể hệ tiếp theo cung cấp rất ít hoặc đôi khi không cải thiện được chất lượng của kết quả. Điều này có thể là do sự thiếu đa dạng trong quần thể của chúng ta và điều này có thể cải thiện bằng cách thay đổi tỉ lệ đột biến một cách hợp lý hơn (hoặc chúng ta cũng có thể tăng không gian tìm kiếm bằng cách thay đổi kích thước quần thể sao cho phù hợp hơn). Hơn nữa, không như việc chúng ta mở rộng không gian tìm kiếm như chúng ta đã làm với thí nghiệm tăng kích thước quần thể, việc tăng số lượng thể hệ chỉ tăng số lần vòng lặp với hy vọng các thể hệ sau tốt hơn các thể hệ đầu, vì lý do đó mà việc tăng số lượng thể hệ không ảnh hưởng gì đến tốc độ hội tụ của giải thuật di truyền. Ngoài ra khi nhìn vào sơ đồ của giá trị trung bình độ phù hợp của quần thể trong mỗi thể hệ, ta thấy rằng việc tăng số lượng thể hệ cũng không gây ra ảnh hưởng lên sự dao động của giá trị trung bình độ phù hợp của quần thể (không như việc tăng số kích thước quần thể, bởi vì tăng kích thước quần thể làm giảm sự dao động này).

c. Thay đổi tỉ lệ lai tạo lên cao

Như đã đề cập về tầm quan trọng của biến tỉ lệ lai tạo trong giải thuật di truyền, nhóm quan tâm tới việc xem xét hành vi của giải thuật khi tỉ lệ lai tạo lên cao từ 0.5 lên 0.85).



Hình 31 Khi thay đổi tỉ lệ lai tạo lên cao



Hình 32 Trung bình độ phù hợp quần thể khi tăng tỉ lệ lai tạo lên cao

After 7 generations, we reached fitness's threshold (760)

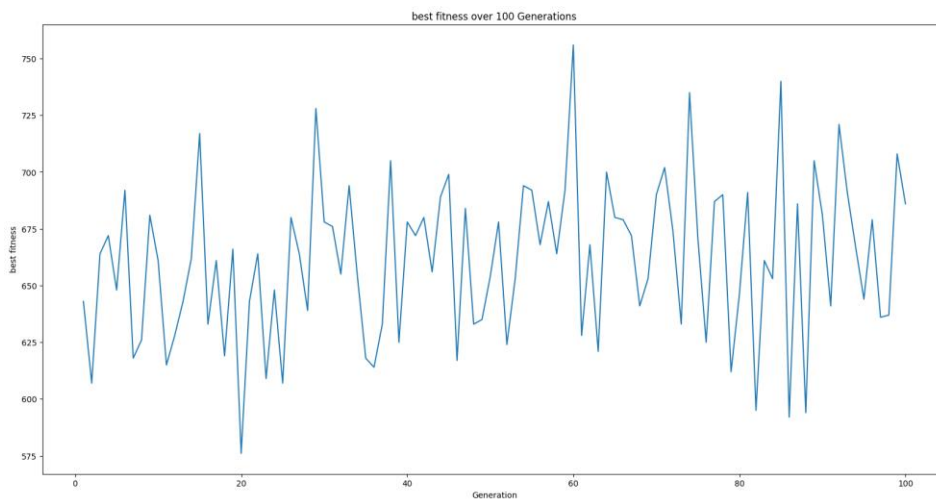
Hình 33 Tốc độ đạt tới kết quả tốt khi tỉ lệ lai tạo tăng lên

Nhận xét: Việc tăng tỉ lệ lai tạo lên cao cho phép một phần lớn cá thể trong quần thể tham gia quá trình lai tạo và việc này thúc đẩy việc có những sự kết hợp khác nhau của chất liệu di truyền từ cá thể cha và cá thể mẹ, do đó dẫn đến việc khám phá rộng hơn trong không gian tìm kiếm và cho phép tìm ra các giải pháp mới và đa dạng hơn. Tuy nhiên việc có tỉ lệ lai tạo quá cao cũng là một lí do ảnh hưởng đến kết quả, cụ thể là do việc có khả năng

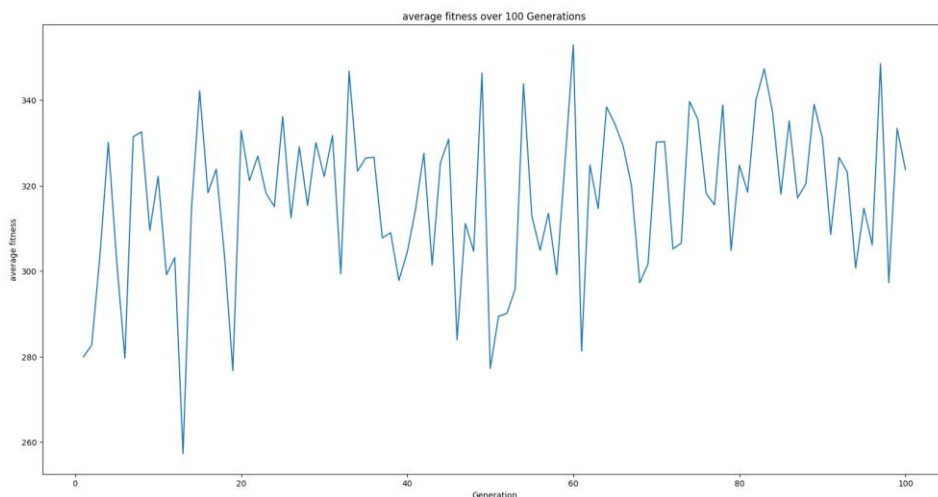
cao trộn lẫn chất liệu di truyền từ cá thể phụ huynh có thể dẫn đến các trở ngại cho việc đạt được một kết quả tốt (đặc biệt là khi các kết quả đủ tốt bị thay đổi do tỉ lệ lai tạo quá cao, làm mất đi đặc tính vốn đã tốt của nó). Bên cạnh đó tỉ lệ lai tạo quá cao cũng làm giảm tốc độ hội tụ bởi vì việc liên tục tái tổ hợp lại vật liệu di truyền làm giảm đi tốc độ đạt tới kết quả mong muốn của giải thuật di truyền khiến cho giải thuật của chúng ta khó đạt được một kết quả mong muốn (bởi vì sự mất đi của các vật liệu di truyền tốt), tuy nhiên điều này vẫn cần được bàn luận thêm bởi vì tác động này của việc tỉ lệ lai tạo cao là chưa rõ ràng hoặc khó nhận biết.

d. Thay đổi tỉ lệ đột biến lên cao.

Tỉ lệ đột biến cũng là một tham số quan trọng trong việc điều chỉnh hành vi của giải thuật di truyền tới một kết quả tốt hơn. Nhóm thay đổi tỉ lệ đột biến được khuyến nghị(từ 0.001 lên 0.5) và quan sát hành vi của thuật toán.



Hình 34 Khi thay đổi tỉ lệ đột biến lên cao



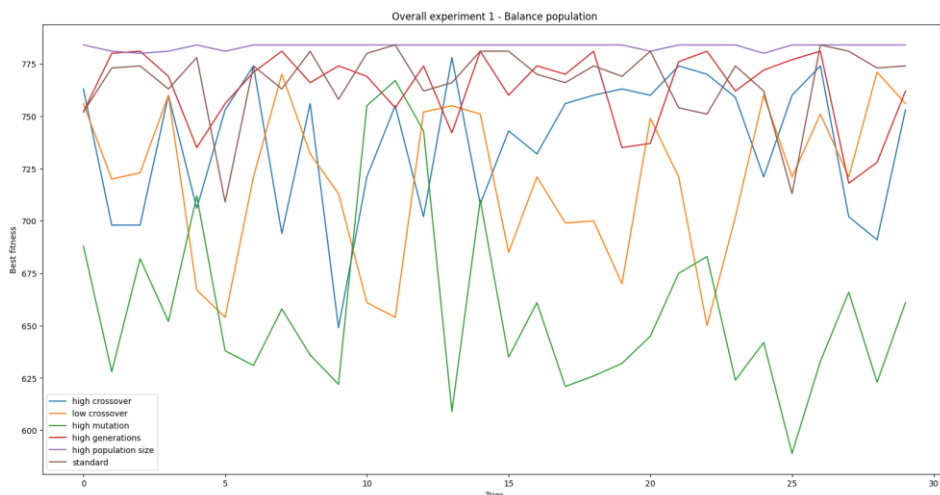
Hình 35 Hành vi của độ phù hợp trung bình quần thể khi tỉ lệ lai tạo cao

Can't reach the limit threshold (760)

Hình 36 Khi tỉ lệ lai tạo quá cao, giải thuật di truyền không đạt được tới kết quả tốt

Nhận xét: Việc tăng tỉ lệ đột biến lên quá cao gây ra ảnh hưởng tiêu cực đến chất lượng của kết quả bài toán knapsack. Lí do là vì khi tỉ lệ đột biến được đẩy lên quá cao, giải thuật di truyền giờ đây sẽ giống nhau việc chúng ta tìm kiếm ngẫu nhiên. Khi tỉ lệ đột biến quá cao, các cá thể có độ phù hợp cao bị đột biến làm cho thay đổi, từ đó trở thành các cá thể kém phù hợp hơn. Việc đột biến xảy ra thường xuyên tạo ra kết quả tiêu cực tới kết quả bài toán. Ngoài ra việc thay đổi tỉ lệ đột biến dường như thay đổi luôn hành vi của giải thuật trong dài hạn. Đối với các tham số vừa thực nghiệm ở trên, ta nhận thấy trong dài hạn, sẽ có sự ổn định về độ phù hợp của toàn bộ quần thể, và quan sát chung, nhận xét thấy các quần ở các thế hệ sau luôn tốt hơn các quần thể ở các thế hệ đầu tiên. Tuy nhiên khi thay đổi tỉ lệ đột biến lên quá cao, sự đột biến làm mất đi đặc tính mạnh mẽ này của giải thuật di truyền (lúc này không có sự chuyển biến tích cực về độ phù hợp của các cá thể trong các thế hệ sau). Do đó ta có thể thấy rằng các tỉ lệ đột biến không nên đặt quá cao do nó sẽ ảnh hưởng mạnh đến hành vi của giải thuật.

Tổng kết thí nghiệm 1: Qua thực nghiệm, nhóm nhận xét thấy rằng việc tăng kích thước quần thể sẽ làm tăng hiệu suất của giải thuật di truyền và việc tăng kích thước quần thể cũng ảnh hưởng nhiều hơn so với thay đổi số lượng thế hệ. Cũng có một vài nghiên cứu khác với kết quả cho được tương tự như những gì nhóm thực nghiệm. (Y. R. Tsoy, 2003). Xu hướng chung, ta thấy giải thuật di truyền cho ra kết quả khả quan khi các quần thể ở các thế hệ sau có độ phù hợp cao hơn các quần thể ở các thế hệ đầu tiên (đặc biệt là so với thế hệ khởi tạo). Ngoài ra, việc có tỉ lệ đột biến quá cao cũng làm giảm đáng kể chất lượng kết quả của giải thuật di truyền.



Hình 37 Bảng tổng kết cho sự thay đổi các siêu tham số.

Lưu ý các điểm dữ liệu từ hình vẽ trên được tạo bởi cá thể tốt nhất trong quần thể cuối cùng trong thế hệ cuối cùng (hoặc thế hệ cuối cùng trước khi bị tiêu chí dừng ngắt đi).

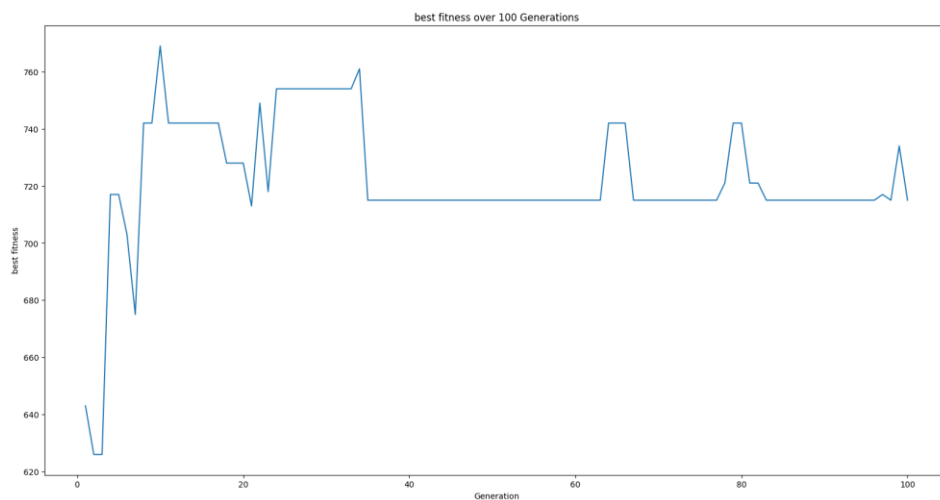
Dựa trên hình trên ta cũng có thể thấy, việc tăng số lượng cá thể trong quần thể là một cách làm cho ra kết quả ổn định và khả năng cao hội tụ tới điểm tối ưu (hoặc gần điểm tối ưu). Tuy nhiên như đã đề cập, tăng kích thước quần thể cũng làm tăng chi phí tính toán. Bên cạnh đó, một lựa chọn ít chi phí tính toán hơn đó chính là sử dụng cách tiếp cận kiểu mặc định (kết quả không cao bằng, tuy nhiên vẫn nằm ở mức chấp nhận được, bên cạnh đó tốn ít chi phí tính toán hơn). Và kể đến chính là tăng số lượng thế hệ, cách làm này cho ra độ phù hợp xấp xỉ với phương pháp truyền thống (nhưng với chi phí tính toán cao hơn). Kể đến chính là 2 đường thể hiện tỉ lệ lai tạo cao và tỉ lệ lai tạo thấp, với tỉ lệ lai tạo cao, chúng ta có thể đạt được những kết quả tốt hơn so với tỉ lệ lai tạo thấp, lí do đã được giải thích ở trên, khi việc tăng tỉ lệ lai tạo làm tăng khả năng các nhiễm sắc thể được tái tổ hợp lại nhiều hơn, từ đó chúng ta có thể khai phá rộng hơn không gian tìm kiếm. Còn việc tăng tỉ lệ đột biến là một bước suy thoái trong quá trình tiến hóa, nên tránh sử dụng (đối với bài toán knapsack này).

2. Các toán tử di truyền.

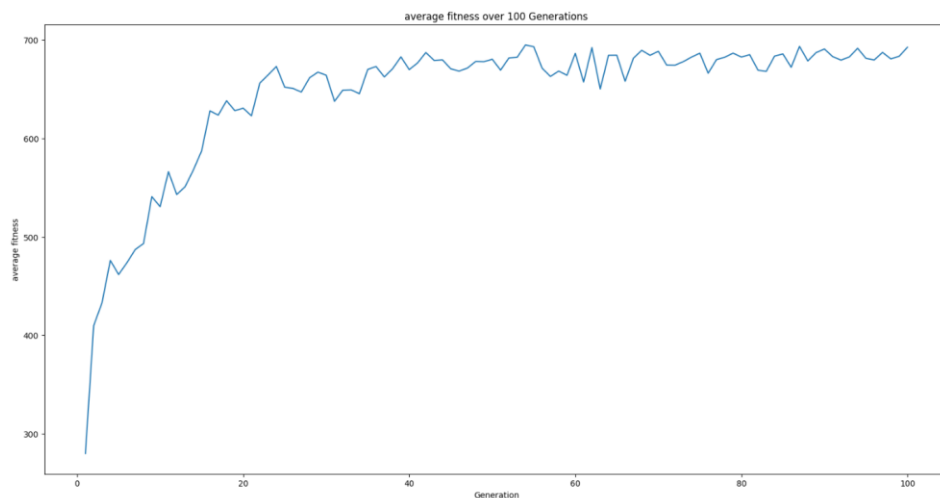
Như đã đề cập về tầm quan trọng của các toán tử di truyền, nhóm tiến hành thực nghiệm các toán tử di truyền với nhau, mỗi thực nghiệm với toán tử di truyền khác nhau đều được đặt dưới giả định các yếu tố khác không đổi (ví dụ như so sánh các phương pháp chọn khác nhau sẽ được tiến hành dưới giả sử các toán tử lai tạo, đột biến là không đổi).

a. Thay đổi toán tử chọn lọc.

Các toán tử chọn lọc đóng vai trò quan trọng trong việc điều hướng hành vi của giải thuật di truyền. Nhóm tiến hành so sánh giữa 3 phương pháp chọn: đấu trường, vòng xe, xếp hạng.



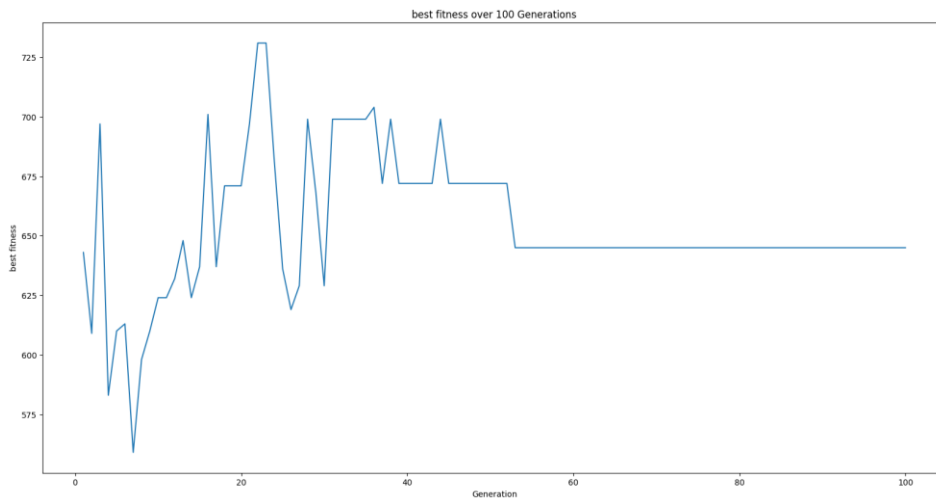
Hình 38 Sử dụng toán tử chọn lọc mới: vòng xe



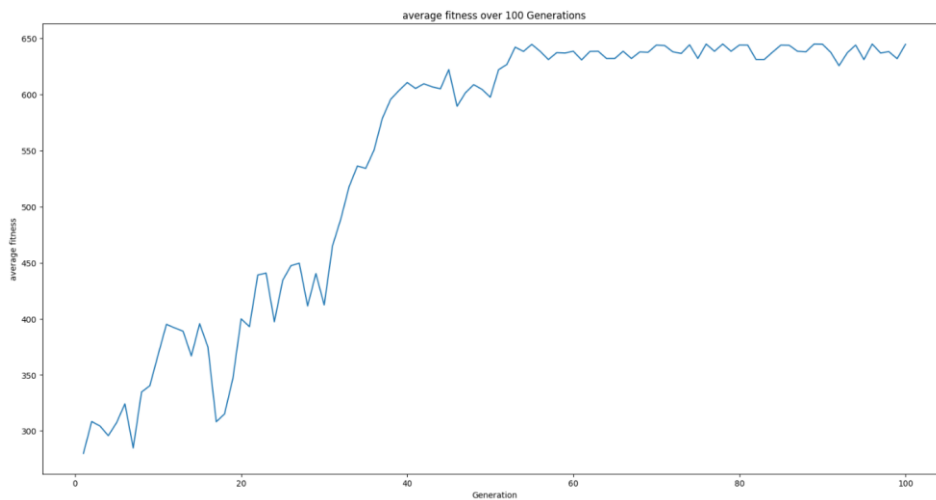
Hình 39 Độ phù hợp trung bình của quần thể khi sử dụng toán tử chọn lọc mới: vòng xe

Nhận xét: Ta thấy rằng nếu sử dụng toán tử chọn lọc kiểu vòng xe làm cho kết quả của bài toán trở nên phần nào đó kém hiệu quả hơn, ta có thể quan sát thấy rằng cá thể có độ phù hợp cao nhất của quần thể trong mỗi thế hệ có sự thay đổi phù hợp hơn ở các thế hệ đầu

nhưng có sự thoái hóa trong những thế hệ sau và dần ổn định từ các thế hệ này. Bên cạnh đó đồ thị thể hiện độ phù hợp trung bình cũng chỉ ra một xu hướng tăng lên về sự phù hợp trong toàn bộ quần thể, tuy nhiên ta thấy có sự hội tụ tới một kết quả chưa được tốt, bên cạnh đó có sự dao động tương đối nhiều so với phương pháp chọn được nhóm sử dụng mặc định (phương pháp chọn kiểu đầu trường).



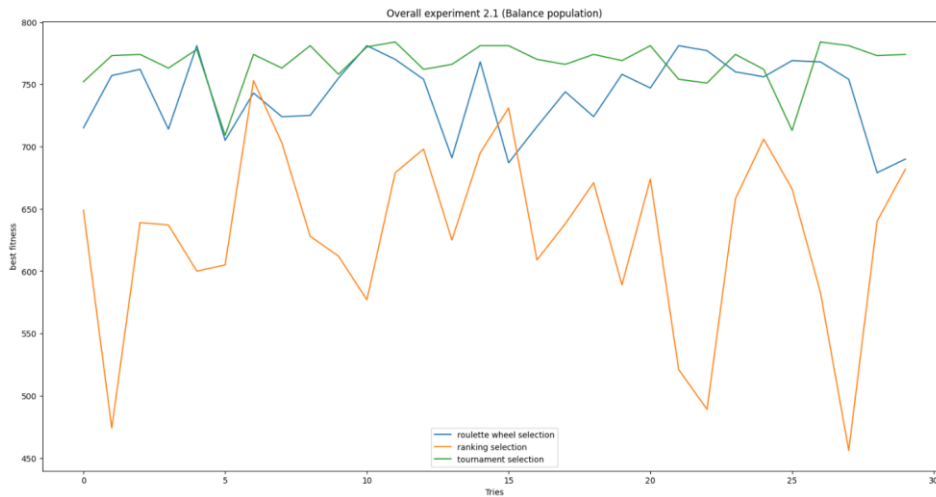
Hình 40 Khi sử dụng toán tử chọn lọc xếp hạng



Hình 41 Trung bình độ phù hợp của toàn bộ quần thể khi sử dụng toán tử chọn lọc xếp hạng

Nhận xét: Ta có thể thấy tình trạng tương tự cũng xuất hiện khi chúng ta sử dụng chọn lọc theo kiểu xếp hạng. Hơn nữa, hành vi của giải thuật có sự bất thường khi chúng ta xem xét

đồ thị về trung bình phù hợp của toàn bộ quần thể, đồ thị mất một lượng thể hệ nhất định mới có thể đạt tới độ phù hợp cao hơn (điểm cao hơn này cũng không hội tụ quanh độ phù hợp mà nhóm cho là hợp lí (760)). Có thể nói hàm chọn theo kiểu xếp hạng này không được tốt cho dữ liệu đang có phục vụ bài toán. Về phần dữ liệu sẽ được thực nghiệm ở phần sau.

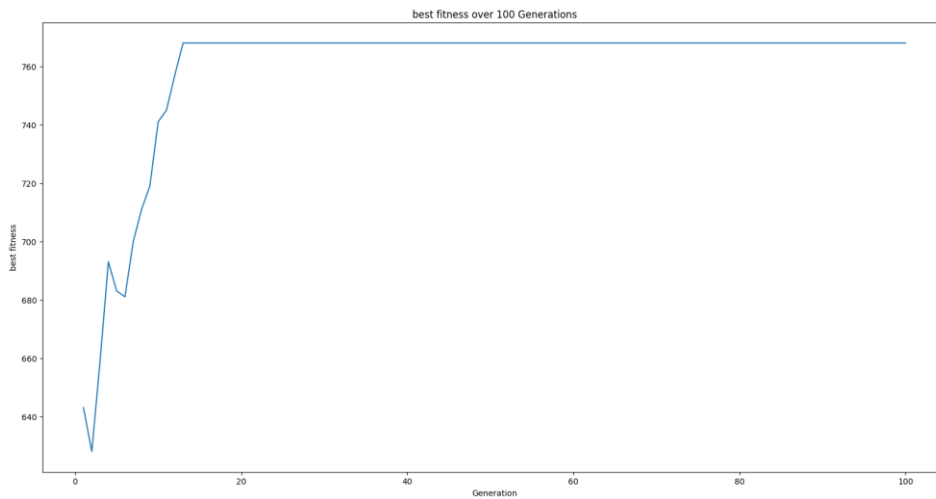


Hình 42 Tổng kết thí nghiệm 2.a

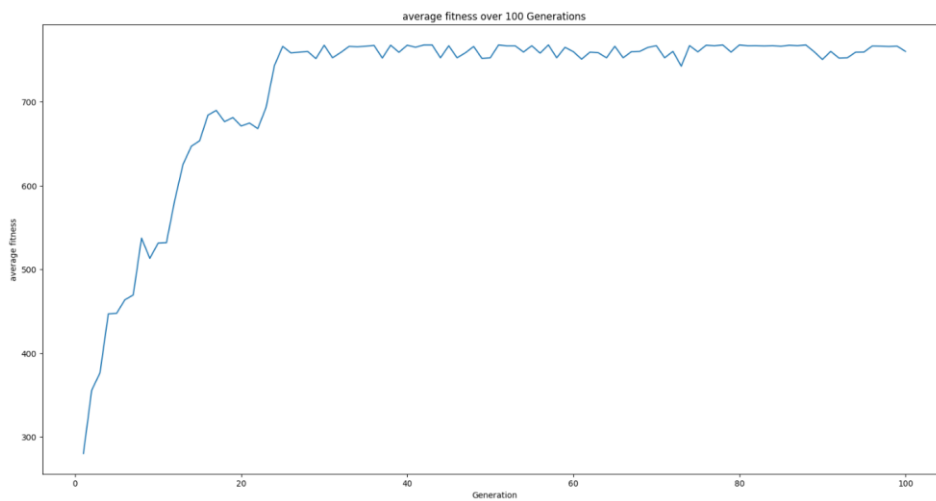
Tổng kết thí nghiệm 2.a: Qua thực nghiệm, nhóm nhận thấy với các thông số tiêu chuẩn, thì hàm chọn theo kiểu đấu trường nhìn chung cho ra kết quả tốt hơn so với hàm chọn theo kiểu bánh xe roulette, và tốt hơn nhiều so với hàm chọn theo kiểu xếp hạng. Lưu ý các đường trên đồ thị được vẽ từ các cá thể có độ phù hợp cao nhất ở cá thể cuối cùng của thể hệ cuối cùng. Bên cạnh đó cũng lưu ý rằng đối với các bài toán khác, có thể toán tử chọn lọc kiểu đấu trường lại không tốt bằng những cách khác nên tùy vào bài toán mà ta nên cân nhắc cân trọng nên sử dụng toán tử đột chọn lọc nào.

b. Thay đổi toán tử lai tạo.

Các toán tử lai tạo đóng vai trò quan trọng trong việc điều chỉnh kết quả của giải thuật di truyền, nhóm sẽ chỉ thực hiện thay đổi toán tử lai tạo hiện tại (lai tạo một điểm) thành lai tạo 2 điểm.



Hình 43 Sử dụng toán tử lai tạo 2 điểm

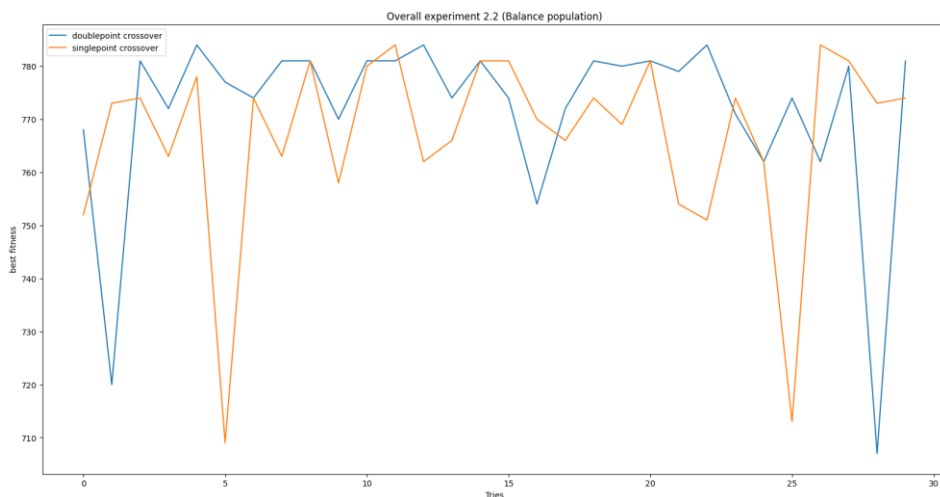


Hình 44 Trung bình độ phù hợp của quần thể khi sử dụng lai tạo 2 điểm.

After 12 generations, we reached fitness's threshold (750)

Hình 45 Tốc độ đạt tới điểm mong muốn của giải thuật khi sử dụng lai tạo 2 điểm.

Nhận xét: Về cơ bản, các tác động của việc sử dụng lai tạo 2 điểm lên hành vi của thuật toán không thay đổi nhiều so với việc sử dụng lai tạo 1 điểm, sự khác biệt đáng lưu tâm có thể nằm ở việc kết quả tốt nhất trong dài hạn (sẽ đề cập ở dưới).

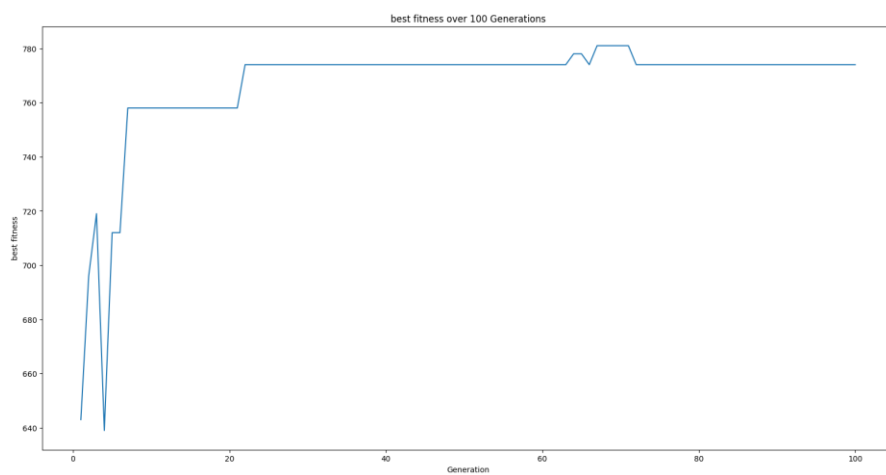


Hình 46 Tổng kết thí nghiệm 2.b

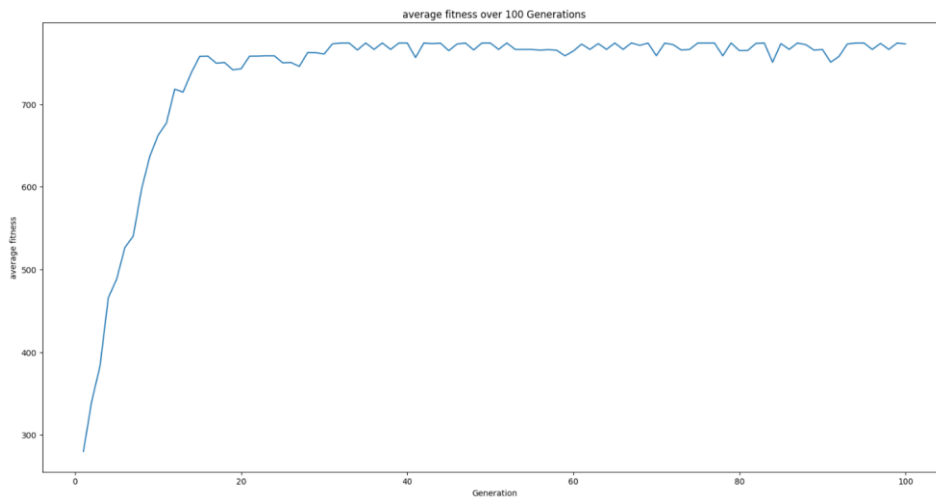
Tổng kết thí nghiệm 2.b: Việc chúng ta thay đổi toán tử lai tạo làm cho kết quả của giải thuật di truyền trở nên tốt hơn, bên cạnh đó việc sử dụng lai tạo 2 điểm thay cho lai tạo 1 điểm vẫn cho ra các kết quả phù hợp và đáng chấp nhận, bên cạnh đó tốc độ hội tụ của việc sử dụng lai tạo 2 điểm cũng nhanh. Không chỉ vậy, sự dao động của các kết quả tốt nhất của việc sử dụng lai tạo 2 điểm cũng tương đối ổn định so với việc sử dụng lai tạo 1 điểm.

c. Thay đổi toán tử đột biến.

Ở đây nhóm sử dụng toán tử đột biến mới: scramble mutation. Toán tử đột biến này thay đổi vị trí gen ngẫu nhiên trong đoạn gen trong một nhiễm sắc thể.



Hình 47 Sử dụng toán tử đột biến kiểu scramble

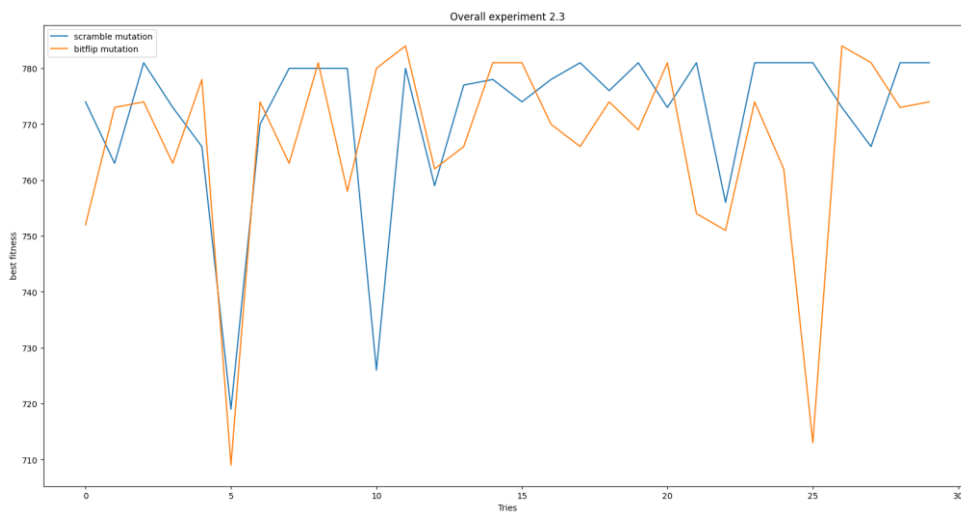


Hình 48 Trung bình độ phù hợp khi sử dụng đột biến kiểu scramble

After 7 generations, we reached fitness's threshold (750)

Hình 49 Tốc độ đạt tới độ phù hợp tốt khi sử dụng đột biến kiểu scramble

Nhận xét: Việc sử dụng đột biến kiểu scramble cũng không có quá nhiều khác biệt (về mặt kết quả) so với đột biến kiểu bit-flip.



Hình 50 Tổng kết thí nghiệm 2.c

Tổng kết thí nghiệm 2.c: Không có nhiều sự khác biệt giữa đột biến kiểu bit-flip và đột biến kiểu scramble, bởi vì nhìn chung, khả năng diễn ra sự đột biến là tương đối thấp. Chúng ta có thể sử dụng kiểu đột biến nào cũng được.

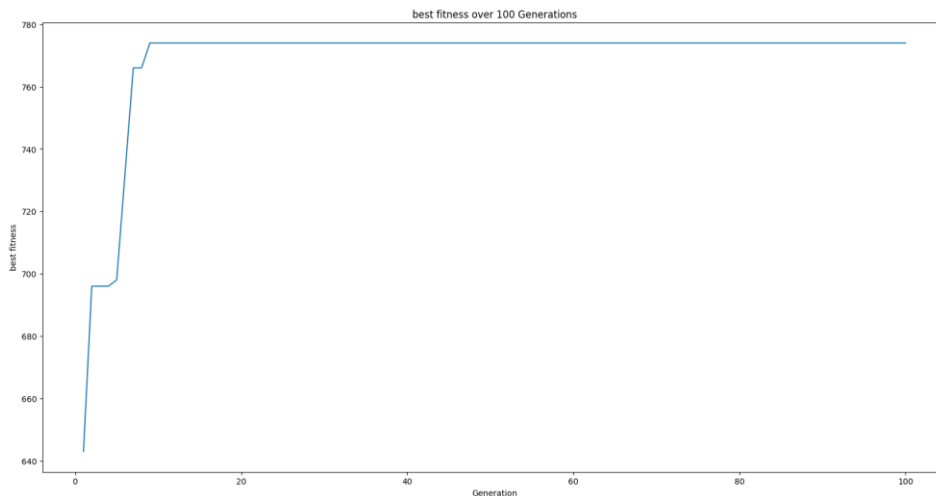
3. Bảo toàn tính trạng tốt của quần thể.

Như các ví dụ ở trên, ta bắt gặp những trường hợp mà ở đó quần thể có sự thoái hóa về độ phù hợp, và điều này là không nên xảy ra trong giải thuật di truyền khi việc này làm giảm đi tính trạng vốn đã tốt của quần thể.

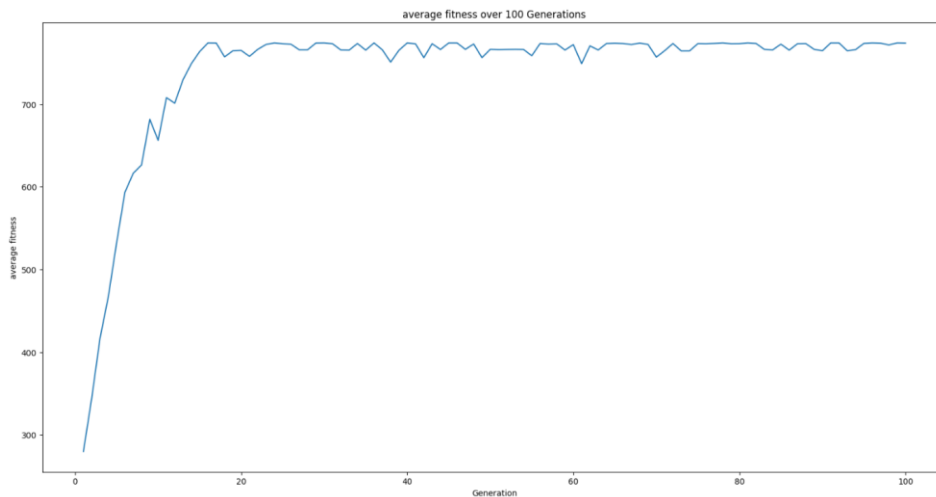
Nhóm tiến hành thực nghiệm các biện pháp khác nhau để xác định nên sử dụng phương pháp nào để các tính trạng tốt nhất của quần thể được giữ lại. 2 hướng tiếp cận mà nhóm sử dụng là: elitism, tiêu chí dừng delta.

a. Hướng tiếp cận elitism.

Hướng tiếp cận này nhằm bảo tồn các cá thể tốt nhất của từng quần thể bằng cách xác định xem cá thể nào đang có độ phù hợp cao nhất và trực tiếp đặt cá thể đó vào thế hệ tiếp theo (như kiểu bảo tồn gen tốt trong sinh học). Về mặt ý tưởng, hướng tiếp cận elitism cần xác định số cá thể ta mong muốn bỏ vào quần thể của thế hệ tiếp theo (nhóm sẽ sử dụng 2 cá thể) và sau đó, các cá thể tốt nhất của các quần thể sau sẽ luôn có độ phù hợp lớn hơn hoặc bằng cá thể tốt nhất của quần thể trước nó.



Hình 51 Sử dụng hướng tiếp cận elitism để hạn chế sự thoái hóa.



Hình 52 Độ phù hợp trung bình qua các thế hệ khi sử dụng hướng tiếp cận elitism

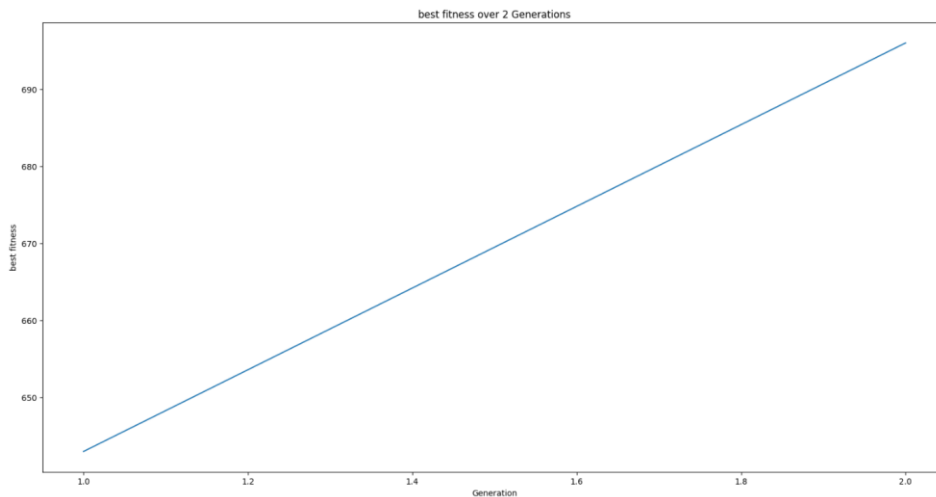
After 7 generations, we reached fitness's threshold (750)

Hình 53 Tốc độ đạt tới một kết quả tốt của giải thuật sử dụng hướng tiếp cận elitism

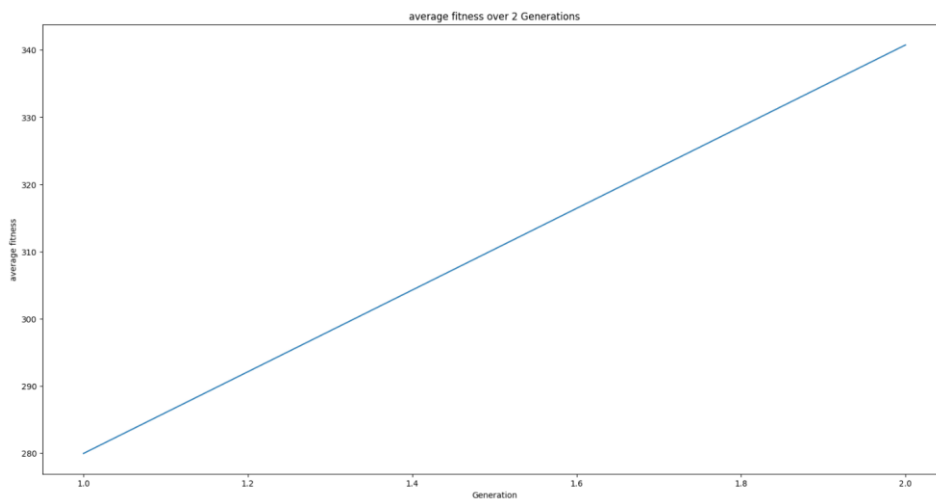
Nhận xét: Về mặt ý tưởng, hướng tiếp cận elitism có thể đảm bảo cá thể tốt nhất của cá thể sau chắc chắn sẽ tốt hơn hoặc bằng cá thể tốt nhất của các thế hệ trước đó. Tuy nhiên, việc này cũng không chắc chắn sẽ khiến cho kết quả của thuật toán sẽ đạt được một mức phù hợp mong muốn. Và cũng như các trường hợp khác, việc sử dụng hướng tiếp cận này cũng có xu hướng làm cho độ phù hợp trung bình của các quần thể sau cao hơn các quần thể đầu, bên cạnh đó độ dao động giá trị phù hợp trung bình ở các quần thể ở các thế hệ sau có vẻ như sẽ hội tụ tới một độ phù hợp nào đó tạm chấp nhận được.

b. Hướng tiếp cận theo tiêu chí dừng delta.

Việc sử dụng tiêu chí dừng delta, đơn giản là nếu như cá thể tốt nhất của cá thể sau có độ phù hợp thấp hơn cá thể tốt nhất một khoảng bằng delta thì chúng ta có thể xem là tiêu chí dừng đã được đáp ứng, từ đó cho dừng thuật toán và sẽ lấy cá thể tốt nhất của quần thể trước đó làm kết quả cho bài toán knapsack.



Hình 54 Áp dụng tiêu chí dừng delta cho giải thuật di truyền



Hình 55 Độ phù hợp trung bình khi sử dụng hướng tiếp cận delta

Can't reach the limit threshold (750)

Hình 56 Khi sử dụng hướng tiếp cận delta có khả năng khiến cho kết quả không đạt được như mong muốn

Nhận xét: Tiêu chí dừng kiểu delta có một nhược điểm đó là khoảng delta đó không được quá cao bởi vì nếu quá cao thì sẽ khó đáp ứng nhưng cũng không được quá thấp bởi vì nếu quá thấp thì tiêu chí dừng sẽ dễ dàng được thực hiện. Như trong hình ở trên, nhóm sử dụng delta bằng 30. Điều này làm cho điểm mạnh của thuật toán trở thành điểm yếu của nó. Và

cũng vì lí do này, chúng ta cần phải xác định khoảng delta sao cho hợp lí bằng cách thực nghiệm nhiều lần để xác định hành vi của giải thuật trước khi xác định số delta.

Tổng kết thí nghiệm 3 : Hướng tiếp cận kiểu elitism có thể hạn chế được sự suy giảm độ phù hợp của cá thể tốt nhất trong mỗi thế hệ, và thường thì hướng tiếp cận này cho ra các kết quả tương đối tốt, còn về hướng tiếp cận kiểu delta thì có thể hạn chế được việc độ phù hợp giảm xuống quá nhiều, tuy nhiên điều này sẽ chỉ xảy ra nếu xác định được một số delta hợp lí. Do vậy nhóm khuyến nghị nên sử dụng hướng tiếp cận kiểu elitism để giải quyết các bài toán dạng này.

4. Ảnh hưởng của dữ liệu.

Việc sử dụng giải thuật di truyền để giải quyết bài toán knapsack, bên cạnh các siêu tham số cần được xác định và các toán tử di truyền cũng như các tiêu chí dừng và cách tiếp cận khác nhau, dữ liệu (cân nặng và giá trị của mỗi vật phẩm và sức chứa của balo) cũng đóng một vai trò quan trọng.

Nhóm thực nghiệm trên 2 trường hợp chính, một trường hợp dễ đáp ứng và một trường hợp khó đáp ứng.

a. Dễ đáp ứng.

Trường hợp dễ đáp ứng là trường hợp các vật phẩm có giá trị và cân nặng như nhau (phương sai bằng 0) và sức chứa của balo là rất lớn.

```
1 values = [15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15]
2 weights = [5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5]
3 capacity = int(sum(weights) * 0.9)
4 print("values: ", values)
5 print("weights: ", weights)
6 print("capacity: ", capacity)

values: [15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15]
weights: [5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5]
capacity: 90
```

Hình 57 Dữ liệu cho bài toán dễ.

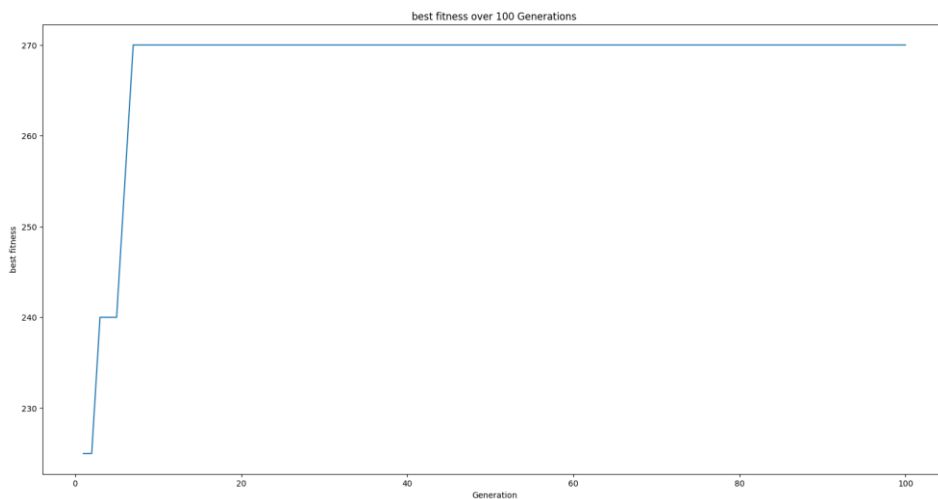
Lúc này, kết quả chính xác của bài toán là như sau:

```
1 correct_ans, best_fitness, total_weight = knapsack_brute_force(weights=weights, values=values, capacity=capacity)
2
3 print("The exact solutions for knapsack is: ", correct_ans)
4 print("Maximum values of knapsack is: ", best_fitness)
5 print("Total weight for the best solution is: ", total_weight)

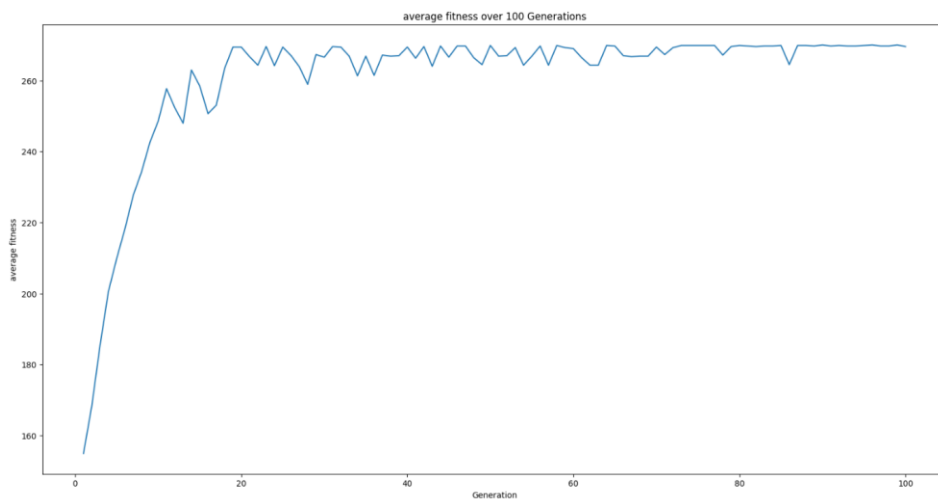
The exact solutions for knapsack is: [0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
Maximum values of knapsack is: 270
Total weight for the best solution is: 90
```

Hình 58 Kết quả chính xác của bài toán dễ

Với trường hợp này, giải thuật di truyền luôn cho ra kết quả chính xác.

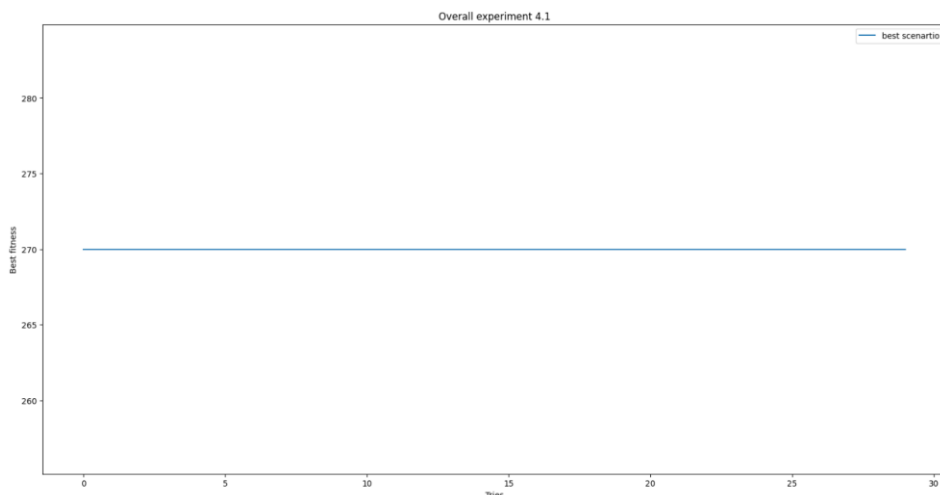


Hình 59 Kết quả của giải thuật di truyền để giải quyết các vấn đề dễ.



Hình 60 Trung bình độ phù hợp của giải thuật di truyền đối với các bài toán dễ.

Nhận xét: Với thí nghiệm này, với tập dữ liệu như trên, bài toán trở nên dễ đáp ứng, lúc này các kết quả của giải thuật di truyền cho ra ở các thế hệ sau luôn là kết quả chính xác. Và ở đồ thị độ phù hợp của cá thể tốt nhất của mỗi quần thể của mỗi thế hệ, không có sự thoái hóa. Có thể kết luận rằng với các bài toán dễ đáp ứng như bài này, giải thuật di truyền có thể hoàn thành tốt.



Tổng kết thí nghiệm 4.a: Sau 30 lần thử, các kết quả tốt nhất của quần thể trong thế hệ cuối cùng là như nhau và độ phù hợp bằng với độ phù hợp của kết quả chính xác. Có thể nói với dữ liệu bài toán như thế này, giải thuật di truyền luôn hội tụ tới một kết quả chính xác (trong trường hợp các siêu tham số và toán tử di truyền được thiết kế hợp lý).

b. Khó đáp ứng:

Đối với trường hợp khó đáp ứng, dữ liệu có phương sai cao và sức chứa của balo là không cao.

```

✖ Weights: [82, 15, 4, 95, 36, 32, 29, 18, 95, 14, 87, 95, 70, 12, 76, 55, 5, 4, 12, 28]
  Values: [30, 65, 78, 4, 72, 26, 92, 84, 90, 70, 54, 29, 58, 76, 36, 1, 98, 21, 90, 55]
  Capacity: 259
  Weights Variance: 1138.96
  Values Variance: 861.8475000000001

```

Hình 61 Dữ liệu của bài toán khó giải quyết

Lúc này, kết quả chính xác cho bài toán knapsack là như sau:

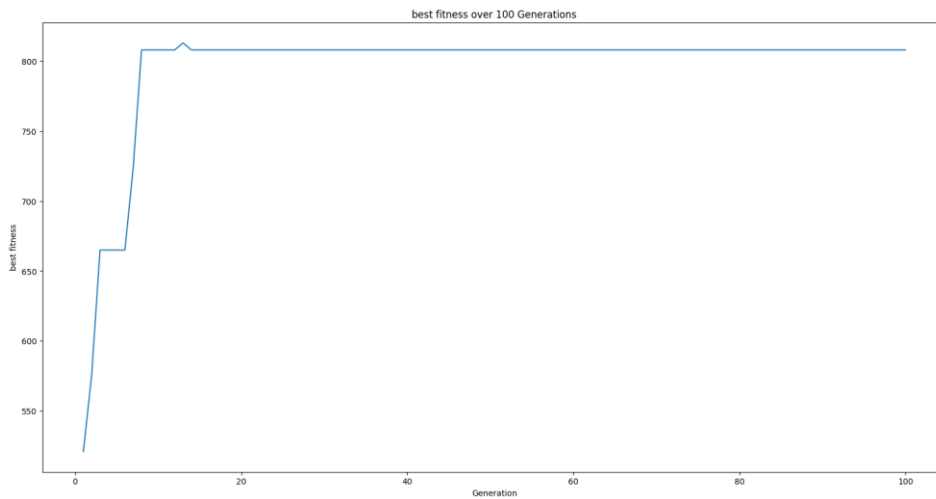
```

=====
The exact solutions for knapsack is: [0, 1, 1, 0, 1, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1]
Maximum values of knapsack is: 859
Maximum weight of knapsack is: 259
=====

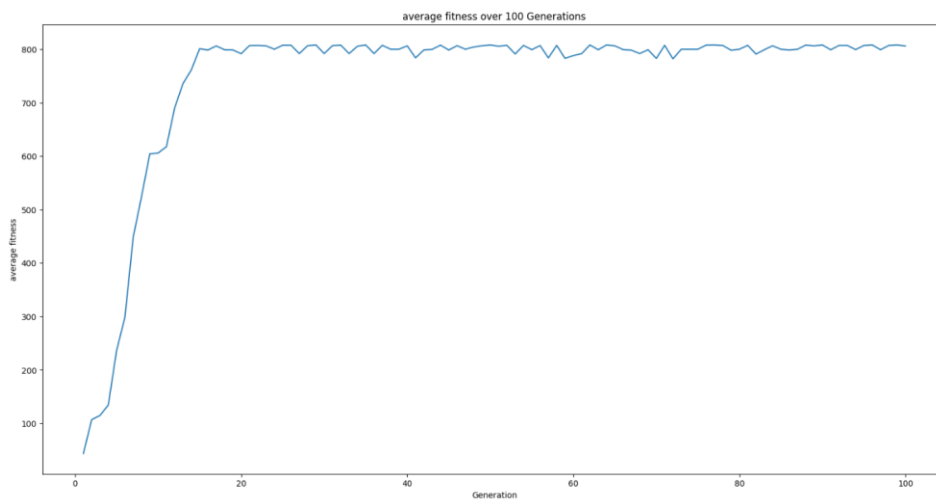
```

Hình 62 Kết quả chính xác của bài toán khó giải quyết

Bây giờ chúng ta quan tâm đến hành vi của giải thuật di truyền trong trường hợp này:

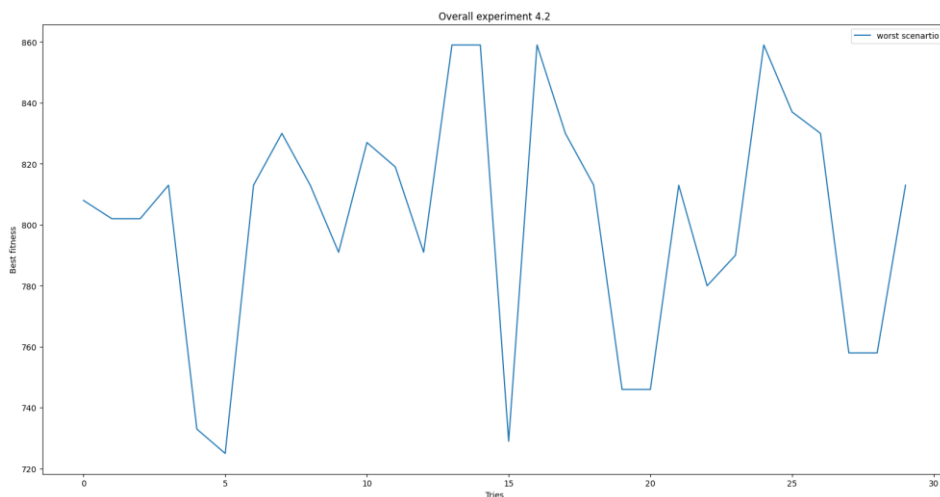


Hình 63 Kết quả của giải thuật di truyền đối với bài toán khó giải quyết.



Hình 64 Độ phù hợp trung bình của giải thuật di truyền với bài toán khó giải quyết.

Nhận xét: Nhìn chung thì việc chúng ta có một bộ dữ liệu có phương sai lớn cũng không ảnh hưởng quá nhiều đến hành vi của giải thuật di truyền (mặc dù các kết quả chưa được tốt lắm).



Hình 65 Tổng kết thí nghiệm 4.b

Tổng kết thí nghiệm 4.b: Ta có thể thấy với phương sai cao làm cho kết quả tốt nhất của thế hệ cuối cùng có sự dao động nhiều, không như với các trường hợp dễ đáp ứng. Như vậy có thể thấy các thông tin đầu vào cũng đóng vai trò quan trọng trong việc xây dựng kết quả của giải thuật.

CHƯƠNG 4. KẾT LUẬN

4.1. Các Kết Quả Đạt Được:

Giải thuật di truyền là một phương pháp tối ưu áp dụng phương pháp tìm kiếm ngẫu nhiên (stochastic search) để tìm ra lời giải tối ưu. Giải thuật di truyền có điểm mạnh như:

- Có khả năng tìm được các lời giải tốt thậm chí ngay cả với các không gian tìm kiếm (lời giải) không liên tục rất phức tạp.
- Giải được nhiều bài toán: Hầu hết các vấn đề trong thực tế đều có thể áp dụng được giải thuật di truyền để giải quyết các bài toán đó ví dụ như trong các lĩnh vực kinh tế, công nghiệp, y học, khoa học máy tính, v.v.... Tuy nhiên để giải thuật hoạt động tốt khi ứng dụng vào thực tế, ta cần biểu diễn bài toán đó dưới dạng mà máy tính có thể hiểu được và đại diện tốt cho bài toán đó trong thực tế, bên cạnh đó ta cũng cần phải chú ý tới cách thiết kế hàm fitness sao cho phù hợp với bài toán ta đang giải.
- Tối ưu đa mục tiêu: Giải thuật di truyền có thể tối ưu hóa nhiều mục tiêu cùng một lúc. Nó có thể tìm kiếm những giải pháp làm giảm độ lệch giữa các mục tiêu, làm cho các giải pháp đó phù hợp với nhiều mục tiêu cùng lúc.
- Tìm kiếm toàn cục: Giải thuật di truyền phù hợp cho việc tìm kiếm các điểm tối ưu toàn cục hoặc các kết quả gần tối ưu. Bằng cách duy trì và phát triển quần thể lời giải, giải thuật di truyền có khả năng tìm kiếm toàn bộ không gian lời giải thay vì chịu cảnh mắc kẹt ở vùng tối ưu cục bộ (các giải thuật tối ưu sử dụng đạo hàm để cập nhật tham số thường vướng vào vấn đề này).

Bàn luận thêm về độ phức tạp của thuật toán: Đã có nhiều bài nghiên cứu bàn luận về độ phức tạp của giải thuật di truyền đối với nhiều bài toán khác nhau. Độ phức tạp của giải thuật là không cố định, do việc xác định độ phức tạp này còn phụ thuộc vào việc ta thiết kế hàm fitness như thế nào, biểu diễn các kết quả dưới dạng nào, và các yếu tố khác. Trong một bài nghiên cứu các phương pháp khác nhau để giải quyết bài toán knapsack 0/1 (Shaheen, 2016 #25), trong nghiên cứu này, độ phức tạp của giải thuật di truyền là $O(N)$, bởi vì việc xác định chuỗi nhiễm sắc thể có độ phức tạp $O(N)$, các hàm lai tạo, tính độ phù hợp, đột biến có độ phức tạp $O(N)$, hàm chọn lọc có độ phức tạp $O(1)$. Do vậy mà tổng độ phức tạp của thuật toán là $O(N)$. Dưới đây là bảng so sánh độ phức tạp về thời gian của các giải thuật được sử dụng trong nghiên cứu đó.

Metric	Greedy	DP	B&B	Genetic
Ex. Time	$O(N \log N)$	$O(W * N)$	$O(2^n)$	$O(N)$

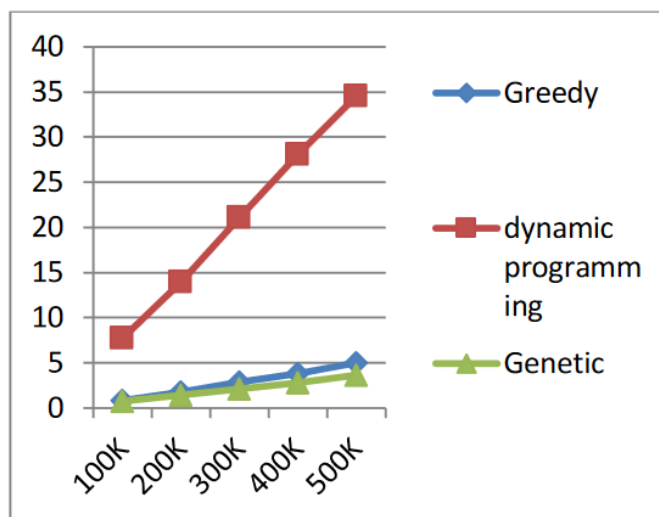
Hình 66 So sánh độ phức tạp về thời gian

Bên cạnh đó bài nghiên cứu này cũng đã sử dụng các thuật toán này để giải quyết bài toán với kích cỡ vật phẩm khác nhau:

Size	Greedy	DP	B&B	Genetic
100K	0.8623	7.7177	NA	0.7325
200K	1.7758	13.9441	NA	1.4669
200K	2.8489	21.0783	NA	2.1652
400K	3.8071	28.1098	NA	2.8336
500K	4.9852	34.5011	NA	3.6795

Hình 67 So sánh các thuật toán với số lượng vật phẩm khác nhau

Thí nghiệm này thử nghiệm các thuật toán khác nhau với nhiều vật phẩm khác nhau, 100K tức là 100,000 vật phẩm. Ta thấy giải thuật di truyền và giải thuật tham lam luôn cho ra kết quả tốt trong thời gian tương đối ngắn (và các kết quả tốt này là các kết quả chấp nhận được). Còn giải thuật Branch and Bound (B&B) có độ phức tạp 2^n nên là thiết bị không chạy được. Và dựa vào kết quả này, ta có thể thấy giải thuật cho ra thời gian chấp nhận được là giải thuật di truyền, tham lam, quy hoạch động. Trong đó giải thuật quy hoạch động luôn cho ra kết quả tối ưu toàn cục còn giải thuật di truyền và tham lam thì lại cho ra kết quả tối ưu cục bộ.



Hình 68 Thời gian thực thi chương trình của từng thuật toán

Bên cạnh đó bài nghiên cứu cũng so sánh kết quả của 3 giải thuật này với nhau để có thể xác định thuật toán nào cho ra kết quả tốt.

Data Size	Greedy	Genetic	Dynamic programming
100	379	387	387
200	496	510	512
300	682	682	682
400	761	771	771
500	805	816	816
600	881	887	887
700	952	945	952
800	952	952	952
900	1002	1002	1002
1000	1009	1009	1015

Hình 69 So sánh các kết quả của từng thuật toán

Ta có thể thấy kết quả cho ra bởi giải thuật di truyền và giải thuật tham lam đều có thể cho ra các kết quả tốt (có thể xấp xỉ kết quả chính xác) với ít thời gian hơn.

4.2. Những Hạn Chế và Hướng Phát Triển:

Bên cạnh các điểm mạnh được đề cập ở trên, các điểm mạnh này đôi khi cũng là con dao hai lưỡi, giải thuật di truyền cũng có một vài điểm yếu như sau:

- Chi phí tính toán cao, đặc biệt là đối với các vấn đề cỡ lớn. Nếu sử dụng để giải các bài toán thông thường như bài toán knapsack thì không phải vấn đề. Nhưng nếu sử dụng cho các bài toán có cỡ lớn (hàm fitness rất phức tạp và yêu cầu nhiều chi phí tính toán, đi kèm với việc quần thể có nhiều cá thể) thì gây ra chi phí tính toán cao, và điều này khiến cho giải thuật di truyền đôi lúc không còn phù hợp.
- Tình hình tham số: Như các thực nghiệm ở trên của nhóm về việc điều chỉnh tham số của giải thuật di truyền, ta có thể thấy các tham số nếu được thiết kế không phù hợp thì sẽ dẫn đến việc chi phí tính toán trở nên cao hơn (ví dụ như việc tăng kích thước quần thể hay là tăng số lượng thế hệ) hoặc là làm cho kết quả bài toán trở nên tệ hơn (tham số delta không phù hợp với vấn đề cần giải quyết hoặc là tỉ lệ đột biến trở nên quá cao làm hàm fitness trở nên ít hữu dụng).
- Do giải thuật di truyền là một phương pháp tìm kiếm ngẫu nhiên nên giải thuật này không chắc chắn sẽ tìm ra được một điểm tối ưu toàn cục (trừ trường hợp đó là bài toán dễ giải quyết như nhóm đã thực hiện ở trên). Do vậy với các trường hợp mà ta thật sự mong muốn điểm tối ưu toàn cục thì ta có thể sử dụng hướng tiếp cận brute-force.

Bên cạnh các việc sử dụng giải thuật di truyền để giải quyết các bài toán khác trong thực tế, ta cũng có thể sử dụng các phương pháp tối ưu khác như tối ưu dựa trên đạo hàm, simulated annealing, v.v.... Nói chung việc sử dụng phương pháp nào cũng còn dựa trên vấn đề mà bài toán cần giải quyết.

Hướng phát triển với giải thuật di truyền: Giải thuật di truyền là một giải thuật có độ ứng dụng cao và có nhiều hướng phát triển. Như đã nhận định ở mục thí nghiệm, việc chúng ta

cải thiện các siêu tham số, thay đổi các toán tử di truyền, kết hợp nhiều phương pháp mới cũng làm cho kết quả của giải thuật trở nên chuẩn xác hơn. Ngoài việc đó, ta cũng có thể cẩn thận thiết kế hàm fitness, việc tiếp cận với hàm fitness một cách sơ xài mà không cân nhắc kĩ cũng vô tình làm tăng thời gian của giải thuật và làm giảm độ chính xác của kết quả. Và do giải thuật di truyền là một thuật toán linh hoạt, ta có thể kết hợp giải thuật di truyền với các thuật toán tối ưu khác.

Các biến thể của giải thuật di truyền để cải tiến kết quả cho vấn đề knapsack cũng có thể hỗ trợ ta, các biến thể hữu dụng cho bài toán này. Sử dụng kỹ thuật tiến hóa phân tầng: Kỹ thuật tiến hóa phân tầng có thể được sử dụng để tìm kiếm các giải pháp tối ưu cho bài toán knapsack. Kỹ thuật này giúp chia bài toán thành các bài toán con nhỏ hơn, tìm kiếm giải pháp tối ưu cho từng bài toán con, sau đó kết hợp các giải pháp con để tạo ra giải pháp tối ưu cho bài toán ban đầu, các kỹ thuật có thể kể đến như tiến hóa phân tầng (Hierarchical Evolution), tiến hóa đa cấp (Multi-Level evolution), tiến hóa động (Dynamic Evolution).

4.3 Các ứng dụng khác trong thực tế:

Giải thuật di truyền (hoặc các hướng tiếp cận khác thuộc nhóm giải thuật tiến hóa) cho ra các kết quả tốt và có thể xấp xỉ chính xác các bài toán khó cho dù vùng tìm kiếm có phức tạp. Nhờ lí do đó mà giải thuật di truyền thường được sử dụng nhiều trong các bài toán thực tế.

- Trong lĩnh vực khoa học máy tính:
 - o Sử dụng để giải quyết vấn đề dữ liệu mất cân bằng. (Karia, V., Zhang, W., Naeim, A., & Ramezani, R., 2019)
 - o Sử dụng giải thuật di truyền để tối ưu các tham số trong mô hình. (Wang, 1997)
 - o V.v...
- Trong thị trường tài chính:
 - o Sử dụng để tìm ra bộ tham số tối ưu ảnh hưởng đến những người giao dịch trên thị trường. (Jiří Štěpánek, Jiří Šťovíček, Richard Cimler, 2012)
 - o Dự đoán chỉ số chứng khoán. (R. Lakshman Naik, D. Ramesh, B. Manjula, Dr. A. Govardhan, 2012)
 - o V.v...
- Trong lĩnh vực y khoa:
 - o Sử dụng giải thuật di truyền để tối ưu các kế hoạch về sức khỏe hiện tại và sức khỏe trong tương lai. (Satoshi Sasaki, Alexis J. Comber, Hiroshi Suzuki, & Chris Brunsdon, 2010)
 - o Sử dụng giải thuật di truyền để dự đoán các đối tượng bệnh lý. (Murat Sari & Can Tuna, 2018)
 - o V.v...
- Trong lĩnh vực kinh doanh:
 - o Khám phá quy tắc ra quyết định của các chuyên gia từ dữ liệu phá sản định tính bằng giải thuật di truyền. (Myoung-Jong Kim & Ingo Han, 2003)

- Sử dụng giải thuật di truyền để định giá bất động sản. (Vincenzo Del Giudice, Pierfrancesco De Paola, & Fabiana Forte, 2017)

TÀI LIỆU THAM KHẢO

- Dantzig, T. &. (n.d.). Number: The language of science.
- Darwin, C. (1859). *On the origins of species*. Routledge.
- Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. The MIT Press.
Retrieved from <https://mitpress.mit.edu/9780262581110/adaptation-in-natural-and-artificial-systems/>
- Jiří Štěpánek, Jiří Šťovíček, Richard Cimler. (2012). Application of Genetic Algorithms in Stock Market Simulation. *Procedia - Social and Behavioral Sciences*, 47, 93-97. doi:<https://www.sciencedirect.com/science/article/pii/S1877042812023555>
- Karia, V., Zhang, W., Naeim, A., & Ramezani, R. (2019). GenSample: A Genetic Algorithm for Oversampling in Imbalanced Datasets. *GenSample: A Genetic Algorithm for Oversampling in Imbalanced Datasets*. Retrieved 2019, from <https://arxiv.org/pdf/1910.10806.pdf>
- Kellerer, H., Pfersch, U., Pisinger, D. (2004). Introduction to NP-Completeness of Knapsack Problems. In: Knapsack Problems. *Knapsack Problems*, 483-493. doi:https://doi.org/10.1007/978-3-540-24777-7_16
- Mathews, G. B. (1896). On the Partition of Numbers. *Proceedings of the London Mathematical Society*, s1-28(1), 489-490. Retrieved from <https://londmathsoc.onlinelibrary.wiley.com/doi/abs/10.1112/plms/s1-28.1.486>
- P.W, D. (2014, 11 20). Retrieved from cornell.edu:
<https://people.orie.cornell.edu/dpw/orie6300/Lectures/lec25.pdf>
- R. Lakshman Naik, D. Ramesh, B. Manjula, Dr. A. Govardhan. (2012). Prediction of Stock Market Index Using Genetic Algorithm. *Computer Engineering and Intelligent Systems*, 3, www.iiste.org. Retrieved from <https://core.ac.uk/download/pdf/234644513.pdf>
- Sasaki, Satoshi. (n.d.).
- Wang, Q. (1997). Using genetic algorithms to optimise model parameters. *Environmental Modelling & Software*, 12(1), 27-34. doi:[https://doi.org/10.1016/S1364-8152\(96\)00030-8](https://doi.org/10.1016/S1364-8152(96)00030-8)
- Aytug, Haldun, & Koehler, Gary J. (2000). New stopping criterion for genetic algorithms. *European Journal of Operational Research*, 126(3), 662-674. doi:[https://doi.org/10.1016/S0377-2217\(99\)00321-5](https://doi.org/10.1016/S0377-2217(99)00321-5)

- Chen, Tianshi, Tang, Ke, Chen, Guoliang, & Yao, Xin. (2012). A large population size can be unhelpful in evolutionary algorithms. *Theoretical Computer Science*, 436, 54-70. doi:<https://doi.org/10.1016/j.tcs.2011.02.016>
- DeJong, KA %J Computer, & Communication Sciences, University of Michigan, Ann Arbor. (1975). Analysis of the behavior of a class of genetic adaptive systems. dept.
- Del Giudice, Vincenzo, De Paola, Pierfrancesco, & Forte, Fabiana (2017). Using genetic algorithms for real estate appraisals. 7(2), 31.
- Frederick, William, Sedlmeyer, Robert, & White, Curt. (1993). *The Hamming Metric in Genetic Algorithms and Its Application to Two Network Problems*.
- Gotshall, Stanley, & Rylander, Bart %J Population. (2002). Optimal population size and the genetic algorithm. 100(400), 900.
- Kim, Myoung-Jong, & Han, Ingo (2003). The discovery of experts' decision rules from qualitative bankruptcy data using genetic algorithms. 25(4), 637-646.
- Lata, Saneh, Yadav, Saneh, & Sohal, Asha. (2017). Comparative Study of Different Selection Techniques in Genetic Algorithm. *International Journal of Engineering Science*.
- Lobo, Fernando G., & Lima, Cláudio F. (2005). *A review of adaptive population sizing schemes in genetic algorithms*. Paper presented at the Proceedings of the 7th annual workshop on Genetic and evolutionary computation, Washington, D.C. <https://doi.org/10.1145/1102256.1102310>
- Piszcz, Alan, & Soule, Terence. (2006). *A survey of mutation techniques in genetic programming* (Vol. 1).
- Reeves, Colin. (1998). Using Genetic Algorithms With Small Populations.
- Sari, Murat, & Tuna, Can. (2018). Prediction of Pathological Subjects Using Genetic Algorithms. *Computational and Mathematical Methods in Medicine*, 2018, 6154025. doi:10.1155/2018/6154025
- Sasaki, Satoshi, Comber, Alexis J., Suzuki, Hiroshi, & Brunsdon, Chris. (2010). Using genetic algorithms to optimise current and future health planning - the example of ambulance locations. *International Journal of Health Geographics*, 9(1), 4. doi:10.1186/1476-072X-9-4
- Srinivas, M., & Patnaik, L. M. (1994). Adaptive probabilities of crossover and mutation in genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics*, 24(4), 656-667. doi:10.1109/21.286385
- Umbarkar, Anant J, & Sheth, Pranali D %J ICTACT journal on soft computing. (2015). Crossover operators in genetic algorithms: a review. 6(1).
- Vrajitoru, Dana. (2000). Large Population or Many Generations for Genetic Algorithms? Implications in Information Retrieval. In Fabio Crestani & Gabriella Pasi (Eds.), *Soft Computing in Information Retrieval: Techniques and Applications* (pp. 199-222). Heidelberg: Physica-Verlag HD.
- Yao, Xin. (1993). An empirical study of genetic operators in genetic algorithms. *Microprocessing and Microprogramming*, 38(1), 707-714. doi:[https://doi.org/10.1016/0165-6074\(93\)90215-7](https://doi.org/10.1016/0165-6074(93)90215-7)
- Yee, Leung, Yong, Gao, & Zong-Ben, Xu. (1997). Degree of population diversity - a perspective on premature convergence in genetic algorithms and its Markov chain

- analysis. *IEEE Transactions on Neural Networks*, 8(5), 1165-1176. doi:10.1109/72.623217
- Zhu, Qi. (2018). Interior design and pattern innovation design study based on generic algorithm logistics information. *Cluster Computing*, 22. doi:10.1007/s10586-018-2080-8
- E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical re-sults," *Evolutionary Computation*, vol. 8, no. 2, pp. 173–195, 2000

PHỤ LỤC

Link mã nguồn: https://github.com/ngnquang/AI_Final.git