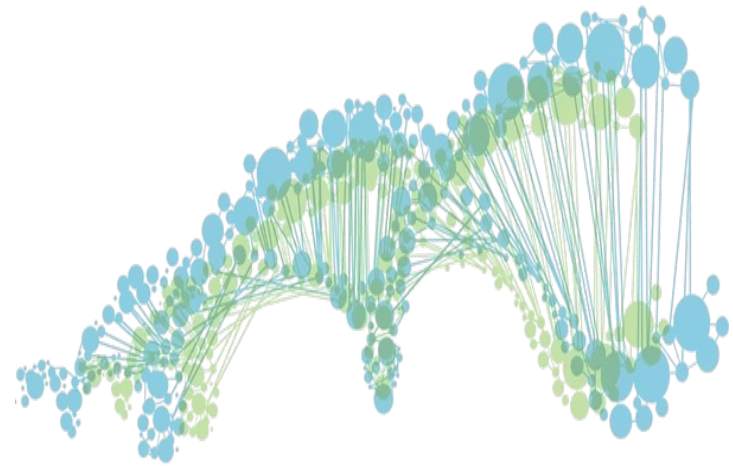
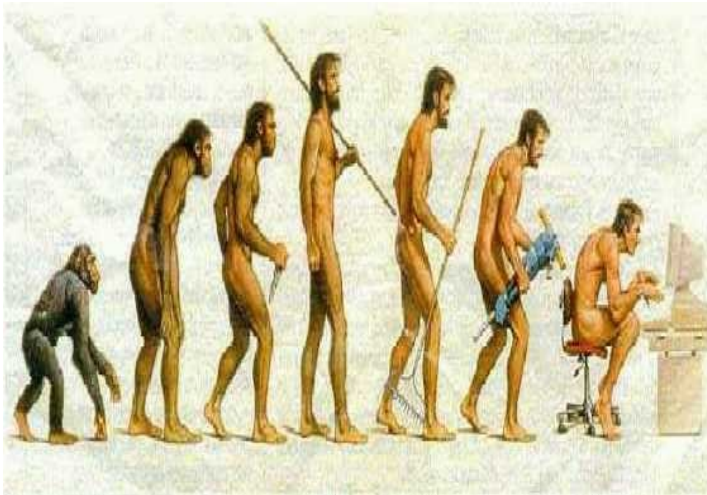


Genetic Programming



PGS.TS Huỳnh Thị Thanh Bình
Email: binhht@soict.hust.edu.vn

Nội dung

2

- Tổng quan Genetic Programming (GP)
- Các toán tử của GP
- Ví dụ minh họa



Tổng quan về Genetic Programming

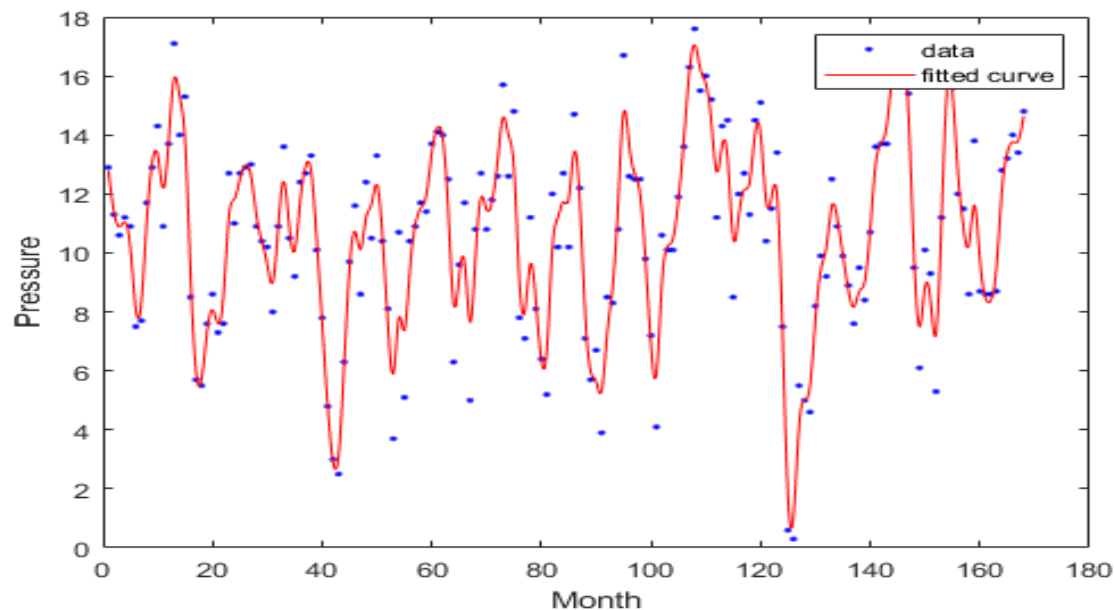
3

- Genetic Programming (Lập trình di truyền - GP) có thể coi là một **thuật toán di truyền đặc biệt**
- Sơ đồ của GP giống sơ đồ của thuật toán GA
- Điểm khác biệt giữa GA và GP
 - GA: Biểu diễn mỗi cá thể (nhiễm sắc thể) dưới dạng chuỗi các alen
 - GP: Mỗi cá thể là một hàm số hay chương trình máy tính, được biểu diễn dưới dạng cây
- Mục tiêu của GP là tìm một chương trình tối ưu trong tập không gian các chương trình có thể, để thu được hiệu suất cao nhất
- Ứng dụng: Tối ưu kiến trúc mạng Neural...

Tổng quan về Genetic Programming

4

- Tại mỗi thế hệ, mỗi cá thể (hàm, chương trình) được tiến hóa để tìm ra hàm số ẩn tối ưu, có độ lỗi thấp nhất cho bài toán
- Ví dụ: Tìm 1 hàm số $f(x)$ sao cho đi qua tất cả các đỉnh A1, A2, A3, A4...



Các toán tử của GP

5

- Biểu diễn cá thể
- Lai ghép
- Đột biến
- Đánh giá độ thích nghi

Biểu diễn cá thể

6

- Mỗi cá thể trong GP biểu diễn một chương trình máy tính hay một hàm ẩn cần tìm
- Để đảm bảo ngữ pháp của chương trình, ta cần xác định hai tập:
 - Tập kết thúc (**terminal set**) chứa các biến, hằng số, modun cơ bản có thể của chương trình
 - Tập hàm (**function set**) chứa tất cả các hàm toán học cơ bản có thể dùng: $+$, $-$, $*$, $/$, \exp , \log , and , or , xor hoặc các cấu trúc quyết định if-then-else ,...

Biểu diễn cá thể

7

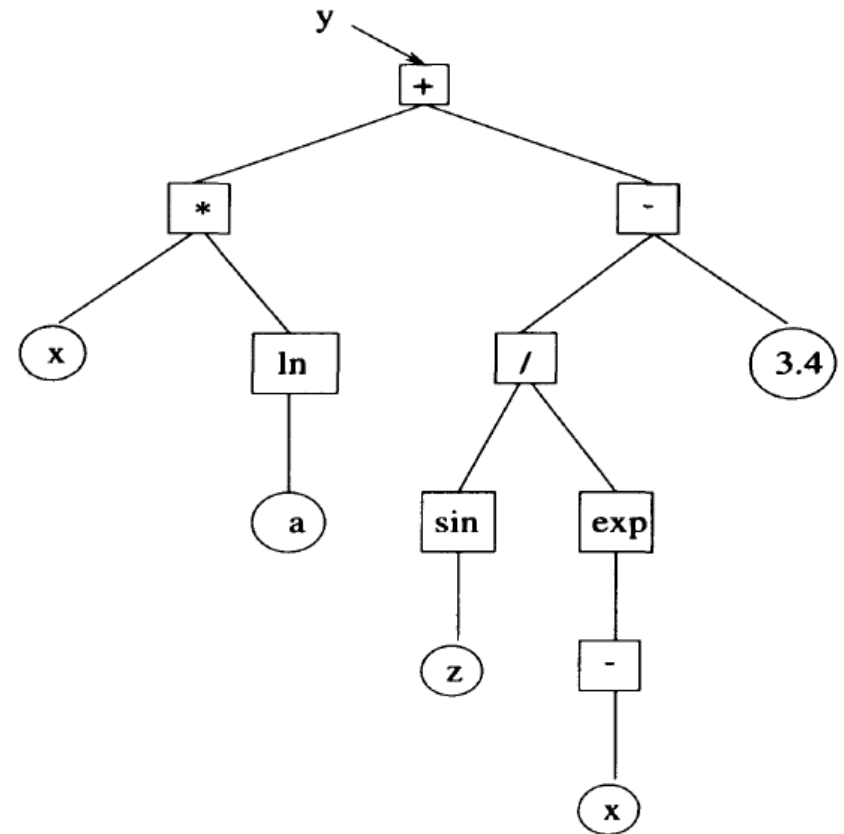
- Mỗi cá thể được biểu diễn dưới dạng một cấu trúc cây
 - Các nút lá của cây: Chọn từ tập kết thúc
 - Các nút trong của cây: Chọn từ tập hàm
- Mỗi cá thể được khởi tạo như sau:
 - Nút gốc: được chọn ngẫu nhiên trong tập hàm
 - Nút không phải gốc: Chọn ngẫu nhiên trong tập hàm hoặc tập kết thúc
 - Chọn trong tập kết thúc: Nút trở thành nút lá
 - Chọn trong tập hàm: Nút là nút trong
 - Số lượng nhánh tại mỗi nút phụ thuộc vào hàm cơ bản tại nút đó

Biểu diễn cá thể - Ví dụ

8

$y := x * \ln(a) + \sin(z) / \exp(-x) - 3.4$

- Tập kết thúc: $x, a, z, 3.4$
- Tập hàm: $*, +, -, /, \ln, \sin, \exp$



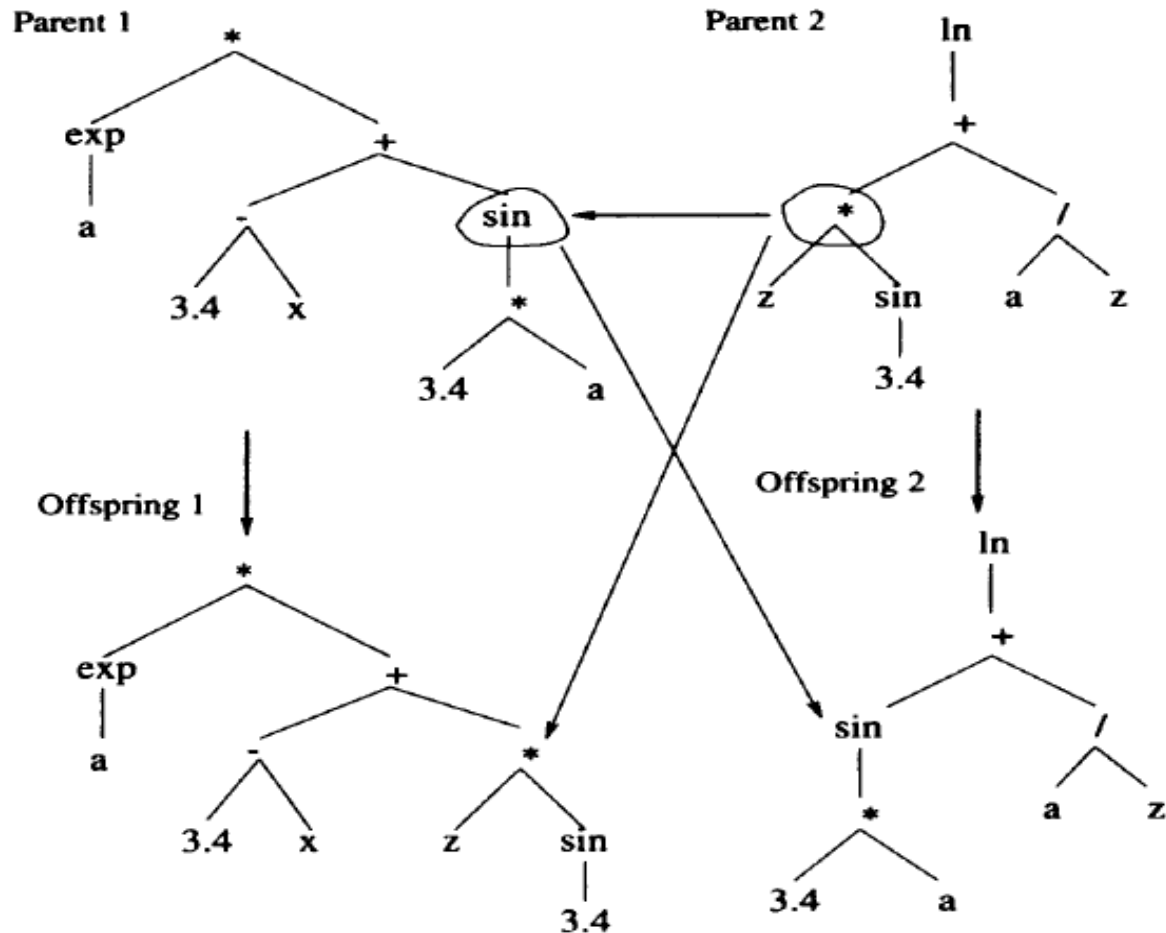
Lai ghép

9

- Các phương pháp chọn lọc cha mẹ sử dụng giống GA
- Toán tử lai ghép trong GP:
Chọn ngẫu nhiên một cây con trong mỗi cây cha mẹ và trao đổi chúng cho nhau

Lai ghép – Ví dụ

10



Đột biến

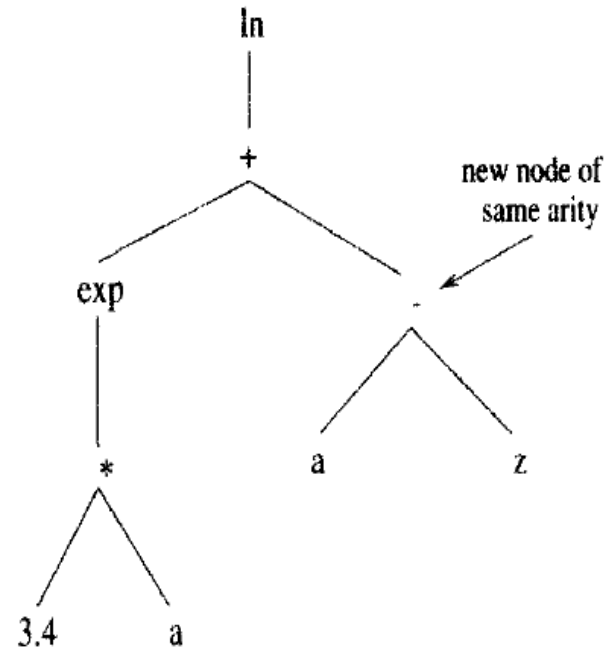
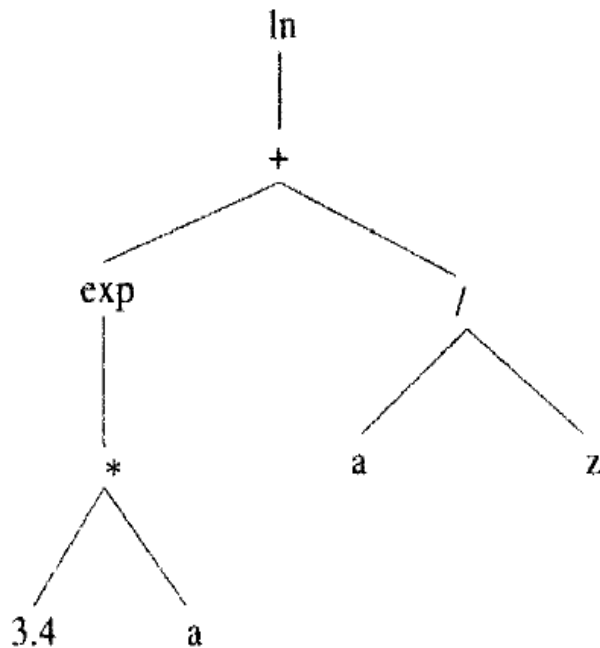
11

- Các phương pháp đột biến trong GP
 - Đột biến nút trong
 - Đột biến nút kết thúc
 - Đột biến đảo
 - Đột biến phát triển cây
 - Đột biến Gauss
 - Đột biến cắt tỉa cây

Đột biến - Đột biến nút trong

12

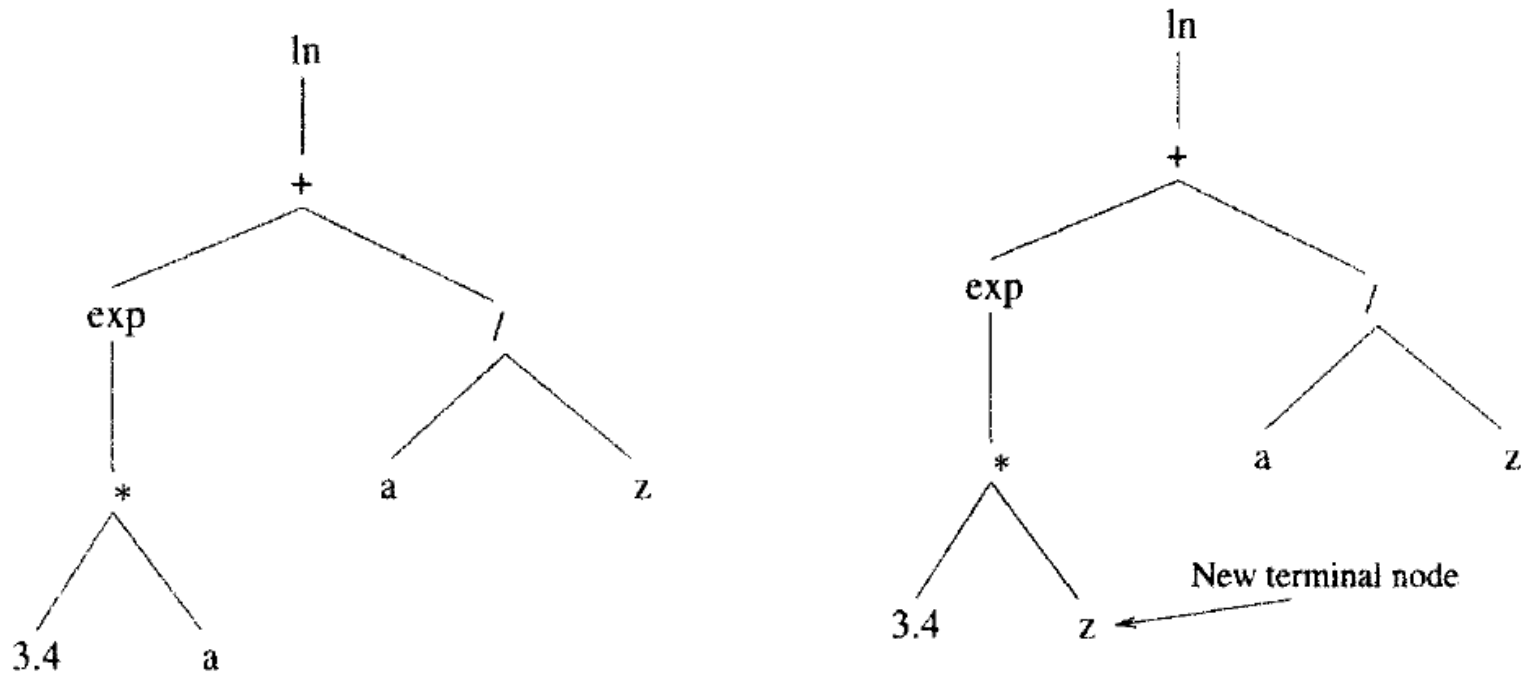
Thay thế hàm tại nút đó bằng một hàm khác trong tập hàm



Đột biến - Đột biến nút kết thúc

13

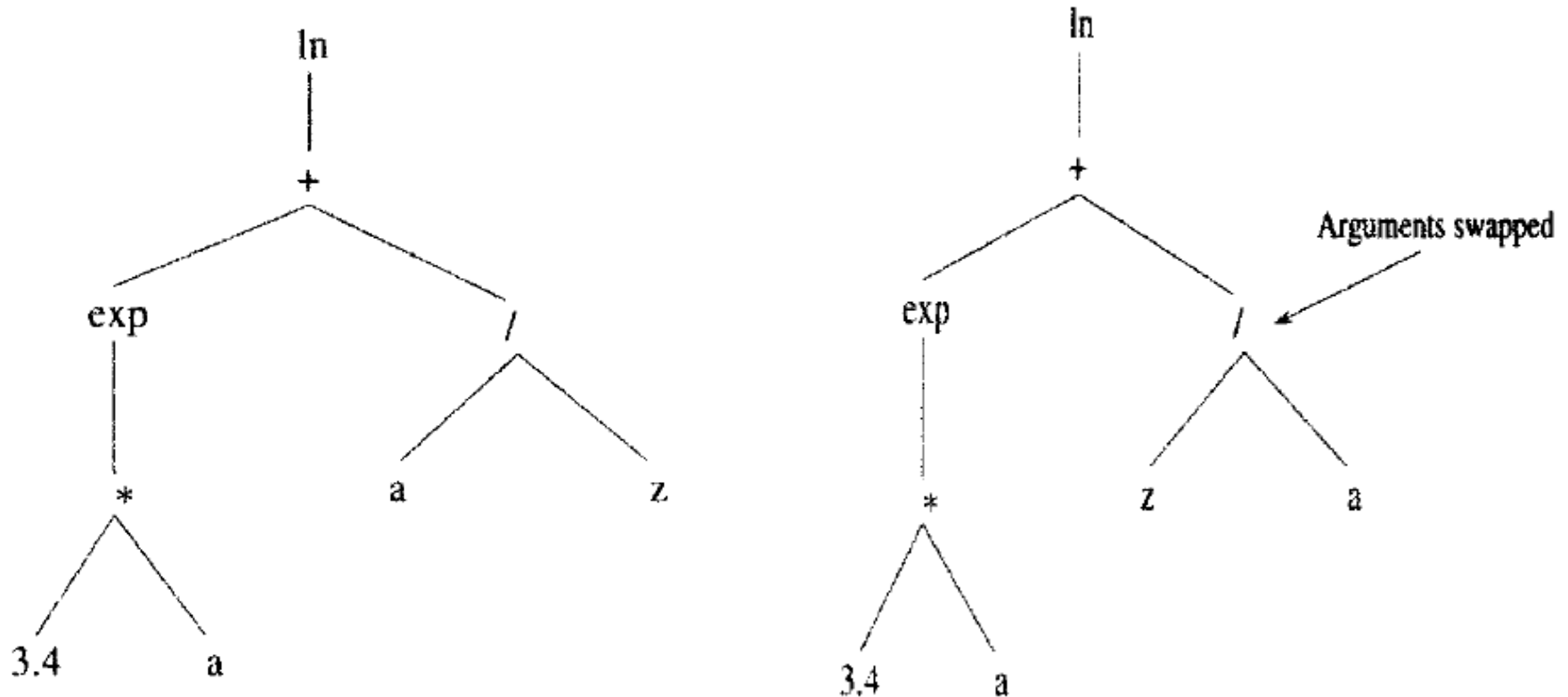
Thay thế biến, hằng tại nút đó bằng một biến, hằng khác trong tập kết thúc



Đột biến - Đột biến đảo

14

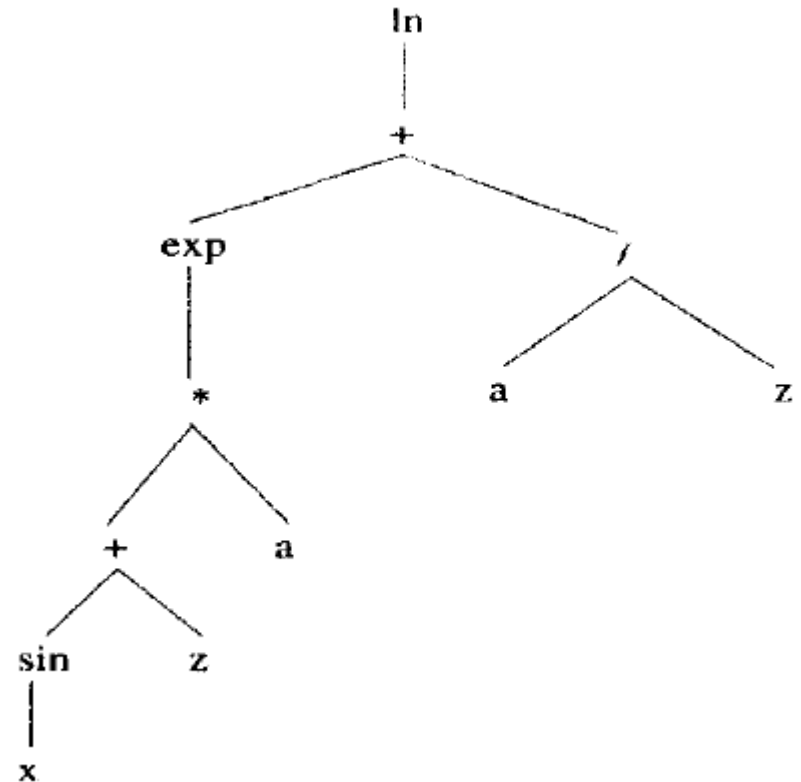
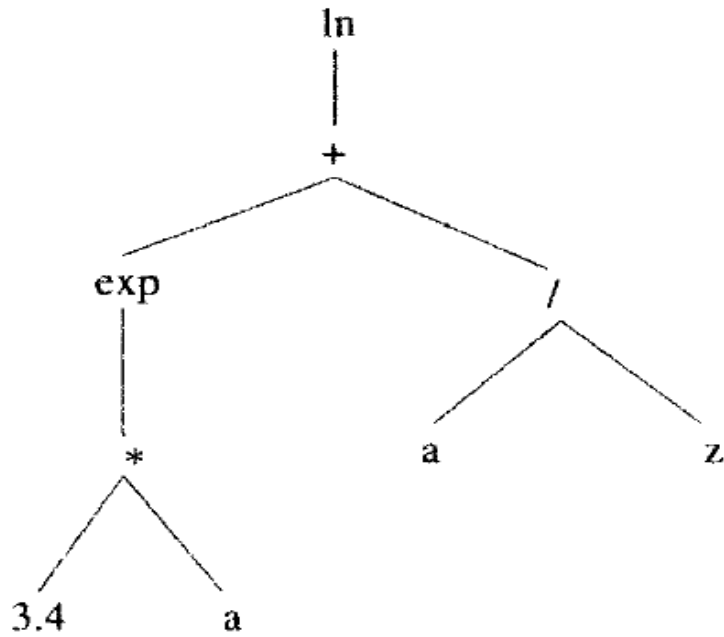
Chọn ngẫu nhiên một nút trong và đảo hai nút con của nó



Đột biến - Đột biến phát triển cây

15

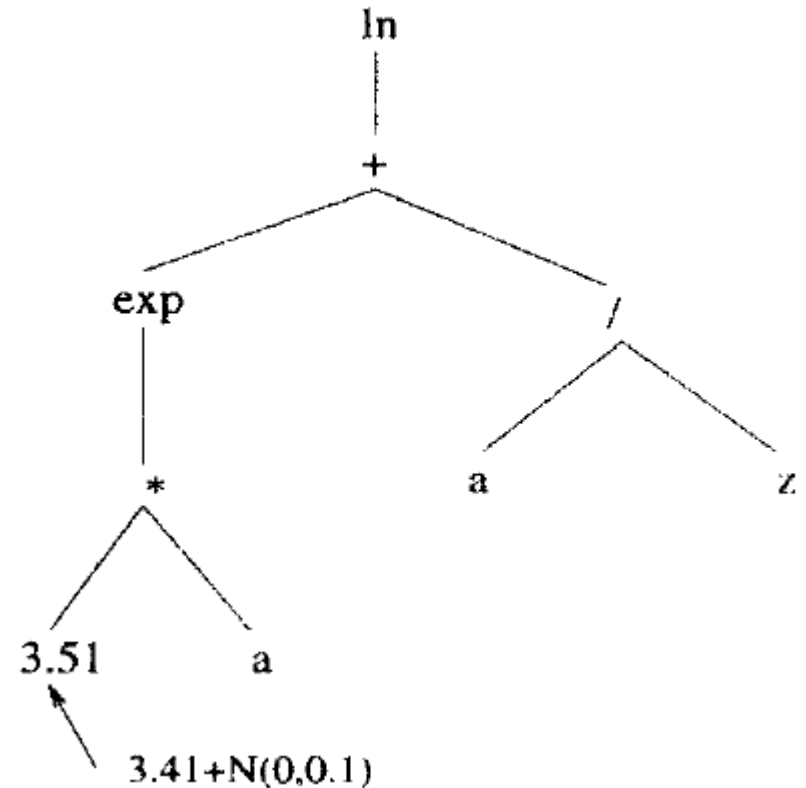
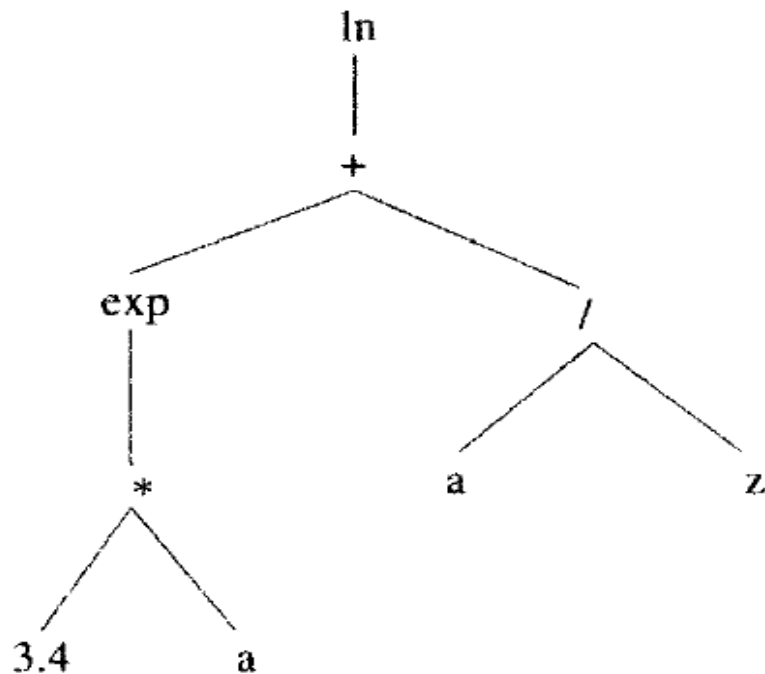
Chọn ngẫu nhiên một nút và thay thế toàn bộ cây con của nút đó bằng một cây con ngẫu nhiên khác



Đột biến - Đột biến Gauss

16

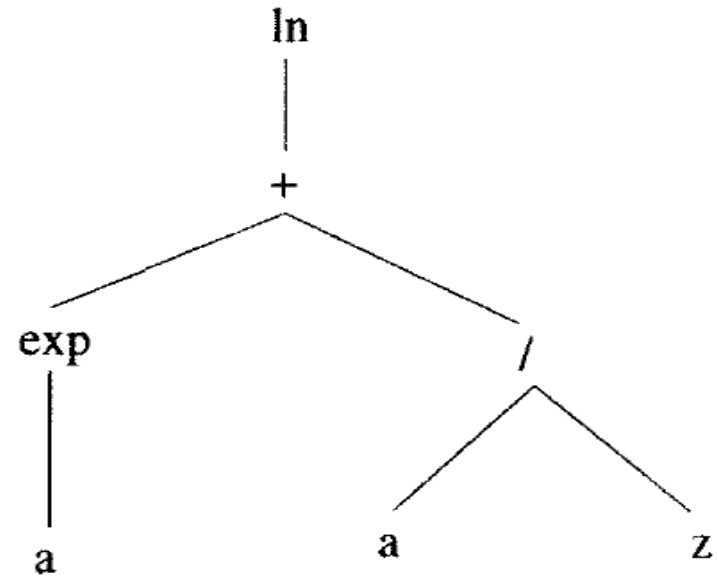
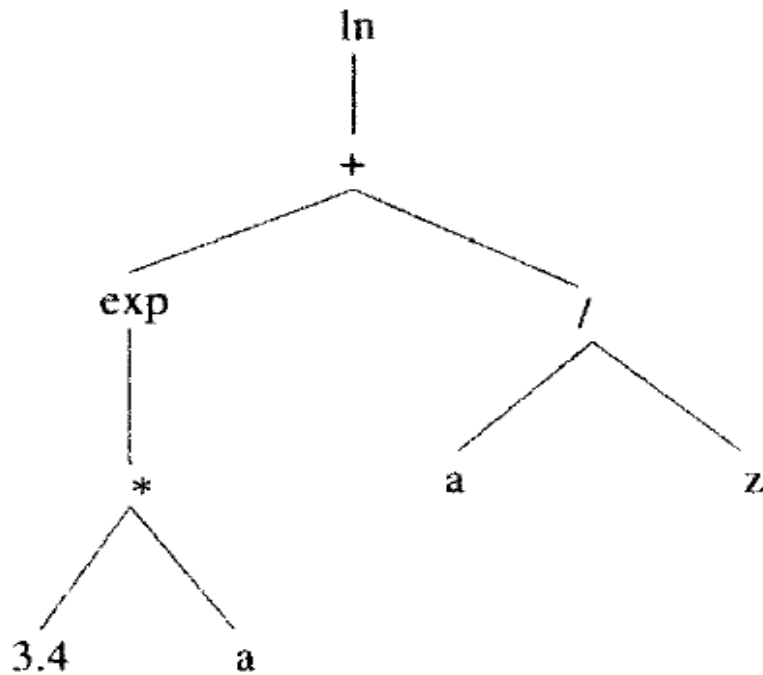
Chọn ngẫu nhiên một nút lá chứa hằng số và thêm giá trị nhiễu Gauss vào nút đó



Đột biến - Đột biến cắt tỉa cây

17

Chọn ngẫu nhiên một nút và thay thế nút đó bởi một biến hay hằng số ngẫu nhiên trong tập kết thức



Đánh giá độ thích nghi

18

- Các cá thể được đánh giá trên cùng tập mẫu dữ liệu và giá trình hiệu suất trung bình thu được trên mẫu đó được sử dụng là giá trị độ thích nghi
- Giả sử có một tập mẫu X và mỗi mẫu chứa ba giá trị đầu vào (a, x, z) và giá trị đích y . Độ thích nghi được tính như sau:
 - Tính giá trị đầu y^{\wedge} ra thu được của chương trình mà cá thể biểu diễn với mỗi đầu vào (a, x, z)
 - Tính giá trị lỗi y^{\wedge} so với y
 - Độ sai số trung bình MSE trên toàn bộ tập dữ liệu được xem như tiêu chí để đánh giá độ

Ví dụ minh họa

19

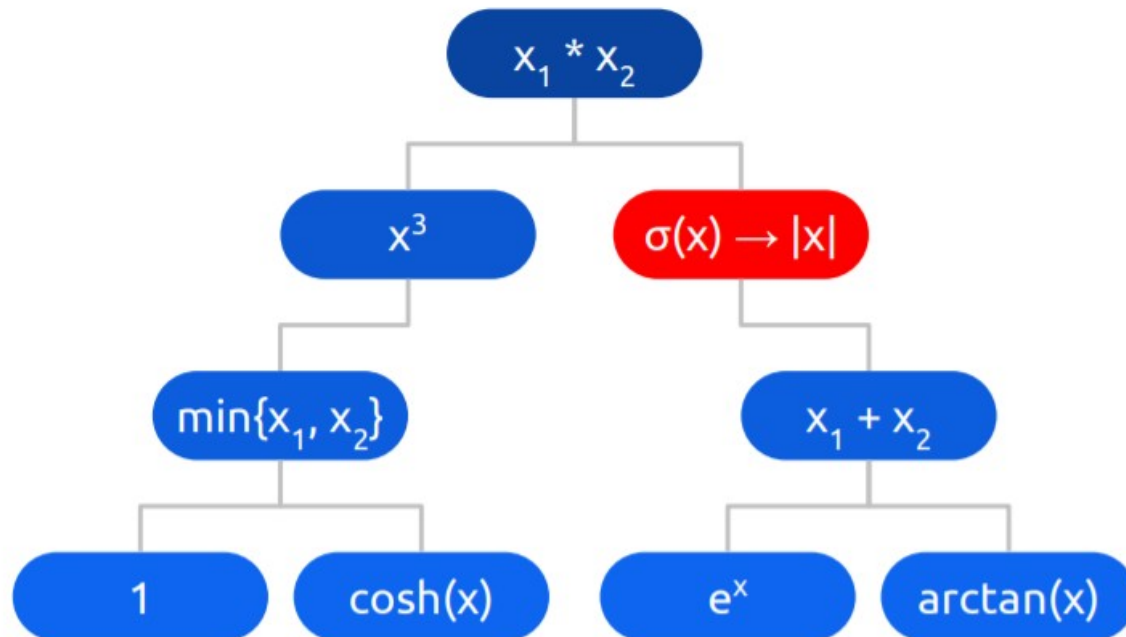
- GP tìm Activation Function tối ưu cho Deeplearning [1]
- Bước 1: Xác định tập kết thúc và tập hàm
 - Tập kết thúc: :
 - Tập hàm: $x_1 + x_2$, $x_1 - x_2$, $x_1 \cdot x_2$, $x_1/(x_2 + \epsilon)$, $\max\{x_1, x_2\}$, $\min\{x_1, x_2\}$

[1] Bingham, Garrett, William Macke, and Risto Miikkulainen. "Evolutionary optimization of deep learning activation functions." *arXiv preprint arXiv:2002.07224* (2020) (GECCO 2020)

Ví dụ minh họa

20

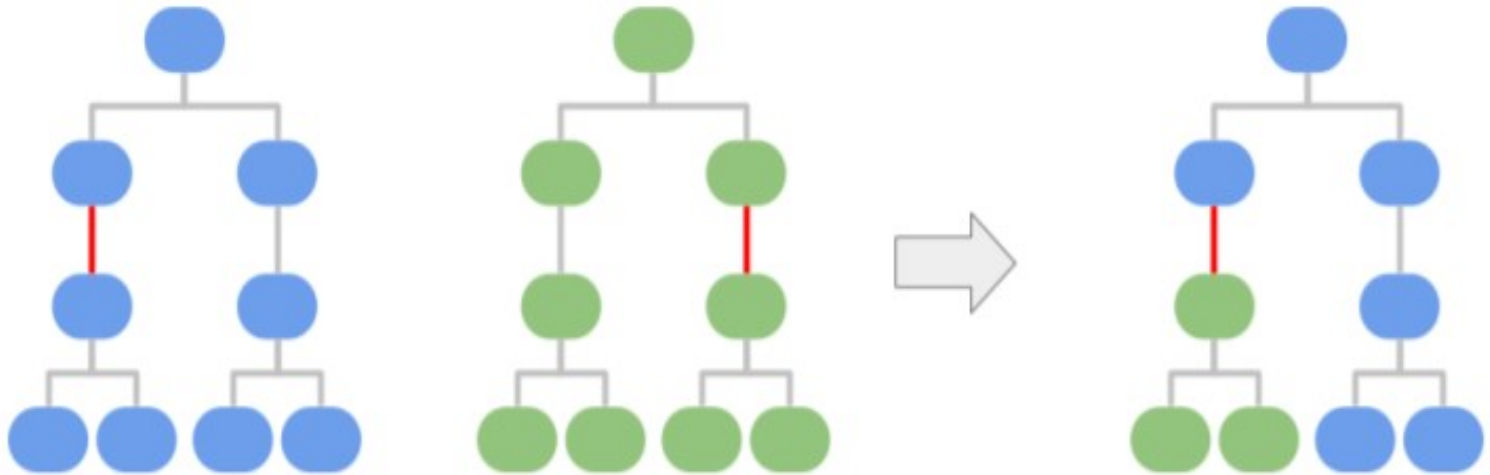
- Đột biến chương trình
-



Ví dụ minh họa

21

■ Lai ghép



Ví dụ minh họa- Kết quả thu được

22

Function Plots

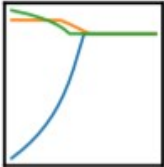


Evolution with Loss-based Fitness (S_2)

- $(e^{(\min\{\text{erf}(x), 0\}) - (\max\{x, 0\})}) * (\min\{(\arctan((x)^3)) * (\max\{|x|, 0\}), 0\})$
- $(e^{\max\{\min\{\text{erf}(x), 0\}, \max\{x, 0\}\}} * (\min\{(\arctan((x)^3)) * (\max\{|x|, 0\}), 0\})$
- $(-((\arctan((x)^3)) * (\cos(1)))) * (-((\arctan(\min\{x, 0\})) * (\max\{|x|, 0\})))$

Accuracy
CIFAR-10 CIFAR-100

94.1 73.9
10.0 01.0
93.9 74.1

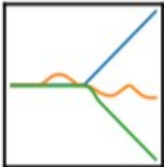


Evolution with Accuracy-based Fitness (S_2)

- $\min\{e^{-(\min\{(\sinh(x))^2, (0)^2\})^2}, \min\{\min\{\text{erf}(\log(1 + e^x)), \text{arcsinh}(x)\}, 0\}\}$
- $\min\{\cos(\max\{(\min\{x, 0\})^3, \log(1 + e^1)\}), e^{-((\max\{x, 0\}) + (e^{\sigma(x)})^2)}\}$
- $\max\{\max\{(\log(|(\min\{x, 0\}) + \epsilon|)) * (\sigma(\text{erf}(x))), 0\}, 0\}$

(*)

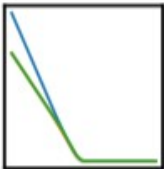
10.0 01.0
93.5 72.5
92.5 72.3



Random Search (S_2)

- $(\min\{\max\{(\log(|(x) + \epsilon|))^3, -(\log(|(x) + \epsilon|))\}, 0\}) + (e^{(\min\{\tanh(x), 0\}) + (\log(|(\max\{x, 0\}) + \epsilon|))}) (*)$
- $(\arctan(\min\{\sinh(\sin(x)), \arctan(\max\{x, 0\})\})) * (\tanh((-(\sin(x))) * (\text{arcsinh}(x))))$
- $(\max\{\frac{\min\{(x)^3, 0\}}{\max\{\sin(x), 0\} + \epsilon}, 0\}) - (\min\{(x)^2, \max\{x, 0\}\})$

93.8 73.9
93.3 72.1
93.9 73.2



Exhaustive Search (S_1)

- $(\arctan(x)) * (\min\{x, 0\})$
- $(\tanh(x)) * (\min\{x, 0\})$
- $(\min\{x, 0\}) * (\text{erf}(x))$

94.0 74.5
94.1 74.3
94.0 74.2



Baseline Activation Functions

- $\text{ReLU}(x)$
- $\text{Swish}(x)$

94.0 73.3
93.8 71.1

Thanks for your attention