

# GIẢI BÀI TOÁN TỐI ƯU THEO THUẬT GIẢI DI TRUYỀN

ThS. Lê Thị Ngọc Hiếu<sup>1</sup>

## TÓM TẮT

*Dùng thuật giải di truyền để giải các bài toán tối ưu trong tin học là điều rất thú vị, bởi vì nguyên lý của nó là dựa vào qui luật đấu tranh sinh tồn trong tự nhiên. Bài viết này đưa ra phương pháp xây dựng thuật giải di truyền để giải bài toán tối ưu trong không gian vô cùng lớn, cùng với ví dụ minh họa là giải bài toán cấp phát trong cơ sở dữ liệu phân tán, được xem là bài toán tối ưu thuộc loại NP-hard.*

**Từ khóa:** bài toán tối ưu, thuật giải di truyền, bài toán cấp phát.

Thuật giải di truyền (*generic algorithms*) là một kỹ thuật của khoa học máy tính nhằm tìm kiếm giải pháp thích hợp cho các bài toán tối ưu tổ hợp (*combinatorial optimization*). Thuật giải di truyền cho phép chúng ta tạo ra những chương trình máy tính giúp giải quyết những vấn đề bằng cách mô phỏng sự tiến hóa của con người hay của sinh vật nói chung (dựa trên thuyết tiến hóa muôn loài của Darwin) trong điều kiện quy định sẵn của môi trường.

Ý tưởng của thuật giải di truyền để giải một bài toán tối ưu là tìm một tập hợp của những giải pháp, sau đó cho "tiến triển" theo hướng chọn lọc để tìm những giải pháp tốt dần hơn. Mục tiêu của thuật giải di truyền là đưa ra lời giải "tốt" có thể là tối ưu hay xấp xỉ tối ưu.

### 1. Thuật giải di truyền

Thuật giải di truyền cũng như các thuật toán tiến hóa nói chung, hình thành dựa trên quan niệm cho rằng quá trình tiến hóa tự nhiên là quá trình hoàn hảo nhất, hợp lý nhất và tự nó đã mang tính tối ưu. Quá trình tiến hóa tối ưu ở chỗ, thế hệ sau bao giờ cũng tốt hơn (phát triển hơn, hoàn thiện hơn) thế hệ trước. Tiến hóa tự nhiên được duy trì nhờ hai quá trình cơ bản: sinh sản và chọn lọc tự nhiên. Xuyên suốt quá trình tiến hóa tự nhiên, các thế hệ mới luôn được sinh ra để bổ sung thay thế thế hệ cũ. Cá thể nào phát triển hơn, thích ứng hơn với môi trường sẽ tồn tại. Cá thể nào không thích ứng được với môi trường sẽ bị đào thải. Sự thay đổi môi trường là động lực thúc đẩy quá trình tiến hóa. Ngược lại, tiến hóa cũng tác động trở lại góp phần làm thay đổi môi trường.

Các cá thể mới sinh ra trong quá trình tiến hóa nhờ sự lai ghép ở thế hệ cha mẹ. Một cá thể mới có thể mang những tính trạng của cha mẹ (di truyền), cũng có thể mang những tính trạng hoàn toàn mới (đột biến). Di truyền và đột biến là hai cơ chế có vai trò quan trọng như nhau trong tiến trình tiến hóa, dù rằng đột biến xảy ra với xác suất nhỏ hơn nhiều so với hiện tượng di truyền. Các thuật toán tiến hóa tuy có những điểm khác biệt nhưng đều mô phỏng bốn quá trình cơ bản: lai ghép, đột biến, sinh sản và chọn lọc tự nhiên.

#### 1.1. Định nghĩa thuật giải di truyền

Về mặt hình thức, thuật giải di truyền được định nghĩa là một bộ 7 (xem [1]):

---

<sup>1</sup>Trường Đại học Đồng Nai

$$GA=(I, \Psi, \Omega, s, t, \mu, \lambda)$$

Trong đó:

- (a)  $I = B^I$ ; không gian quần thể
- (b)  $\Psi: I \rightarrow \mathbf{R}^+$ ; ký hiệu hàm thích nghi (fitness) (độ thích nghi của một cá thể)
- (c)  $\Omega$ ; tập các phép toán di truyền (lai, đột biến, tái sinh)
- (d)  $s: I^{\mu+\lambda} \rightarrow I^\mu$ ; ký hiệu phép chọn, giữ lại  $\mu$  cá thể từ  $\mu+\lambda$  cá thể ban đầu
- (e)  $t: I^\mu \rightarrow \{true, false\}$ ; là tiêu chuẩn dừng
- (f)  $\mu$ ; số cá thể trong thế hệ cha mẹ
- (g)  $\lambda$ ; số cá thể trong thế hệ con cái

## 1.2. Các phương thức tiến hóa của thuật giải di truyền

### (a) Phương thức lai ghép (phép lai)

Phép lai là quá trình hình thành nhiễm sắc thể mới trên cơ sở các nhiễm sắc thể cha mẹ, bằng cách ghép một hay nhiều đoạn gen của hai (hay nhiều) nhiễm sắc thể cha mẹ với nhau. Phép lai xảy ra với xác suất  $p_c$ , có thể mô phỏng như sau:

- Chọn ngẫu nhiên hai (hay nhiều) cá thể bất kỳ trong quần thể. Giả sử các nhiễm sắc thể của cha mẹ đều có  $m$  gen.
- Tạo một số ngẫu nhiên trong khoảng từ 1 đến  $m-1$  (ta gọi là điểm lai). Điểm lai chia các chuỗi cha mẹ dài  $m$  thành hai nhóm chuỗi con dài  $m_1$  và  $m_2$ . Hai chuỗi nhiễm sắc thể con mới sẽ là  $m_{11}+m_{22}$  và  $m_{21}+m_{12}$ .
- Đưa hai cá thể mới này vào quần thể để tham gia các quá trình tiến hóa tiếp theo.

### (b) Phương thức đột biến (phép đột biến)

Đột biến là hiện tượng các thể con mang một số tính trạng không có trong mã di truyền của cha mẹ. Phép đột biến xảy ra với xác suất  $p_m$ , nhỏ hơn rất nhiều so với xác suất lai  $p_c$ . Phép đột biến có thể mô phỏng như sau:

- Chọn ngẫu nhiên một cá thể bất kỳ cha mẹ trong quần thể.
- Tạo một số ngẫu nhiên  $k$  trong khoảng từ 1 đến  $m$ ,  $1 \leq k \leq m$ .
- Thay đổi gen thứ  $k$  và trả cá thể này về quần thể để tham gia quá trình tiến hóa tiếp theo.

### (c) Phương thức sinh sản (phép tái sinh)

Phép tái sinh là quá trình trong đó các cá thể được sao chép trên cơ sở thích nghi của nó. Độ thích nghi là một hàm gán một giá trị thực cho các cá thể trong quần thể. Quá trình này có thể được mô phỏng như sau:

- Tính độ thích nghi của từng cá thể trong quần thể hiện hành, lập bảng cộng dồn các giá trị thích nghi (theo số thứ tự gán cho từng cá thể). Giả sử quần thể có  $n$

cá thể. Gọi độ thích nghi của cá thể thứ  $i$  là  $F_i$ , tổng dồn thứ  $i$  là  $F_{ti}$ , tổng độ thích nghi của toàn quần thể là  $F_m$ .

- Tạo một số ngẫu nhiên  $F$  trong đoạn từ 0 đến  $F_m$ .
- Chọn cá thể thứ  $k$  đầu tiên thỏa  $F \geq F_{tk}$  đưa vào quần thể của thế hệ mới.

**(d) Phương thức chọn lọc (phép chọn)**

Phép chọn là quá trình loại bỏ các cá thể xấu trong quần thể để chỉ giữ lại trong quần thể các cá thể tốt. Phép chọn có thể được mô phỏng như sau:

- Sắp xếp quần thể theo thứ tự độ thích nghi giảm dần.
- Loại bỏ các cá thể cuối dãy để chỉ giữ lại  $n$  cá thể tốt nhất. Ở đây, ta giả sử quần thể có kích thước cố định  $n$ .

### 1.3. Thuật giải di truyền tổng quát

Begin

$t:=0$ ;

Khởi tạo quần thể  $P(0):=\{a_1(0), a_2(0), \dots, a_\mu(0)\} \in I^\mu$

Tính độ thích nghi của các cá thể  $\Psi(a_1(0)), \Psi(a_2(0)), \dots, \Psi(a_\mu(0))$

While ( $t \neq true$ ) do

$t:=t+1$ ;

Tái sinh  $P(t)$  từ  $P(t-1)$ ;

Lai  $Q(t)$  từ  $P(t-1)$ ;

Đột biến  $R(t)$  từ  $P(t-1)$ ;

$P(t):= P(t-1) \cup Q(t) \cup R(t)$ ;

Tính độ thích nghi của các cá thể trong quần thể  $P(t)$ :

$\Psi(a_1(t)), \Psi(a_2(t)), \dots, \Psi(a_\mu(t)), \dots, \Psi(a_{\mu+\lambda}(t))$

Sắp xếp  $P(t)$  theo thứ tự  $\Psi(a_i(t))$  giảm dần;

Loại  $\lambda$  cá thể cuối cùng (giữ lại  $\mu$  cá thể tốt nhất);

End;

End;

### 1.4. Cơ chế thực hiện của thuật giải di truyền

Bước đầu tiên của quá trình tìm lời giải cho vấn đề là khởi tạo quần thể, nghĩa là phát sinh ra một số lượng lớn, hữu hạn các cá thể. Sau đó, dựa trên một hàm số nào đó gọi là hàm thích nghi để xác định giá trị độ thích nghi, giá trị này có thể hiểu là “độ tốt” của cá thể. Vì được phát sinh một cách ngẫu nhiên nên độ tốt của một lời giải hay tính thích nghi của một cá thể trong một quần thể là không xác định.

Chúng ta hãy tưởng tượng một máy tính giải quyết bài toán theo kiểu leo đồi, trong không gian tìm kiếm của bài toán là một vùng đất gập ghềnh, có nhiều ngọn đồi cao thấp khác nhau. Một người leo núi với tư tưởng càng leo càng cao (càng gần với độ tốt), nhưng có khả năng người leo núi sẽ bị kẹt ở một đỉnh đồi thấp. Như vậy, nếu có nhiều người cùng leo đồi ở nhiều địa điểm khác nhau thì có khả năng có một trong số các người này leo đến đỉnh đồi cao nhất sẽ cao hơn.

Đến đây chúng ta có thể sẽ nảy sinh ý tưởng: sử dụng nhiều thế hệ các người leo đồi. Nghĩa là toàn bộ những người leo đầu tiên (chẳng hạn 100 người) đều chưa gặp đỉnh đồi cao nhất thì sẽ cho 100 người khác tiếp tục leo. Nhưng có khả năng trong nhóm những người leo đồi mới, không ai trong số họ leo được những ngọn đồi cao hơn nhóm người trước. Hay nói một cách tổng quát: phải làm cách nào để giữ lại những người leo cao nhất cho thế hệ sau. Tiến trình cứ tiếp tục cho đến khi có một thế hệ nào đó, có một người leo đến đỉnh đồi cao nhất hoặc hết thời gian cho phép. Trong trường hợp hết thời gian thì trong toàn bộ các thế hệ, người nào leo cao nhất sẽ được chọn.

Như vậy, để cải thiện tính thích nghi của quần thể, người ta tìm cách tạo ra quần thể mới. Có hai thao tác thực hiện trên thế hệ hiện tại để tạo ra một thế hệ khác có độ thích nghi tốt hơn.

- Thao tác 1: sao chép nguyên mẫu một nhóm các cá thể tốt từ thế hệ trước rồi đưa sang thế hệ sau bằng phép tái sinh và chọn lọc. Thao tác này đảm bảo độ thích nghi của thế hệ sau luôn luôn tốt hơn hoặc bằng thế hệ trước.
- Thao tác 2: là tạo ra các cá thể mới bằng phép lai ghép và đột biến. Trong phép lai ghép, gen của hai cá thể tốt được chọn trong thế hệ trước sẽ được phối hợp với nhau (theo một qui tắc nào đó) để tạo ra hai cá thể mới. Phép đột biến là biến đổi ngẫu nhiên một hoặc nhiều thành phần gen của một cá thể ở thế hệ trước để tạo ra một cá thể hoàn toàn mới ở thế hệ sau. Phép đột biến có thể làm xáo trộn và làm mất đi tính thích nghi cao của cá thể được chọn.

## 2. Giải bài toán tối ưu bằng thuật giải di truyền

Để giải quyết một bài toán theo thuật giải di truyền chúng ta cần thực hiện các công việc sau:

- (a) Xem đối tượng cần tìm của bài toán như là 1 cá thể. Mô tả lớp đối tượng cần tìm theo một cấu trúc dữ liệu, biểu diễn không gian tìm kiếm của bài toán.
- (b) Xây dựng phương pháp khởi tạo quần thể ban đầu.
- (c) Định nghĩa hàm thích nghi, xác định độ thích nghi của các cá thể trong quần thể.
- (d) Xác định các phương thức tiến hóa của thuật giải di truyền như lai ghép, đột biến, tái sinh và chọn lọc.
- (e) Xác định các tham số cần thiết cho thuật giải như kích thước quần thể, xác suất lai, xác suất đột biến, số thế hệ tiến hóa, ...

(f) Xây dựng thuật giải.

Bước 1: Khởi tạo quần thể ban đầu.

Bước 2: Xác định độ thích nghi của các cá thể.

Bước 3: Thực hiện các phương thức lai ghép, đột biến, tái sinh và chọn lọc để sinh ra quần thể mới.

Bước 4: Tính độ thích nghi của các cá thể mới, loại bỏ đi các cá thể kém nhất, giữ lại một số nhất định các cá thể tốt.

Bước 5: Nếu chưa tìm được cá thể tốt (lời giải tối ưu) hay chưa hết số thế hệ tiến hoá (thời gian ấn định), quay lại bước 3.

Bước 6: Tìm được lời giải tối ưu hay thời gian cho phép đã hết thì kết thúc thuật giải và báo cáo kết quả tìm được.

### 3. Ví dụ minh họa: Bài toán cấp phát trong cơ sở dữ liệu phân tán

#### 3.1. Phát biểu bài toán:

Giả sử rằng có một tập các mảnh  $F = \{F_1, F_2, \dots, F_n\}$  và một mạng bao gồm các site (vị trí)  $S = \{S_1, S_2, \dots, S_m\}$  trên đó có một tập các ứng dụng  $Q = \{Q_1, Q_2, \dots, Q_q\}$ . Bài toán cấp phát là tìm một phân phối tối ưu của  $F$  cho  $S$  (xem [2]).

\* Tính tối ưu có thể được định nghĩa tương ứng với 2 điều kiện :

- Chi phí nhỏ nhất : Bao gồm chi phí lưu mỗi mảnh  $F_i$  tại vị trí  $S_j$ , chi phí vận tin  $F_i$  tại vị trí  $S_j$ , chi phí cập nhật  $F_i$  tại tất cả vị trí có chứa nó, và chi phí truyền dữ liệu. Vì thế bài toán cấp phát cố gắng tìm một lược đồ cấp phát với hàm chi phí tổ hợp thấp nhất.
- Hiệu năng : Hai chiến lược đã biết là hạ thấp thời gian đáp ứng và tăng tối đa lưu lượng hệ thống tại mỗi vị trí.

\* Ma trận FAM ( $n \times m$ ) thể hiện cách bố trí các mảnh vào site :

$$FAM_{ij} = \begin{cases} 1 & \text{nếu mảnh } F_i \text{ được cấp phát tại site } S_j \\ 0 & \text{trong trường hợp ngược lại} \end{cases}$$

Ví dụ ma trận FAM sau cho biết mảnh  $F_1$  được cấp phát tại site  $S_2$  và  $S_3$ .

$$\begin{array}{cccc} & S_1 & S_2 & S_3 & S_4 \\ \begin{array}{c} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \end{array} & \begin{array}{c} 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{array} & \begin{array}{c} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{array} & \begin{array}{c} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{array} & \begin{array}{c} 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{array} \end{array}$$

#### 3.2. Yêu cầu về thông tin :

- Thông tin về cơ sở dữ liệu:

Kích thước của mỗi mảnh (tính theo byte).

Ký hiệu  $SIZE(F_i)$  kích thước của mảnh  $F_i$ .

- Thông tin về giao tác :

Chúng ta có 2 ma trận thể hiện số lần truy vấn như sau :

\* Ma trận RM ( $q \times n$ ) (dòng là các truy vấn, cột là các mảnh) thể hiện số lần truy vấn chỉ đọc. Phần tử  $r_{ij}$  trong ma trận RM thể hiện số lần câu truy vấn  $q_i$  thực hiện chỉ đọc tại mảnh  $F_j$ .

\* Ma trận UM ( $q \times n$ ) (dòng là các truy vấn, cột là các mảnh) thể hiện số lần truy vấn cập nhật. Phần tử  $u_{ij}$  trong ma trận UM thể hiện số lần câu truy vấn  $q_i$  thực hiện cập nhật tại mảnh  $F_j$ .

Ví dụ : RM :

$$q_i \begin{pmatrix} F_1 & F_2 & F_3 & F_4 & F_5 \\ 2 & 3 & 0 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 \\ 0 & 0 & 3 & 0 & 0 \\ 3 & 0 & 2 & 0 & 0 \end{pmatrix}$$

UM:

$$q_i \begin{pmatrix} F_1 & F_2 & F_3 & F_4 & F_5 \\ 0 & 0 & 0 & 1 & 2 \\ 0 & 3 & 0 & 0 & 0 \\ 2 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{pmatrix}$$

Ma trận RM cho biết truy vấn  $q_3$  thực hiện chỉ đọc 3 lần ở mảnh  $F_3$ .

Ma trận UM cho biết truy vấn  $q_3$  thực hiện cập nhật 2 lần ở mảnh  $F_1$ , 1 lần ở mảnh  $F_2$  và 1 lần ở mảnh  $F_4$ .

Ngoài ra, chúng ta cũng cần phải có ma trận FREQ ( $q \times m$ ) (dòng là các truy vấn, cột là các site) thể hiện tần số chạy truy vấn tại mỗi site. Phần tử  $FREQ_{ij}$  thể hiện số lần câu truy vấn  $q_i$  thực hiện tại site  $S_j$ .

Ví dụ : ma trận FREQ

$$q_i \begin{pmatrix} S_1 & S_2 & S_3 & S_4 \\ 0 & 2 & 3 & 1 \\ 0 & 3 & 0 & 0 \\ 2 & 0 & 1 & 0 \\ 0 & 0 & 4 & 0 \end{pmatrix}$$

Ma trận FREQ cho biết truy vấn  $q_3$  được chạy 2 lần ở site  $S_1$ , 1 lần ở site  $S_3$ .

- Thông tin về mạng :

Ma trận CTR( $m \times m$ ) thể hiện chi phí truyền dữ liệu. Phần tử  $CTR_{ij}$  là chi phí truyền 1 đơn vị dữ liệu từ  $S_i$  đến  $S_j$

$$\begin{matrix}
 & S_1 & S_2 & S_3 & S_4 \\
 \begin{matrix} S_1 \\ S_2 \\ S_3 \\ S_4 \end{matrix} & \begin{pmatrix} 0 & 0.32 & 0.48 & 0.16 \\ 0.32 & 0 & 0.64 & 0.32 \\ 0.48 & 0.64 & 0 & 0.64 \\ 0.16 & 0.32 & 0.64 & 0 \end{pmatrix}
 \end{matrix}$$

### 3.3 . Hàm đánh giá :

Để đánh giá xem 1 phương án cấp phát mảnh vào các site có tổng chi phí là bao nhiêu, chúng ta có thể dựa vào hàm đánh giá chi phí :

$$Cost = CC_{load} + CC_{proc}$$

Trong đó  $CC_{load}$  là chi phí nạp dữ liệu vào mạng.  $CC_{proc}$  là chi phí truyền tin để xử lý truy vấn.

Ngày nay, với sự phát triển của công nghệ thông tin và truyền thông thì chi phí load dữ liệu không còn quan trọng nữa. Vì thế chúng ta chỉ quan tâm đến chi phí truyền tin để xử lý truy vấn. Như vậy hàm đánh giá chi phí có thể xem như sau (xem [3]):

$$Cost = CC_{proc} = \sum_{k=1}^m \sum_{i=1}^q FREQ_{ik} * (TR_i + TU_{i,k})$$

$$\text{Trong đó : } TR_i = \sum_{j=1}^n r_{ij}$$

$$TU_{i,k} = \sum_{j=1}^n u_{ij} * \sum_{l=1}^m FAM_{jl} * SIZE(F_j) * CTR_{kl}$$

### 3.4 . Bài toán tối ưu :

Như trên đã đề cập, bài toán tối ưu là tìm một phương án cấp phát các mảnh  $F_i$  vào các site  $S_j$  sao cho chi phí là nhỏ nhất hay hàm Cost đạt giá trị min. Sự cấp phát đó phải thỏa điều kiện:

- Mỗi site phải được cấp ít nhất 1 mảnh.
- Mỗi mảnh phải được cấp phát cho ít nhất 1 site.

### 3.5 . Các bước thực hiện giải bài toán

(a) Xem đối tượng cần tìm của bài toán như là 1 cá thể. Mô tả lớp đối tượng cần tìm theo một cấu trúc dữ liệu, biểu diễn không gian tìm kiếm của bài toán.

- Cá thể : là một phương án cấp phát
- Cấu trúc biểu diễn cá thể : ma trận  $FAM = [x_{ij}] \quad \forall i=1..m, j=1..n$

(b) Xây dựng phương pháp khởi tạo quần thể ban đầu.

- Khởi tạo quần thể : tạo 100 cá thể, mỗi cá thể được xây dựng bằng cách tạo ma trận  $FAM$  có các phần tử  $x_{ij}$  được chọn ngẫu nhiên trong tập  $\{0, 1\}$ , thỏa điều kiện số các phần tử  $x_{ij}=1$  trên mỗi dòng lớn hơn 1 và số các phần tử  $x_{ij}=1$  trên mỗi cột lớn hơn 1.

(c) Định nghĩa hàm thích nghi, xác định độ thích nghi của các cá thể trong quần thể.

- Hàm thích nghi  $\Psi$  : là hàm đánh giá chi phí trong (5.2)

- Cách đánh giá 1 cá thể tốt: một cá thể được coi là tốt nếu hàm thích nghi có giá trị đạt cực tiểu so với các cá thể khác trong quần thể.

(d) Xác định các phương thức tiến hóa của thuật giải di truyền đột biến, tái sinh và chọn lọc.

- Phép đột biến : chọn vị trí  $(i, j)$  ngẫu nhiên, nếu giá trị  $x_{ij}$  là 1 thì chuyển thành 0, nếu giá trị  $x_{ij}$  là 0 thì chuyển thành 1.

Ví dụ : Cho cá thể  $FAM_i = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$

- Phép đột biến có thể được thực hiện như sau: chọn ngẫu nhiên vị trí (3, 2) (dòng 3 cột 2), giá trị  $x_{32}=1$  nên được chuyển thành 0.

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & \textcircled{1} & 0 \\ 0 & 1 & 0 \end{bmatrix} \xrightarrow{\substack{\text{Đột biến đổi 1} \\ \text{thành 0}}} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & \textcircled{0} & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

- Phép chọn lọc: Sắp xếp các cá thể trong quần thể theo thứ tự độ thích nghi giảm dần (hàm chi phí tăng dần). Loại bỏ các cá thể cuối dãy để chỉ giữ lại 1 cá thể tốt nhất.

- Phép tái sinh: Ở thế hệ sau giữ lại cá thể tốt nhất ( $FAM$  có chi phí thấp nhất) trong thế hệ trước, khởi tạo ngẫu nhiên các cá thể còn lại trong quần thể.

Ý tưởng chính của việc xây dựng thuật giải di truyền cho bài toán cấp phát là trong quá trình tiến hóa chỉ sử dụng 2 phép tác động quần thể: phép đột biến và phép tái sinh.

Trong phép tái sinh, cá thể tốt nhất của quá trình trước sẽ được giữ lại, vì thế luôn luôn bảo đảm rằng kết quả tìm được ở quá trình sau sẽ tốt hơn hoặc bằng quá

trình trước. Trong khi đó, phép đột biến có thể cho phương án tìm được không tốt hơn, thì thuật toán cũng cho phép hủy bỏ thao tác đó.

#### 4. Kết luận

Bài báo này đã trình bày một phương pháp giải bài toán tối ưu trong không gian vô cùng lớn bằng thuật giải di truyền. Hiện nay, thuật giải di truyền được xem như là một công cụ mạnh mẽ để giải quyết vấn đề tìm ra lời giải tối ưu cho các bài toán như lập thời gian biểu, điều khiển robot, bài toán vận tải, người đi du lịch...

Khả năng của thuật giải di truyền vẫn đang còn được khám phá và khai thác để giải các bài toán trong tin học. Nhiều nhà khoa học đã có các công trình nghiên cứu thành công về lý thuyết và ứng dụng trong nhiều lĩnh vực của thế giới thực, điều đó chứng minh rằng thuật giải di truyền là một kỹ thuật mạnh mẽ và thiết thực.

#### TÀI LIỆU THAM KHẢO

1. Nguyễn Đình Thúc (chủ biên) (2002), *Trí tuệ nhân tạo – Lập trình tiến hóa*, Nxb.Giáo dục, Hà Nội.
2. Tamer Özsu - Patrick Valduriez (Trần Đức Quang biên dịch), (1999), *Nguyên lý các hệ cơ sở dữ liệu phân tán*, Nxb.Thống kê, Hà Nội.
3. Yin-Fu Huang, Jyn-Her Cheng: "Fragment Allocation in Distributed Database Design". In: *Journal of information science and engineering*, 2001.

#### STUDYING NON-LINEAR MODELS IN SOLVING THE OPTIMIZATION PROBLEMS WITH GENETIC ALGORITHMS

#### ABSTRACT

*Using the generic algorithms to generate solutions to the optimization problems in computer science is fascinating as its principles are based on the laws of struggle for existence in nature. This research suggests a method of using genetic algorithms to generate solutions to optimization problems in a vast space. Together illustrative examples, allocation problems in distributed database are considered as an optimization problem of NP-hard.*

**Key words:** *optimization problems, generic algorithms, allocation problems.*