Tìm kiếm Heuristic – Leo đồi, Các Thuật toán Tìm kiếm Cục bộ và Thuật giải Di truyền

Lê Ngọc Thành

Khoa Công nghệ Thông tin

lnthanh@fit.hcmus.edu.vn

Tổng quát

- Thuật giải leo đồi
- Vấn đề của thuật giải leo đồi
- Thuật giải leo đồi ngẫu nhiên
- Bài toán tối ưu hoá và các thuật toán tìm kiếm cục bộ
- Thuật giải di truyền
- Một số vấn đề lựa chọn của thuật giải di truyền
- Một ví dụ đơn giản

Thuật giải leo đồi

Các thuật toán tìm kiếm toàn cục: sử dụng quá nhiều tài nguyên (A*) hoặc thời gian (IDA*) để tìm được lời giải tối ưu.

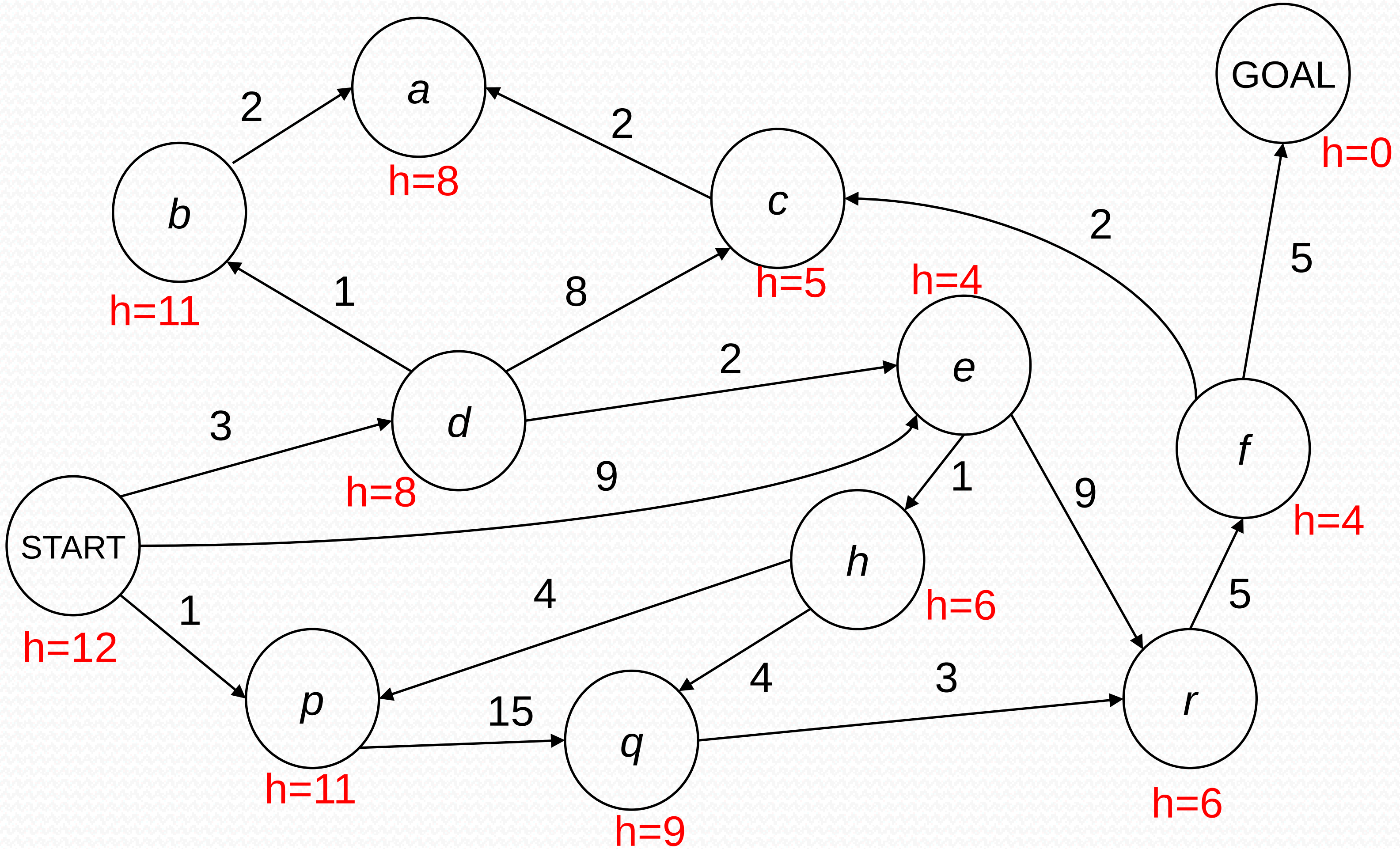
Ta có thể thực hiện việc tìm kiếm lời giải trong thời gian và không gian hợp lý?

Thuật giải leo đồi

Leo đồi: Cố gắng tối đa hoá Eval(X) bằng cách di chuyển đến cấu hình cao nhất trong tập di chuyển của mình – Leo đồi dốc đứng

Đặt S := trạng thái ban đầu
Lặp
Tìm trạng thái con S' của S với Eval(S') thấp nhất
Nếu Eval(S') không tốt hơn Eval(S) thì
return S
Ngược lại
S = S'

Thuật giải leo đồi



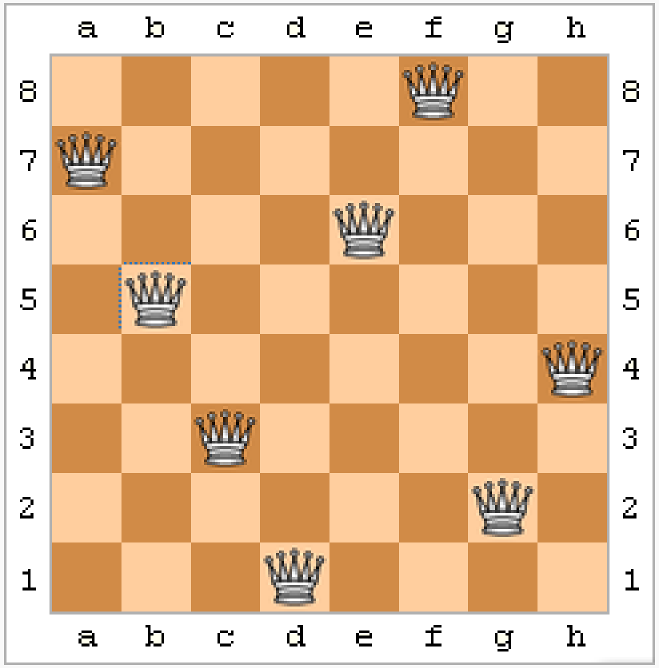
Leo đồi ngẫu nhiên

Đặt S := trạng thái ban đầu
Lặp sau một MAX lần cố gắng nào đó
   Lấy một trạng thái con ngẫu nhiên S’ của S
      Nếu Eval(S’) tốt hơn Eval(S) thì
         S= S’
Cuối lặp
Return S

Sau khi chạy vài lần có thể đưa đến trạng thái đích

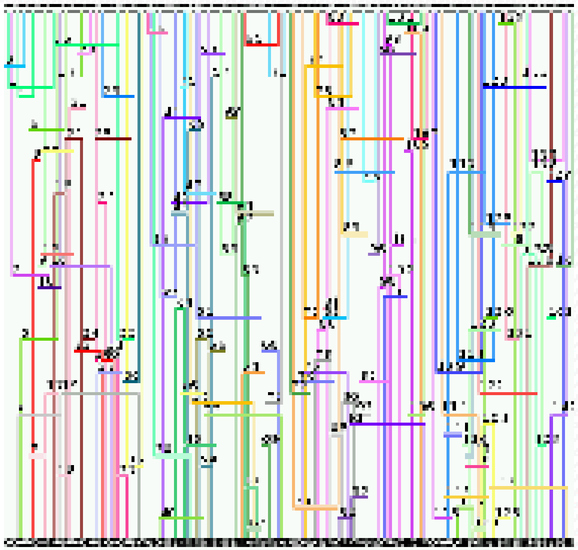
Ví dụ về bài toán tối ưu hoá

- Bài toán n-Hậu
  - Đây là một bài toán Thoả mãn Ràng buộc (Contraint Satisfaction Problem CSP)
  - Có thể xem xét dưới dạng một bài toán tối ưu hoá với hàm lượng giá ***h*** = *số lượng cặp hậu đe doạ lẫn nhau*

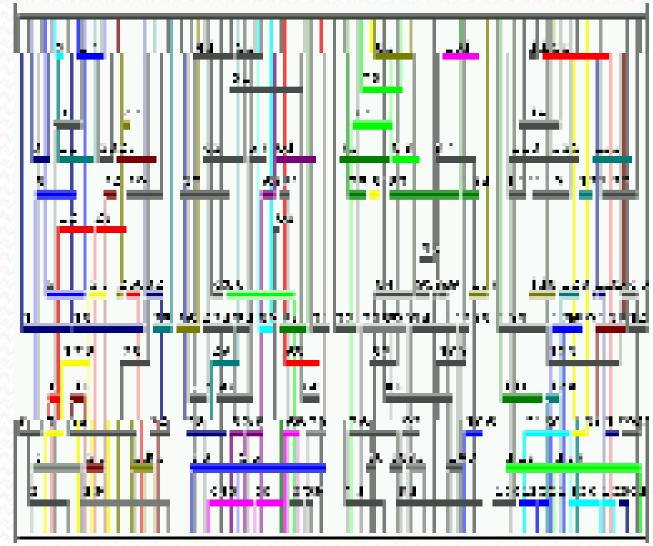


One possible solution

Ví dụ về bài toán tối ưu hoá

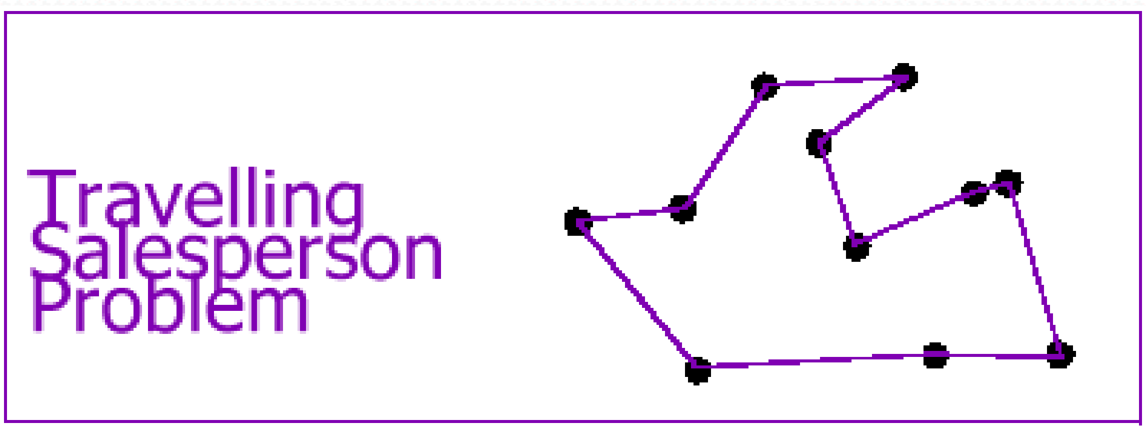


Thiết kế
Mạch điện



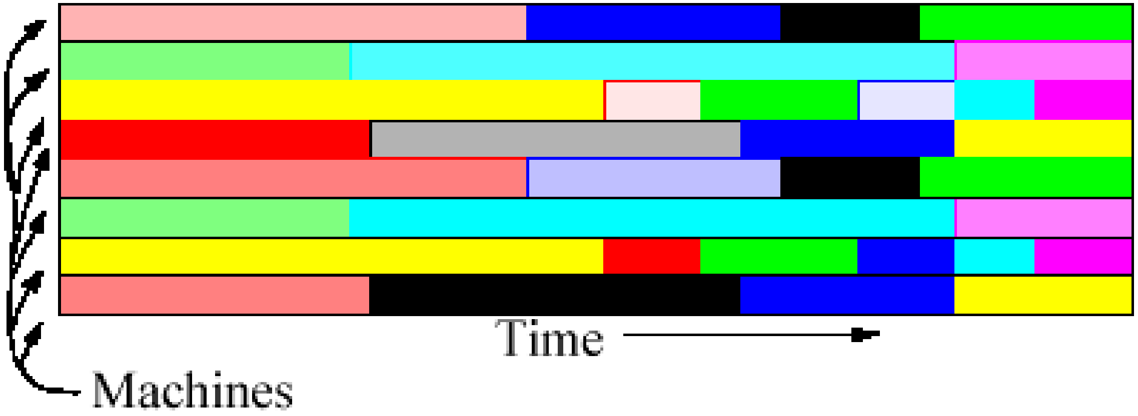
Có rất nhiều chip cố định

Cùng số kết nối nhưng tốn ít không gian hơn



Ví dụ về bài toán tối ưu hoá

Least cost, constrained, schedule



Boolean SATisfiability

$A \vee \neg B \vee C$

$\neg A \vee C \vee D$

$B \vee D \vee \neg E$

$\neg C \vee \neg D \vee \neg E$

$\neg A \vee \neg C \vee E$

(2000 variables, 8500 clauses)

Bài toán tối ưu hoá

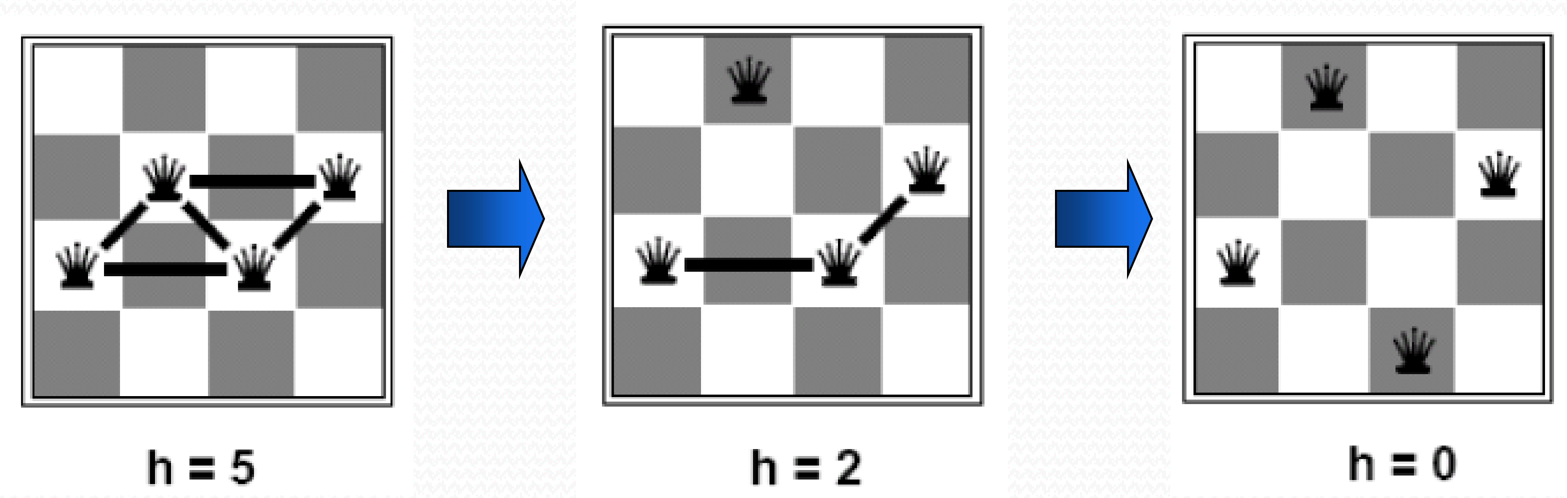
- Ta chỉ quan tâm đến việc đạt được một cấu hình tối ưu mà không cần quan tâm đến đường đi
- Xây dựng một tập di chuyển (moveset) từ một trạng thái sang một trạng thái khác

  VD: Cho biết tập di chuyển của

  Bài toán N-queen?
- Phát sinh ngẫu nhiên trạng thái ban đầu
- Thực hiện di chuyển xuống (lên) đồi

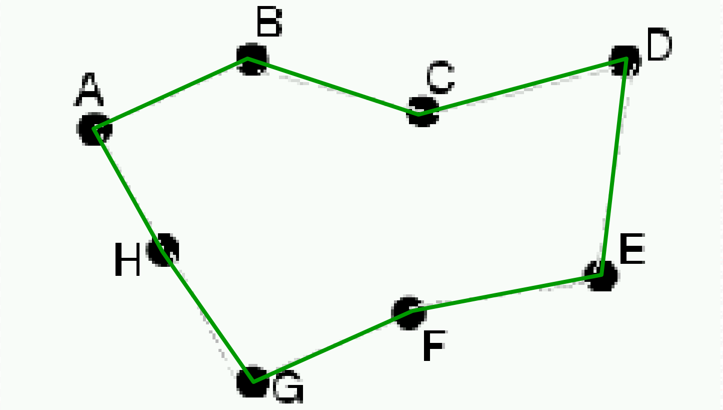
Ví dụ về bài toán tối ưu hoá

- Thuật giải leo đồi thực hiện với bài toán n-Hậu



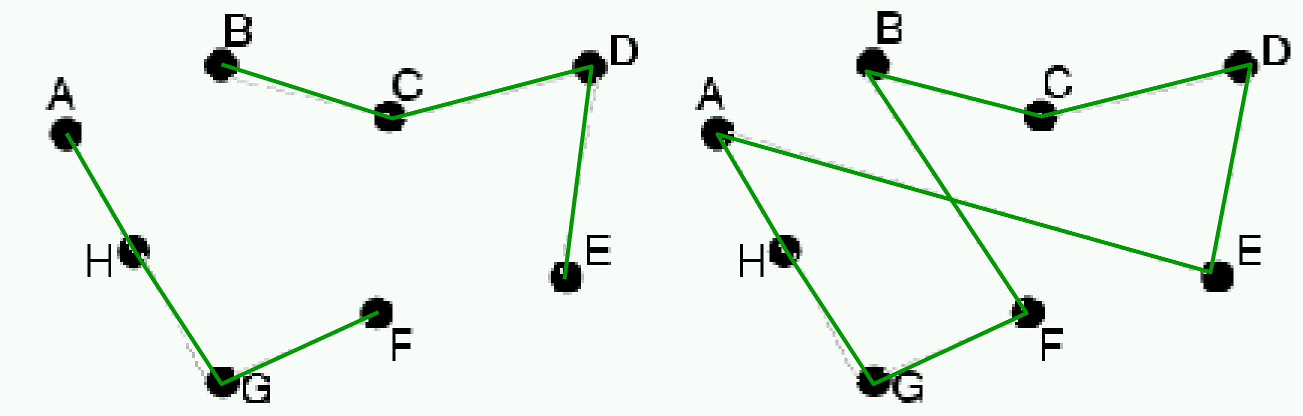
Ví dụ Leo đồi: TSP

Tối thiểu hóa: Eval(Config) = độ dài đường đi

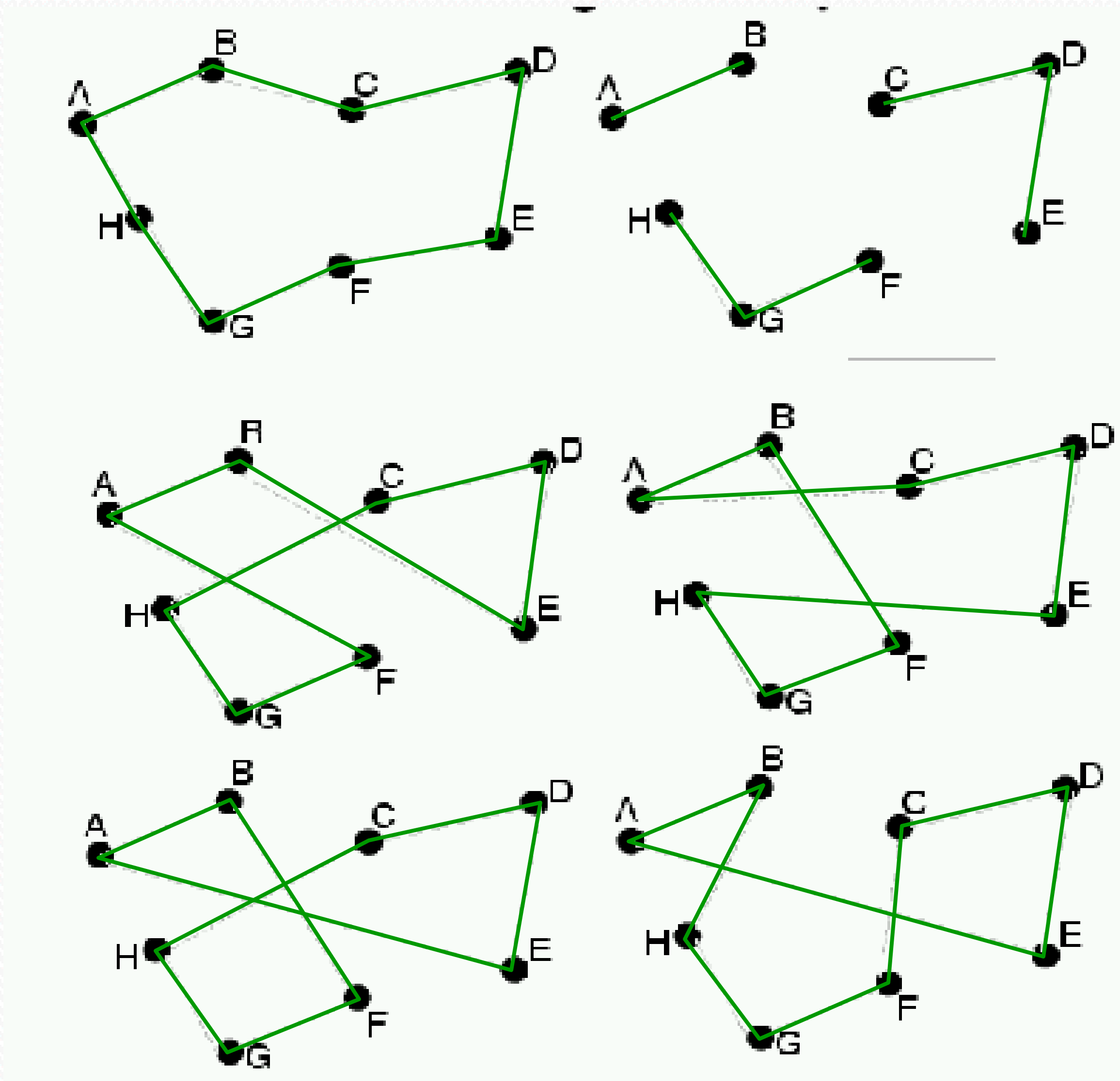


Tập di chuyển: 2-change … k-change

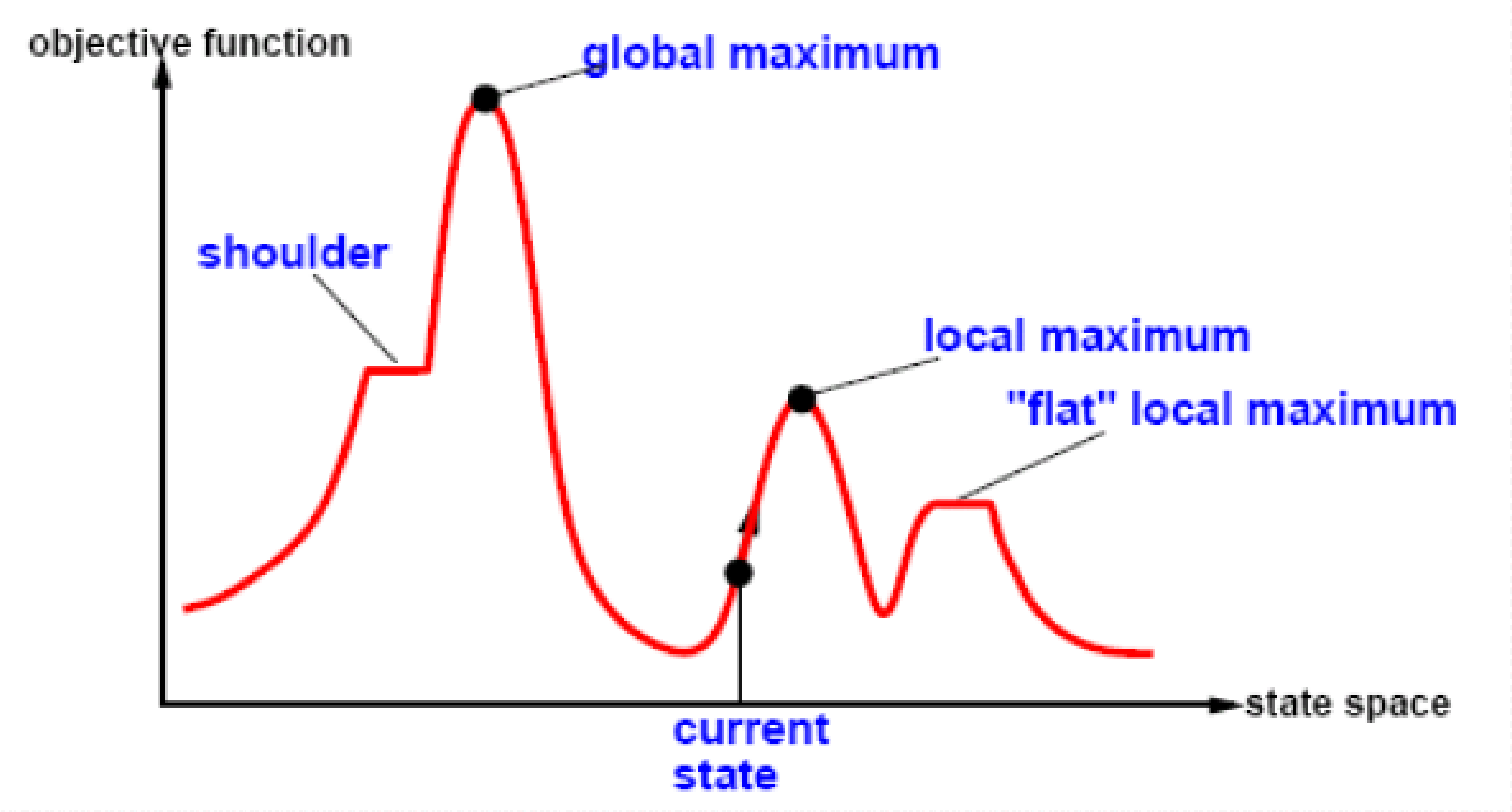
Ví dụ: 2-change



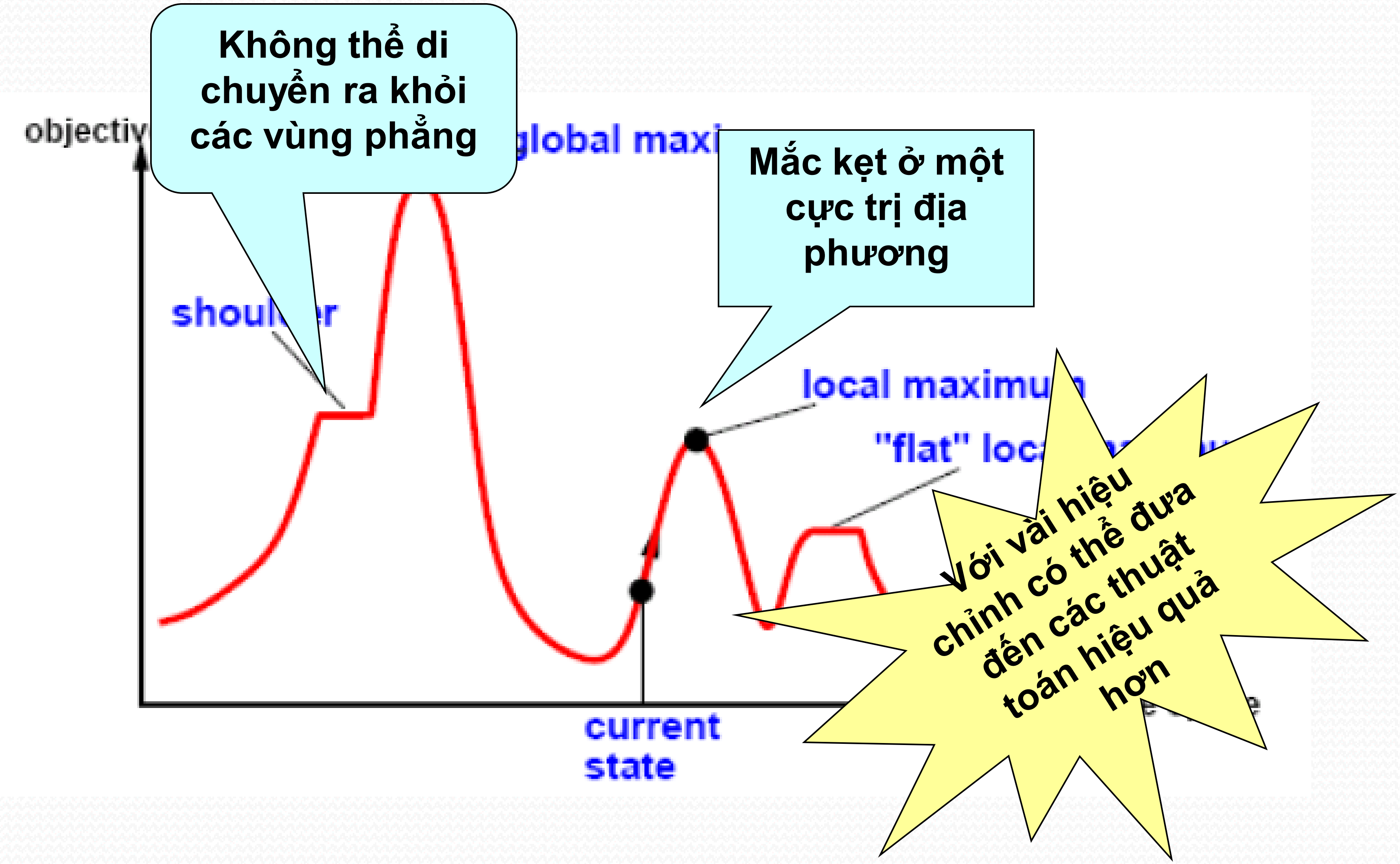
Ví dụ 3-change



Các vấn đề của leo đồi…



Các vấn đề của leo đồi…



Tìm kiếm leo đồi

- Leo đồi với khởi tạo ngẫu nhiên nhiều lần
- Local beam search:
  - Theo dõi *k* trạng thái cùng một lúc
  - Khởi tạo với *k* trạng thái phát sinh ngẫu nhiên
  - Tại mỗi lần lặp, tất cả trạng thái con của *k* trạng thái được phát sinh
  - Nếu xuất hiện trạng thái đích thì dừng lại; ngược lại chọn *k* trạng thái con tốt nhất từ toàn bộ danh sách và lặp lại

Luyện Thép

1. Đặt X := cấu hình ban đầu
2. Đặt E := Eval(X)
3. Đặt i = di chuyển ngẫu nhiên từ moveset
4. Đặt E_i := Eval(move(X,i))
5. Nếu E < E_i thì

   X := move(X,i)

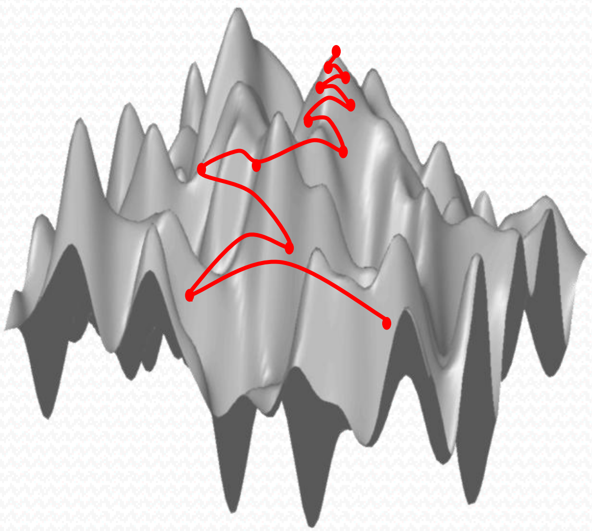
   E := E_i

   Ngược lại với xác suất nào đó, chấp nhận di chuyển ngay cả khi mọi chuyện xấu hơn:

      X := move(X,i)

      E := E_i

6. Quay lại 3 đến khi kết thúc.



Luyện Thép

1. Đặt X := cấu hình ban đầu
2. Đặt E := Eval(X)
3. Đặt i = di chuyển ngẫu nhiên từ moveset
4. Đặt E_i := Eval(move(X,i))
5. Nếu E < E_i thì

    X := move(X,i)

    E := E_i

    Ngược lại với xác suất nào đó, chấp nhận di chuyển ngay cả khi mọi chuyện xấu hơn:

    X := move(X,i)

    E := E_i

6. Quay lại 3 đến khi kết thúc.

Chúng ta sẽ chọn xác suất chấp nhận một di chuyển tồi hơn như thế nào?

- Xác suất = 0.1
- Xác suất giảm theo thời gian
- Xác suất

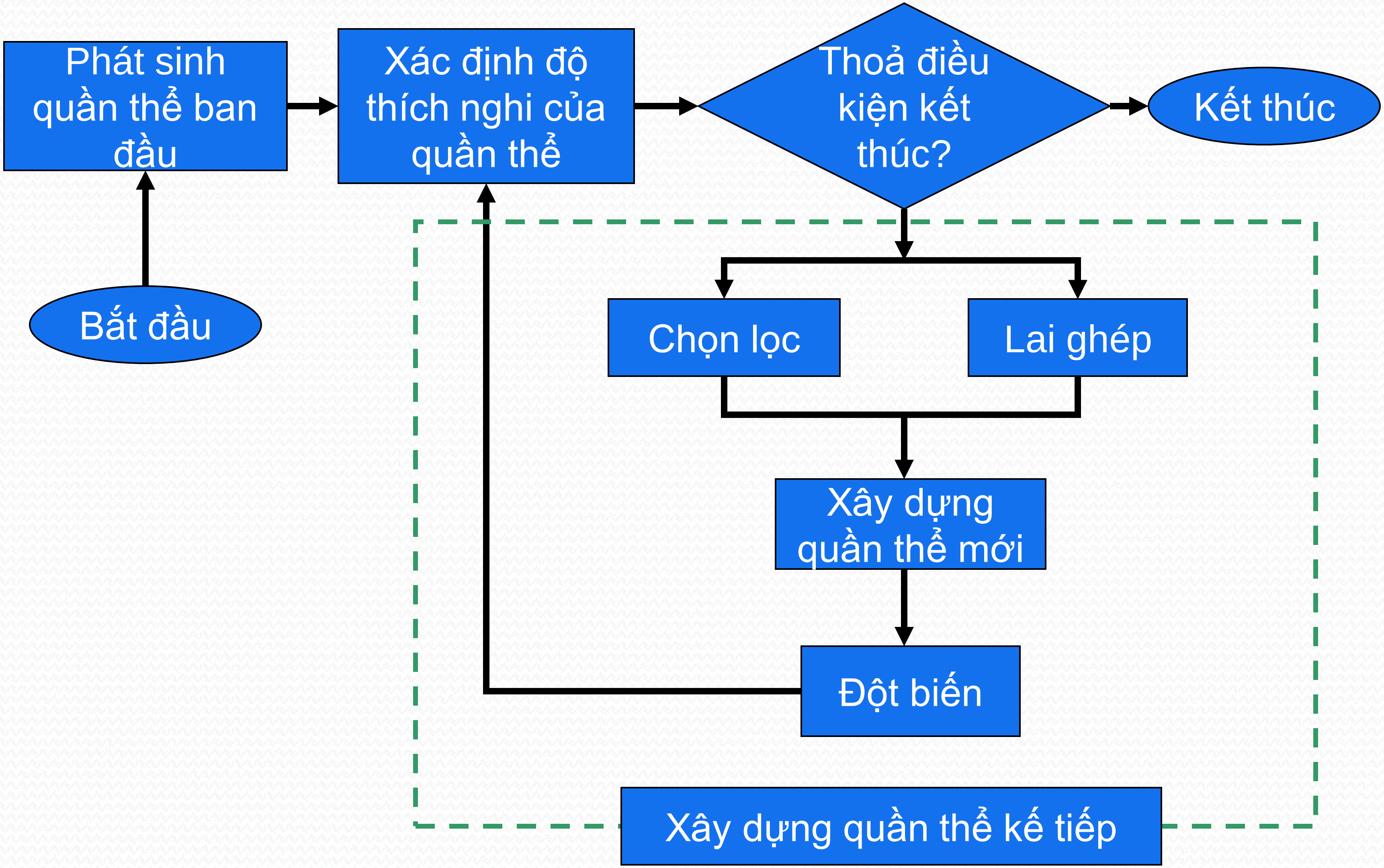
  exp (-(E - E*i*)/**T*i***): T_i là tham số nghiệt độ

Tương tự như quá trình làm lạnh trong luyện thép vật lý

Thuật giải di truyền

- Được giới thiệu bởi John Holland năm 1975, cho phép thực hiện tìm kiếm ngẫu nhiên
- Mã hoá các lời giải tìm năng của bài toán bằng các nhiễm sắc thể
- Đánh giá độ tốt của các lời giải qua độ thích nghi của các nhiễm sắc thể
- Lưu trữ một quần thể các lời giải tiềm năng
- Thực hiện các phép toán di truyền để phát sinh các cá thể mới đồng thời áp dụng chọn lọc tự nhiên trên các lời giải

Thuật giải di truyền



Một số cách biểu diễn gen

- Để có thể giải bài toán bằng thuật giải di truyền ta phải gen hóa cấu trúc dữ liệu của bài toán. Có hai cách biểu diễn gen:

1. Biểu diễn gen bằng chuỗi số nguyên (hay thực)
   - VD: Bài toán 8 hậu -> 12534867
2. Biểu diễn gen bằng chuỗi nhị phân
   - VD: Bài toán 8 hậu: dùng 8 x $\log_2 8$ bit để biểu diễn
   - Làm sao biểu diễn nghiệm thực bằng chuỗi nhị phân ???
   - Trả lời: Rời rạc hoá miền trị với một độ chính xác cho trước

Các khái niệm cơ bản

- Độ tốt của một cá thể
  - Là giá trị của cá thể cho một vấn đề bài toán cụ thể.

    Ví dụ: Trong bài toán tối ưu cực đại một hàm f, nếu chọn một cá thể là một nghiệm của bài toán thì một cá thể càng tốt khi làm cho giá trị hàm càng lớn.

  - Để xác định được độ tốt của các cá thể ta cần một hàm để làm việc này. Hàm này gọi là Hàm mục tiêu .

Các khái niệm cơ bản

- Hàm mục tiêu
  - Dùng để đánh giá độ tốt của một lời giải hoặc cá thể.
  - Hàm mục tiêu nhận vào tham số là gen của một cá thể và trả ra một số thực.
  - Tùy theo giá trị của số thực này mà ta biết được độ tốt của cá thể đó .

Các khái niệm cơ bản

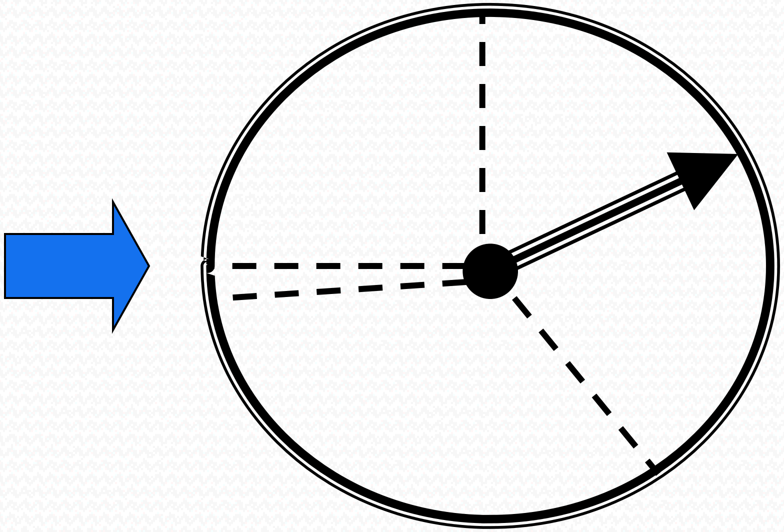
- Độ thích nghi của các cá thể (fitness)
  - Là khả năng cá thể đó được chọn lọc vào thế hệ sau hoặc là được chọn lọc cho việc lai ghép để tạo ra cá thể con .
  - Vì độ thích nghi là một xác suất để cá thể được chọn nên người ta thường ánh xạ độ thích nghi vào đoạn [0,1] (độ thích nghi chuẩn)

$$F(a_i) = \frac{F(a_i)}{\sum_{j=1}^{N} F(a_j)} \quad i = 1,2\ldots N$$

Các toán tử cơ bản

- Toán tử lai ghép:
  - Các cá thể được chọn để lai ghép dựa vào dựa vào độ thích nghi
  - Dùng qui tắc bàn quay rollete:
    - Vd: các ta có quần thể với độ thích nghi chuẩn sau

STT	Cá thể	ĐTN chuẩn
1	0010001	0,4
2	0010101	0,3
3	0101000	0.05
4	1100011	0.25



Các toán tử cơ bản

- Toán tử lai ghép:
  - Lấy giá trị ngẫu nhiên $p \in [0,1]$ để chọn cá thể lai ghép, cá thể có độ thích nghi cao có xác xuất lựa chọn nhiều hơn
  - Sau khi lựa chọn một cặp cá thể cha mẹ, hoán vị các nhiễm sắc thể tại vị trí ngẫu nhiên với xác suất p_c
- Toán tử lai ghép có xu hướng kéo quần thể về phía các cá thể có độ thích nghi cao => cục bộ địa phương

Các toán tử cơ bản

- Toán tử đột biến:
  - Giúp lời giải có thể nhảy ra khỏi các cực trị địa phương
  - Với mỗi cá thể trong quần thể, thực hiện đột biến với xác suất p_m tại một vị trí ngẫu nhiên (thông thường $p_m << 0.1$)

0010001

0011001

Kế tiếp: Hãy xem xét một ví dụ rất đơn giản sau

Ví dụ: Giải phương trình bậc hai

$$X^2 = 64$$

- Xác định kích thước quần thể: **n= 4**
- Chọn phương pháp mã hóa nghiệm:
  - Xác định nghiệm nguyên trong miền trị: [0, 31]
  - Mã hoá theo chuỗi nhị phân: số bit mã hoá =5
- Lựa chọn hàm thích nghi
  - Hàm thích nghi = 1000 – (X2 – 64), chọn nghiệm có hệ số thích nghi ~ 1000

Ví dụ: Giải phương trình bậc hai

$$X^2 = 64$$

- Xác định kích thước quần thể: **n= 4**
- Chọn phương pháp mã hóa nghiệm
  - Xác định nghiệm nguyên tr 64]
  - Mã hoá theo chuỗi n oa =5
- Lựa chọn hàm thích
  - Hàm thích nghi = 1 4), chọn nghiệm có hệ số thích 000

Các bước sau đây được thực hiện dựa vào "ngẫu nhiên"

Ví dụ: Giải phương trình bậc hai

$$X^2 = 64$$

- Phát sinh tập quần thể ban đầu

STT	Nhị phân	Nghiệm
1	00100	4
2	10101	21
3	01010	10
4	11000	24

Ví dụ: Giải phương trình bậc hai

$$X^2 = 64$$

- Tính hệ số thích nghi (Fitness) cho quần thể

STT	Nhị phân	Nghiệm	$X^2 - 64$	Hệ số thích nghi
1	00100	4	-48	1048
2	10101	21	377	623
3	01010	10	36	964
4	11000	24	512	488

Ví dụ: Giải phương trình bậc hai

$$X^2 = 64$$

- Chọn lọc nghiệm và lai ghép

Chọn nghiệm 4 và 10 để tiến hành lai ghép với xác suất p_c và vị trí pos= 2

4	**00100**	**01000**	**8**
10	**01010**	**00110**	**6**

Ví dụ: Giải phương trình bậc hai

$$X^2 = 64$$

- Đột biến một cá thể

Với một xác suất p_m đột biến lời giải thứ 4 với vị trí pos= 4

00110 6 → 01110 14

Ví dụ: Giải phương trình bậc hai

$$X^2 = 64$$

- Tính lại hệ số thích nghi cho nghiệm mới và tiến hành chọn lọc

STT	Nhị phân	Nghiệm	$X^2 - 64$	Hệ số thích nghi
1	00100	4	-48	1048
2	01010	10	36	964
3	01000	8	0	1000
4	01110	14	132	868

Điều cần nắm

- Hiểu được thuật giải leo đồi, leo đồi ngẫu nhiên
- Nắm được các vấn đề của leo đồi
- Hiểu được các ý tưởng đằng sau Luyện thép
- Hiểu và nắm được các bước thực hiện của GA