

GIẢI THUẬT DI TRUYỀN (GAs) VÀ CÁC ỨNG DỤNG

Ngô Thúy Ngân

Trường Đại học Thủ đô Hà Nội

Tóm tắt: Giải thuật di truyền - Genetic algorithms (GAs) trong lĩnh vực tin học là một trong những giải thuật thú vị, bởi vì nó mô phỏng quy luật đấu tranh sinh tồn của tự nhiên và cũng là một giải thuật vô cùng hiệu quả đối với các loại bài toán tối ưu. Trong bài viết này, tác giả trình bày những đặc điểm cơ bản nhất của Gas.

Từ khóa: Giải thuật di truyền, ứng dụng của Gas, quần thể, tối ưu hóa.

Nhận bài ngày 6.5.2023; gửi phản biện, chỉnh sửa và duyệt đăng ngày 24.7.2023

Liên hệ tác giả: Ngô Thúy Ngân; Email: ntngan@hnmu.edu.vn

1. MỞ ĐẦU

Nhà bác học Charles Darwin đã nêu ra lý thuyết về sự tiến hóa tự nhiên của các loài vật, qua nhiều thế hệ sinh vật phát triển dựa trên nguyên lý của sự chọn lọc tự nhiên “loài nào thích nghi thì sẽ tồn tại”, như ta thấy trong tự nhiên các loài vật sẽ cạnh tranh nhau về nơi trú ẩn, thực phẩm,... các cá thể cùng loài còn cạnh tranh nhau để thu hút bạn tình trong mùa sinh sản do đó những cá thể nào ít thích nghi thì ít có cơ hội tồn tại hơn và những cá thể thích nghi được thì sẽ phát triển và cho ra nhiều con cái.

Trong quá trình sinh sản sẽ tổ hợp các đặc tính tốt từ tổ tiên, sau một vài thế hệ những loài tiến hóa tự nhiên sẽ thích nghi hơn trong môi trường phát triển. Dựa trên nền tảng lý thuyết tiến hóa tự nhiên này, đến năm 1975 Holland đã phát triển ý tưởng này vào hệ thống nhân tạo, ông áp dụng nguyên tắc này để tối ưu hóa các vấn đề và xây dựng thuật toán di truyền (GAs). Hiện nay GAs được xem như một công cụ mạnh mẽ để giải quyết các vấn đề về tìm kiếm và tối ưu hóa phức tạp như thời gian biểu, lập kế hoạch mua sắm,... Trong bài viết này, tác giả nêu ra cách thức hoạt động và các ứng dụng của GAs để giải quyết các bài toán cụ thể.

2. NỘI DUNG

2.1. Giải thuật di truyền - Genetic algorithms (GAs)

GAs 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), là một phân ngành của giải thuật tiến hóa, vận dụng các nguyên lý của tiến hóa như: di truyền, đột biến, chọn lọc tự nhiên, và trao đổi chéo. Nó sử dụng ngôn ngữ máy tính để mô phỏng quá trình tiến hoá của một tập hợp những đại diện trừu tượng (gọi là những nhiễm sắc thể), của các giải pháp có thể (gọi là những cá thể) cho bài toán tối ưu hóa vấn đề. Tập hợp này sẽ tiến triển theo hướng chọn lọc những giải pháp tốt hơn.

GAs cũng như các thuật toán tiến hoá, đều được hình thành dựa trên một quan niệm được coi là một tiên đề phù hợp với thực tế khách quan. Đó là quan niệm “Quá trình tiến hoá tự nhiên là quá trình hoàn hảo nhất, hợp lý nhất và tự nó đã mang tính tối ưu”. Quá trình tiến hoá thể hiện tính tối ưu ở chỗ thế hệ sau bao giờ cũng tốt hơn thế hệ trước.

Ngày nay, GAs càng trở nên quan trọng, đặc biệt là trong lĩnh vực tối ưu hoá, một lĩnh vực có nhiều bài toán thú vị, được ứng dụng nhiều trong thực tiễn nhưng thường khó và chưa có phương pháp hiệu quả để giải quyết.

2.1.1. Các tính chất của giải thuật di truyền

GAs là kỹ thuật chung, giúp giải quyết 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 qui định sẵn của môi trường. Mục tiêu của GAs không nhằm đưa ra lời giải chính xác tối ưu mà là đưa ra lời giải tương đối tối ưu.

Một cá thể trong GAs sẽ biểu diễn một giải pháp của bài toán. Tuy nhiên, không giống với trong tự nhiên là một cá thể có nhiều nhiễm sắc thể (NST) mà để giới hạn trong GAs, ta quan niệm một cá thể có một NST. Do đó, khái niệm cá thể và NST trong GAs coi như là tương đương. Một NST được tạo thành từ nhiều gen, mỗi gen có thể có các giá trị khác nhau để quy định một tình trạng nào đó.

Trong GAs, một gen được coi như một phần tử trong chuỗi NST. Một quần thể là tập hợp các cá thể sinh vật cùng một loài, cùng sống trong một không gian xác định, vào thời điểm nhất định, có lịch sử phát triển chung và cách ly với quần thể cùng loài khác. Trong thuật giải di truyền, ta quan niệm quần thể là một tập các lời giải của một bài toán.

2.1.2. Các bước cơ bản của giải thuật di truyền

Như trong hình 1, ta thấy giải thuật di truyền đơn giản được thực hiện qua các bước cơ bản sau:

Bước 1: [Bắt đầu] Nhận các tham số cho thuật toán.

Bước 2: [Khởi tạo] Sinh ngẫu nhiên một quần thể gồm n cá thể (là n lời giải cho bài toán)

Bước 3: [Quần thể mới] Tạo quần thể mới bằng cách lặp lại các bước sau cho đến khi quần thể mới hoàn thành

- [Thích nghi] Ước lượng độ thích nghi $eval(x)$ của mỗi cá thể.

- [Kiểm tra] Kiểm tra điều kiện kết thúc giải thuật.

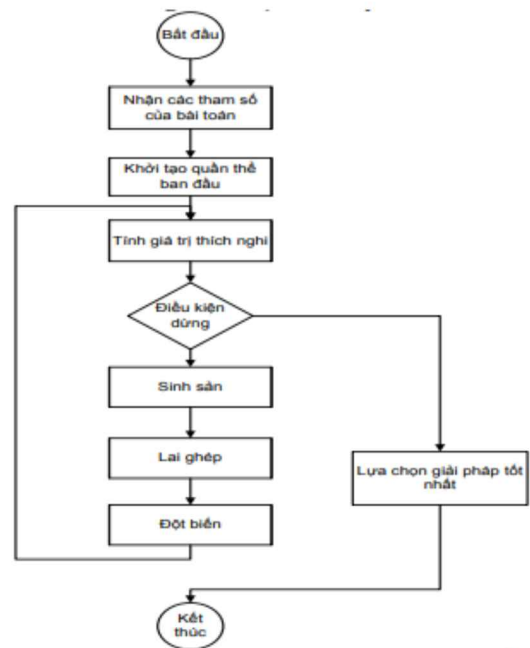
- [Chọn lọc] Chọn hai cá thể bố mẹ từ quần thể cũ theo độ thích nghi của chúng (cá thể có độ thích nghi càng cao thì càng có nhiều khả năng được chọn)

- [Lai ghép] Với một xác suất lai ghép được chọn, lai ghép hai cá thể bố mẹ để tạo ra một cá thể mới.

- [Đột biến] Với một xác suất đột biến được chọn, biến đổi cá thể mới

Bước 4. [Chọn kết quả] Nếu điều kiện dừng được thỏa mãn thì thuật toán kết thúc và trả về lời giải tốt nhất trong quần thể hiện tại.

GAs có hai loại điều kiện dừng cơ bản (1) dựa trên cấu trúc nhiễm sắc thể, kiểm soát số gen được hội tụ, nếu số gen hội tụ vượt quá số phần trăm nào đó của tổng số gen, việc tìm kiếm



Hình 1: Sơ đồ thực hiện giải thuật di truyền đơn giản

sẽ kết thúc; (2) dựa trên ý nghĩa đặc biệt của một nhiễm sắc thể, đo tiến bộ của giải thuật trong một số thế hệ cho trước, nếu tiến bộ này nhỏ hơn một hằng số ε xác định, kết thúc tìm kiếm.

2.2. Nguyên lý hoạt động

Nền tảng lý thuyết của GAs dựa trên biểu diễn chuỗi nhị phân và lý thuyết sơ đồ. Một sơ đồ là một chuỗi, dài bằng chuỗi nhiễm sắc thể, các thành phần của nó có thể nhận một trong các giá trị của tập ký tự biểu diễn gen hoặc một ký tự đại diện “*”. Sơ đồ biểu diễn một không gian con của không gian tìm kiếm. Không gian con này là tập tất cả các chuỗi trong không gian lời giải mà với mọi vị trí trong chuỗi giá trị của gen trùng với giá trị của sơ đồ [1]

Ví dụ: các chuỗi và sơ đồ có chiều dài 10.

Sơ đồ (*111100100) sẽ khớp với hai chuỗi: {(0111100100), (1111100100)}

Và sơ đồ (*1*1100100) sẽ khớp với 4 chuỗi {(0101100100), (0111100100), (1101100100), (1111100100)}

Đương nhiên, sơ đồ (1001110001) chỉ khớp với chính nó, và sơ đồ (*****) khớp với tất cả các chuỗi có chiều dài 10. Rõ ràng là mỗi sơ đồ cụ thể có tương ứng 2^r chuỗi, với r là số ký tự đại diện “*” có trong sơ đồ. Mặc khác, mỗi chuỗi chiều dài m sẽ khớp với 2^m sơ đồ. Một chuỗi chiều dài m , sẽ có tối đa 2^m sơ đồ. Trong một quần thể kích thước n , có thể có tương ứng từ 2 m đến $n \times 2^m$ sơ đồ khác nhau. Các sơ đồ khác nhau có những đặc trưng khác nhau. Các đặc trưng này thể hiện qua hai thuộc tính quan trọng bậc và chiều dài xác định.

Bậc của sơ đồ S (ký hiệu $\sigma(S)$) là chiều dài của chuỗi trừ đi số ký tự đại diện. Bậc xác định đặc trưng của sơ đồ.

Khái niệm bậc của sơ đồ giúp cho việc tính xác suất sống còn của sơ đồ do ảnh hưởng của đột biến. Chiều dài xác định của sơ đồ S (ký hiệu là $\delta(S)$) là khoảng cách giữa hai vị trí cố định ở đầu và cuối. Nó định nghĩa “độ nén” của thông tin chứa trong một sơ đồ.

Ví dụ: $\delta(S1) = 10 - 4 = 6$; $\delta(S2) = 9 - 5 = 4$; $\delta(S3) = 10 - 1 = 9$

Như vậy, một sơ đồ chỉ có một vị trí cố định duy nhất thì sẽ có chiều dài xác định là 0. Khái niệm chiều dài xác định của sơ đồ giúp tính xác suất sống còn của sơ đồ do ảnh hưởng của phép lai.

GAs sử dụng một quần thể của các lời giải có thể. Mỗi lời giải được đại diện bởi một NST, nó chỉ là một đại diện trừu tượng. Các NST được mã hóa thành các chuỗi nhị phân, mỗi vị trí trên chuỗi tồn tại hai giá trị là “1” hoặc “0”. Chẳng hạn như: 1 0 0 1 0 1 0 1 0.

Độ tốt của một cá thể được đánh giá bằng hàm mục tiêu $g(x)$ với x là một NST. Hàm mục tiêu $g(x)$ sau khi được tính toán sẽ là cơ sở để đánh giá độ thích nghi của cá thể. Hàm thích nghi $f(x)$ là sẽ quyết định khả năng một cá thể được chọn lọc vào thế hệ sau, việc ánh xạ $g(x) \rightarrow f(x)$ có nhiều phương pháp ánh xạ khác nhau phụ thuộc vào mục đích của bài toán

2.3. So sánh GAs với kỹ thuật tối ưu khác

Hoạt động của GAs đơn giản là việc mô phỏng sự tiến hóa và chọn lọc tự nhiên bằng máy tính bắt đầu từ một quần thể ngẫu nhiên. Bên cạnh đó để tối ưu ta cần hàm lượng giá hoặc hàm thích nghi để chọn cá thể tốt và loại bỏ cá thể xấu.

Thuật toán di truyền (GAs) khác với kỹ thuật tối ưu khác ở chỗ [2]:

- GAs làm việc với bộ mã của biến chứ không phải làm việc trực tiếp trên biến.

- Hầu hết các kỹ thuật tối ưu thông thường tìm kiếm từ một đỉnh, trong khi đó GAs luôn hoạt động trên tập hợp đỉnh (điểm tối ưu), điều này là một ưu điểm của GAs giúp tăng cơ hội tiếp cận tối ưu toàn cục và tránh hội tụ sớm tại điểm cục bộ địa phương.

- GAs đánh giá hàm mục tiêu để phục vụ quá trình tìm kiếm, vì vậy có thể ứng dụng cho bất kì bài toán tối ưu nào (liên tục hay rời rạc).

- GAs thuộc lớp các thuật toán xác suất, các thao tác cơ bản của GAs dựa trên khả năng tích hợp ngẫu nhiên trong quá trình xử lý.

2.4. Ứng dụng của GAs

GAs được sử dụng cho những bài toán khó, và đã được ứng dụng thành công cho một số bài toán như: lập kế hoạch, điều khiển tương thích, chương trình trò chơi, các bài toán vận tải, bài toán người đi du lịch. Sau đây là một vài ứng dụng tiêu biểu của GAs [1].

a. Bài toán người du lịch (TSP)

TSP được mô tả như sau: Một du khách muốn thăm những thành phố anh quan tâm; mỗi thành phố thăm qua đúng một lần; rồi trở về điểm khởi hành. Biết trước chi phí di chuyển giữa hai thành phố bất kỳ. Yêu cầu của bài toán là xây dựng một lộ trình thỏa các điều kiện trên với tổng chi phí nhỏ nhất.

TSP là bài toán tối ưu tổ hợp, không gian tìm kiếm là tập các hoán vị của n thành phố. Bất cứ hoán vị nào của n thành phố cũng là một lời giải chấp nhận được. Lời giải tối ưu là một hoán vị với chi phí tối thiểu của hành trình. Không gian tìm kiếm là $n!$. Có thể giải bài toán này bằng nhiều phương pháp: phương pháp nhánh cận, phương pháp gần đúng hay những phương pháp tìm kiếm Heuristic. Phương pháp nhánh cận đã được chứng minh đạt sự tối ưu về lời giải, tuy nhiên phương pháp này lại mất khá nhiều thời gian khi số đỉnh của đồ thị lớn.

Trong những năm gần đây, đã xuất hiện nhiều thuật toán đạt gần đến lời giải tối ưu của bài toán TSP: láng giềng gần nhất, đảo gần nhất, đảo xa nhất... và TSP cũng trở thành một đích ngắm của cộng đồng GAs.

Với bài toán này chúng ta sẽ đánh số các thành phố và dùng một vector nguyên để biểu diễn một NST lộ trình $v = \langle i_1, i_2, \dots, i_n \rangle$ biểu diễn một lộ trình: từ i_1 đến $i_2, \dots$, từ i_{n-1} đến i_n và trở về i_1 (v là một hoán vị của vector), hàm lượng giá chính là chi phí của lộ trình.

b. Bài toán lập lịch

Lập lịch là bài toán tổ chức sản xuất. Một công ty cần sản xuất nhiều loại hàng hóa; những hàng hóa này có thể được sản xuất theo những kế hoạch khác nhau. Mỗi kế hoạch xử lý gồm một chuỗi thao tác; những thao tác này sử dụng một số tài nguyên và cần thời gian chạy máy. Một lịch sản xuất là một kế hoạch thực hiện các đơn đặt hàng. Trong đó, một số đơn đặt hàng có thể được thực hiện với cùng những thao tác tương đương. Nhiệm vụ là lên kế hoạch, lập lịch sản xuất, để thực hiện các đơn đặt hàng này nhanh nhất có thể.

Bài toán lập lịch là chọn một chuỗi các thao tác đồng thời chỉ định về thời gian bắt đầu/kết thúc và các tài nguyên cần thiết cho mỗi thao tác. Điều cần quan tâm chính yếu là chi phí thời gian máy rồi, năng lực lao động và các đơn đặt hàng cần hoàn thành đúng hạn. Ý tưởng chính trong phương pháp là mã hóa biểu diễn của lịch phân công là các toán tử di truyền phải thực hiện theo cách có ý nghĩa, và một bộ giải mã phải luôn tạo ra một lời giải hợp lệ cho bài toán. Thủ tục giải mã mô phỏng các thao tác của công việc theo cách mà khi một máy tính sẵn sàng chọn lựa, thì thao tác cho phép đầu tiên từ danh sách ưu tiên được lấy ra.

3. KẾT LUẬN

Bài toán tối ưu là bài toán tìm kiếm giải pháp (tốt nhất) trong không gian (vô cùng lớn) các giải pháp. Khi không gian tìm kiếm nhỏ, các phương pháp cổ điển cũng đủ thích hợp; nhưng khi không gian lớn hơn thì cần phải dùng đến những kĩ thuật đặc biệt, GAs là một trong những kĩ thuật đó. GAs là hệ thống đầu tiên dựa trên mô phỏng sự sống. Thông qua chọn lọc, lai ghép,

đột biến, GAs sẽ hội tụ qua nhiều thế hệ theo hướng tối ưu toàn cục. GAs hi vọng hướng tới một giải pháp tối ưu hơn bằng cách kết hợp thông tin tốt ẩn trong tập hợp các giải pháp, để tạo ra giải pháp mới với những thông tin tốt thừa hưởng từ cả cha và mẹ. Khả năng của thuật toán vẫn đang còn được khám phá và khai thác, một số lượng lớn những nghiên cứu thành công về lý thuyết cũng như ứng dụng trong nhiều lĩnh vực thế giới thực đã chứng minh rằng GAs là một kỹ thuật tối ưu mạnh mẽ và thiết thực.

TÀI LIỆU THAM KHẢO

1. N. Đ. Thúc (2001). *Trí tuệ nhân tạo-Lập trình tiến hóa*. Nxb. Giáo Dục.
2. S. N. Sivanandam, S. N. Deepa (2008). *Introduction to Genetic Algorithms*. Springer.
3. Freisleben, P. Mers (1996). *New genetic local search operator for the travelling*. University of Siegen

GENETIC ALGORITHMS (GAs) AND APPLICATIONS

Abstract: Genetic algorithms (GAs) in the field of informatics is one of the interesting algorithms, because it simulates the laws of nature's struggle for survival and is also an extremely effective algorithm for all kinds of problems. optimization math. In this article, the author presents the most basic characteristics of Gas.

Keywords: Genetic algorithms, Applications of Gas, Population, Optimization.