

KẾT HỢP GIẢI THUẬT DI TRUYỀN VÀ TÌM KIẾM TABU GIẢI BÀI TOÁN TỐI ƯU ĐA MỤC TIÊU

Vũ Mạnh Xuân*

Trường Đại học Sư phạm – ĐH Thái Nguyên

TÓM TẮT

Nhiều bài toán quyết định trong đời sống là đa mục tiêu. Đặc tính tự nhiên của quyết định đa mục tiêu là: có nhiều hơn một mục tiêu và cái đích của các mục tiêu không giống nhau, thậm chí xung đột nhau. Vì vậy tối ưu đa mục tiêu luôn là bài toán khó và thời sự. Mục đích của tối ưu đa mục tiêu là sinh ra một danh sách các lời giải gọi là tập pareto. Các thuật toán tiến hóa thường tỏ ra có hiệu quả trong việc giải bài toán MOPs bởi các kết quả thu được là đa dạng và gần với tập nghiệm tối ưu. Bài báo này trình bày phương pháp kết hợp Giải thuật di truyền và giải thuật tìm kiếm Tabu giải bài toán tối ưu đa mục tiêu. Kết quả của các phương pháp này được kiểm nghiệm qua việc test một số bài toán cụ thể.

Từ khóa: thuật toán di truyền; tìm kiếm Tabu; tối ưu hóa đa mục tiêu

ĐẶT VẤN ĐỀ

Lớp bài toán tối ưu đa mục tiêu có nhiều ứng dụng thực tế, nhất là trong thiết kế. Chẳng hạn người ta muốn thiết kế sản phẩm sao cho chi phí thấp, tiết kiệm nguyên liệu nhưng chất lượng tốt, ... Đã có nhiều phương pháp giải khác nhau được đề xuất. Tuy nhiên lời giải của bài toán tối ưu đa mục tiêu nói chung khó xác định và không duy nhất. Việc giải lớp bài toán này thường chỉ là đưa ra nhiều phương án để người sử dụng chọn lựa. Nói cách khác, các phương pháp giải bài toán tối ưu đa mục tiêu thường dẫn đến hai mục đích sau [1], [2], [5], [8]:

- Đưa ra nhiều lời giải Pareto để người sử dụng chọn.
- Tập lời giải này càng đa dạng càng tốt.

Do đặc thù của giải thuật di truyền (GA: Genetic Algorithm) là làm việc trên một quần thể (tập phương án) nên đã có nhiều nghiên cứu áp dụng GA vào giải bài toán tối ưu đa mục tiêu. Tuy nhiên việc tích hợp nhiều kỹ thuật tính toán thông minh nhằm nâng cao hiệu suất tính toán vẫn còn là vấn đề thời sự và có ý nghĩa khoa học.

Bài báo này đề cập đến việc tích hợp giải thuật di truyền và tìm kiếm Tabu (TS: Tabu Search) giải bài toán tối ưu đa mục tiêu nhằm tăng tính đa dạng của quần thể.

Bài báo có cấu trúc sau: Phần thứ nhất nêu bài toán tối ưu đa mục tiêu, tổng quan về GA và TS. Phần thứ hai trình bày giải pháp kết hợp GA và TS giải bài toán tối ưu đa mục tiêu và minh họa trên một số bài toán cụ thể.

MỘT SỐ KIẾN THỨC LIÊN QUAN

Bài toán tối ưu đa mục tiêu có dạng:

$$F(x) \rightarrow \max (\min)$$

$$x \in D \subset \mathbb{R}^n$$

Trong đó $F(x) = (f_1(x), \dots, f_k(x)) \in \mathbb{R}^k$ gọi là vector mục tiêu.

X gọi là phương án; D gọi là tập các phương án; f_1, \dots, f_k là các hàm mục tiêu.

Phương án x^* là nghiệm của bài toán nếu $F(x^*) \leq F(x)$ với mọi $x \in D$. Tuy nhiên hầu như không có nghiệm như vậy và người ta sử dụng khái niệm nghiệm Pareto sau:

(i) Cho phương án $x, y \in D$. Khi đó, x được gọi là **trội hơn** y (kí hiệu $x \prec y$), nếu ta có: $F(x) \leq F(y)$ và $F(x) \neq F(y)$; y còn được gọi là **được trội bởi** x . Nếu ngược lại, y được gọi là **không bị trội bởi** x .

(ii) Một phương án $x \in D$ được gọi là **nghiệm Pareto tối ưu** (hay **nghiệm Pareto**) nếu không có $y \in D$ mà y trội hơn x . Tập tất cả các nghiệm Pareto gọi là **tập Pareto**.

Rõ ràng nếu bài toán tối ưu đa mục tiêu có nghiệm thì nghiệm đó phải là một nghiệm Pareto.

Trên thực tế, việc tìm tập lời giải Pareto của các bài toán tối ưu đa mục tiêu là khó khăn và

* Tel: 0912 700396, Email: vmxuan08@gmail.com

thường ít thực hiện được. Vì vậy, một số chiến lược tìm kiếm ngẫu nhiên (như thuật toán tiến hóa, phương pháp vùng cấm, mô phỏng luyện kim,...) đã được phát triển. Mặc dù các chiến lược này thường không đảm bảo xác định chính xác tập tối ưu Pareto, nhưng đều cố gắng tìm ra một tập xấp xỉ tốt.

Giải thuật di truyền (GA) là một giải thuật tìm kiếm lời giải của bài toán tối ưu dựa trên sự mô phỏng quá trình tiến hóa của tự nhiên. Xuất phát từ một quần thể (tập lời giải ban đầu), giải thuật tiến hành quá trình tiến hóa dựa trên ba toán tử di truyền là lai ghép (crossover), đột biến (mutation) và chọn lọc (selection) nhằm tạo ra thế hệ mới "tốt hơn" thế hệ trước đó. Ban đầu GA được sử dụng để giải bài toán tối ưu một mục tiêu, song với sự phát triển của nó cũng như nhu cầu thực tế, nhiều công trình nghiên cứu sử dụng GA cho tối ưu đa mục tiêu đã được công bố [5], [6], [8].

Tìm kiếm Tabu (TS) là một kỹ thuật tìm kiếm dựa trên quy định về luật cấm kết hợp đối với các cá thể có "quan hệ gần" nhằm tránh suy thoái và tăng tính đa dạng của quần thể. Với ý nghĩa đó, việc kết hợp GA và TS nhằm nâng cao hiệu suất tính toán và đạt được hai mục đích giải bài toán tối ưu đa mục tiêu nêu trên, nhất là mục đích tăng tính đa dạng của quần thể là có ý nghĩa khoa học và thực tiễn [3], [7].

GIẢI PHÁP KẾT HỢP GA VÀ TS

Nói chung để sử dụng GA trong tối ưu đa mục tiêu, ta thường quy bài toán tối ưu đa mục tiêu về bài toán tối ưu một mục tiêu. Trong bài báo này chúng tôi sử dụng phương pháp trọng số: gán mỗi hàm mục tiêu f_i với một trọng số w_i , khi đó bài toán đưa về dạng tìm Min $F(x)$ với $F(x) = \sum w_i f_i(x)$. Các trọng số w_i có thể cho trước bởi người sử dụng song cũng có thể tự sinh ra trong quá trình tiến hóa.

Tư tưởng của giải thuật kết hợp TS và GA nhằm tăng tính đa dạng của quần thể bằng cách sau: khi chọn cá thể cha mẹ để tiến hành lai ghép tại mỗi bước tiến hóa, ta tránh không chọn các cá thể "gần nhau" theo một nghĩa nào đó. Do các cá thể con tiếp nhận và kế thừa nhiều đặc tính của các cá thể cha mẹ, nên khi các cá thể cha mẹ không "gần nhau"

thì rõ ràng tính đa dạng của quần thể được tăng cường. Điều khó khăn cơ bản là xác định khái niệm "gần nhau" như thế nào, đặc biệt là nếu không cẩn thận thì quá trình xử lý trung gian này lại chiếm quá nhiều thời gian làm hạn chế thuật toán.

Chúng tôi đề xuất hai kỹ thuật để đánh giá hai cá thể như thế nào là "gần nhau" như sau:

Phương án 1. Mỗi cá thể từ khi sinh ra đều được đánh dấu độ tuổi và mỗi khi tiến hành lai ghép để tạo ra thế hệ kế tiếp, độ tuổi này được tăng tương ứng. Bằng cách đó ta có thể ngăn cấm các cá thể quá "trẻ" hoặc "cùng tuổi" kết hợp với nhau.

Để thuận tiện khi lập trình, có thể làm đơn giản bằng cách thêm vào một thuộc tính "tuổi" cho mỗi cá thể. Khi bắt đầu khởi tạo hoặc khi được sinh ra, giá trị ban đầu được gán là 1 và được tăng lên sau mỗi lần tiến hóa. Khi chọn cá thể cha mẹ để tiến hành lai ghép, ta chỉ chọn các cá thể khác nhau trên thuộc tính này hoặc có giá trị lớn hơn một "ngưỡng" nào đó.

Thuật toán cụ thể như sau:

B1) Khởi tạo ngẫu nhiên quần thể ban đầu Q_0

Gán "tuổi" mỗi cá thể là 1; $t = 0$;

B2) Thực hiện quá trình tiến hóa:

2.1. $t = t + 1$;

2.2. Chọn từ Q_t các cá thể cha mẹ X và Y sao cho Tuổi(X) khác Tuổi(Y);

2.3. Lai ghép X và Y sinh ra các con C_1 và C_2

2.4. Đột biến C_1 hoặc C_2 theo xác suất P_m .

2.5. Chọn từ tập $\{X, Y, C_1, C_2\}$ hai cá thể thay cho X và Y từ Q_t .

Nếu thỏa điều kiện dừng thì chuyển đến B3), nếu không thì quay lại B2).

B3) Đưa ra các cá thể không trội được trong Q_t để người sử dụng chọn lựa.

Phương án 2. Độ "gần nhau" giữa hai cá thể có thể được đo bởi khoảng cách Hamming tính theo công thức sau:

$$d(x,y) = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - y_i)^2}.$$

Khi chọn hai cá thể cha mẹ, ta chỉ cho chúng lai ghép với nhau nếu khoảng cách giữa chúng không nhỏ hơn một "ngưỡng" nào đó.

Như vậy thuật toán này chỉ bổ sung cách chọn cá thể để tiến hành lai ghép. Để đơn giản có thể làm như sau: Chọn ngẫu nhiên một cá thể X; tính khoảng cách cực đại giữa cá thể này với các cá thể khác của quần thể chẳng hạn là d. Tiến hành chọn cá thể thứ hai Y sao cho khoảng cách X đến Y không nhỏ hơn d/2. Khi đó mới tiến hành lai ghép X với Y.

Thuật toán cụ thể như sau:

B1) Khởi tạo ngẫu nhiên quần thể ban đầu Q_0

$t = 0$;

B2) Thực hiện quá trình tiến hóa:

2.1. $t = t + 1$;

2.2. Chọn các cá thể cha mẹ X và Y theo cách sau:

2.2.1. Chọn cá thể X

2.2.2. Tính khoảng cách lớn nhất giữa X và các cá thể khác trong Q_t , ký hiệu là d.

2.2.3. Chọn cá thể Y sao cho khoảng cách X và Y lớn hơn d/2.

2.3. Lai ghép X và Y sinh ra các con C_1 và C_2

2.4. Đột biến C_1 hoặc C_2 theo xác suất P_m .

2.5. Chọn từ tập $\{X, Y, C_1, C_2\}$ hai cá thể thay cho X và Y từ Q_t .

Nếu thỏa điều kiện dừng thì chuyển đến B3), nếu không thì quay lại B2).

B3) Đưa ra các cá thể không trội được trong Q_t để người sử dụng chọn lựa.

THỬ NGHIỆM

Xét một số bài toán sau [4], [8]:

Bài toán

Các hàm sau có dạng chung tương tự nhau nhưng các hàm thành phần khác nhau và do đó dạng tập Pareto cũng khác nhau. Cấu trúc chung như sau:

$\text{Min } T(x) = (f_1(x_1), f_2(x))$.

trong đó $f_2(x) = g(x_2, \dots, x_n) \cdot h(f_1(x_1), g(x_2, \dots, x_n))$.

$x = (x_1, \dots, x_n) \in D$.

Các hàm mục tiêu cụ thể như sau:

Bài toán T1:

$$f_1(x_1) = x_1$$

$$g(x_2, \dots, x_n) = 1 + 9 \sum_{i=2}^n \frac{x_i}{n-1}$$

$$h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}}$$

Với $n = 30$; $x_i \in [0, 1]$; mặt tối ưu Pareto lồi ứng với $g(x) = 1$.

Bài toán T2:

$$f_1(x_1) = x_1$$

$$g(x_2, \dots, x_n) = 1 + 9 \sum_{i=2}^n \frac{x_i}{n-1}$$

$$h(f_1, g) = 1 - \left(\frac{f_1}{g} \right)^2$$

Với $n = 30$; $x_i \in [0, 1]$; mặt tối ưu Pareto không lồi ứng với $g(x) = 1$.

Bài toán T3:

$$f_1(x_1) = x_1$$

$$g(x_2, \dots, x_n) = 1 + 9 \sum_{i=2}^n \frac{x_i}{n-1}$$

$$h(f_1, g) = 1 - \sqrt{\frac{f_1}{g}} - \left(\frac{f_1}{g} \right) \sin(10\pi f_1)$$

Với $n = 30$; $x_i \in [0, 1]$; mặt tối ưu Pareto rời rạc gồm với một số phần không lồi ứng với $g(x) = 1$.

Để tiện so sánh, với mỗi bài toán trên, chúng tôi cùng sử dụng một quần thể khởi tạo ban đầu, chương trình tiến hành theo ba dạng tiến hóa khác nhau: tiến hóa bằng GA thông thường, tiến hóa bằng GA kết hợp TS dạng 1 và tiến hóa bằng GA kết hợp TS dạng 2.

Chương trình thiết lập sử dụng lai ghép SBX với phân phối mũ, đột biến đều, các tham số được sử dụng trong cả hai bài toán này là:

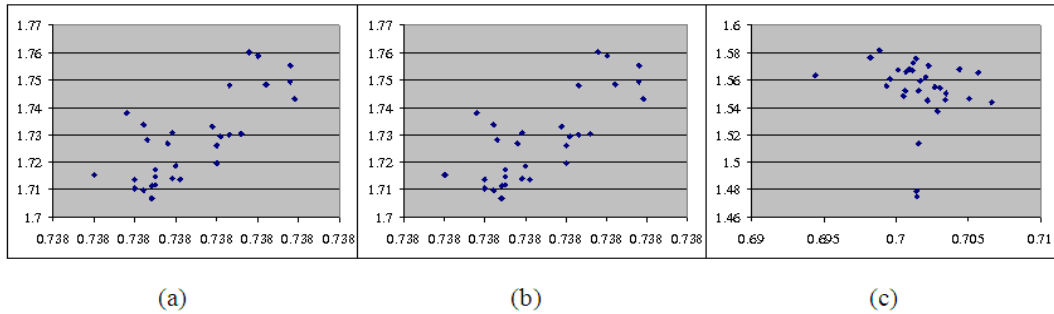
- Kích thước quần thể: 30
- Xác suất lai ghép 1, đột biến 0.1.
- Chương trình thực hiện tiến hoá sau 1000 thế hệ với mỗi dạng nêu trên.

Kết quả thực hiện bài toán T1

Bảng 1. Giá trị hàm trung bình ở mỗi dạng tiến hóa bài toán T1

Hàm	Quần thể khởi tạo	Tiến hóa thường	GA+TS dạng 1	GA+TS dạng 2
f_1	0.55744	0.73784	0.61182	0.70159
f_2	3.8638	1.7283	1.589	1.5526

Sự phân bố giá trị hàm của quần thể cuối cùng được thể hiện trong hình 1.

**Hình 1.** Phân bố giá trị hàm mục tiêu bài toán T1

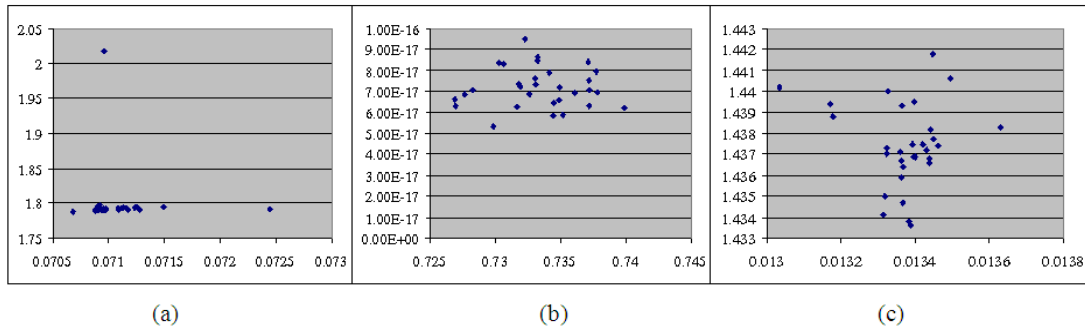
a: GA thông thường; b: GA – TS dạng 1; c: GA – TS dạng 2.

Kết quả thực hiện bài toán T2

Bảng 2. Giá trị hàm trung bình ở mỗi dạng tiến hóa bài toán T2

Hàm	Quần thể khởi tạo	Tiến hóa thường	GA+TS dạng 1	GA+TS dạng 2
f_1	0.52922	0.071077	7.18E-17	0.013372
f_2	5.3999	1.7997	1	1.4374

Sự phân bố giá trị hàm của quần thể cuối cùng được thể hiện trong hình 2.

**Hình 2.** Phân bố giá trị hàm mục tiêu bài toán T2

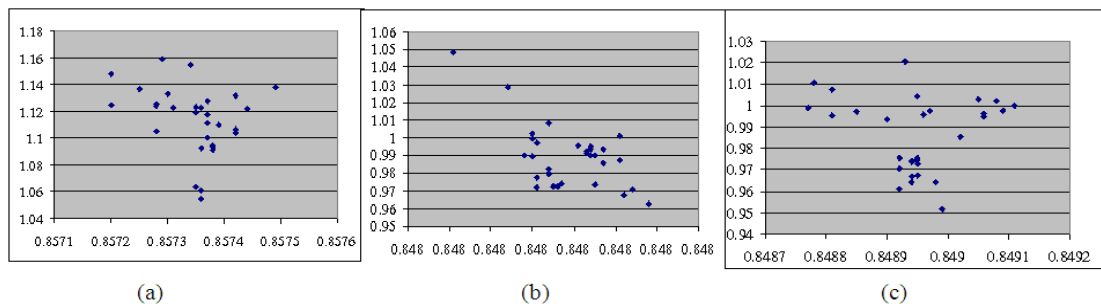
a: GA thông thường; b: GA – TS dạng 1; c: GA – TS dạng 2.

Kết quả thực hiện bài toán T3

Bảng 3. Giá trị hàm trung bình ở mỗi dạng tiến hóa bài toán T3

Hàm	Quần thể khởi tạo	Tiến hóa thường	GA+TS dạng 1	GA+TS dạng 2
f_1	0.4951	0.8573	0.84839	0.84895
f_2	3.9933	1.1139	0.98956	0.98643

Sự phân bố giá trị hàm của quần thể cuối cùng được thể hiện trong hình 3.

**Hình 3.** Sự phân bố giá trị hàm của quần thể cuối cùng
a: GA thông thường; b: GA+TS dạng 1; c: GA+TS dạng 2

Xem bảng giá trị trung bình của các hàm mục tiêu, ta thấy nói chung phương pháp kết hợp giới thiệu ở trên về cơ bản là tốt hơn GA thông thường. Tuy nhiên để có thể kết luận thì còn phải tiến hành thử nghiệm nhiều hơn nữa. Chúng tôi cũng đã tiến hành thử nghiệm trên một vài bài toán Benchmark và cũng cho thấy kết quả tương tự.

KẾT LUẬN VÀ THẢO LUẬN

Phát triển các kỹ thuật tính toán trong việc giải bài toán tối ưu đa mục tiêu là công việc có ý nghĩa khoa học và ứng dụng. Sử dụng các kỹ thuật tính toán tiến hoá là một hướng đi hứa hẹn nhiều triển vọng. Tuy nhiên có thể sẽ có kết quả tốt hơn nếu kết hợp một số kỹ thuật tính toán thông minh khác như kết hợp với tìm kiếm Tabu nhằm nâng cao hiệu suất tính toán và làm mềm dẻo hơn quá trình xử lý. Bài báo này mới chỉ đề cập đến hai kỹ thuật xác định “họ hàng gần” của cá thể, tuy nhiên còn có nhiều khả năng khác đặc biệt là vấn đề xác định Tabu List vẫn có thể phát triển.

Chúng tôi cũng cho rằng việc xác định vấn đề “họ hàng gần” giữa các cá thể của quần thể

cũng có thể được cho bởi một khái niệm mờ, như vậy thì chương trình sẽ mềm dẻo hơn.

TÀI LIỆU THAM KHẢO

- [1]. Bùi Minh Trí, Quy hoạch toán học, Nxb Khoa học Kỹ thuật, 2006.
- [2]. Nguyễn Đình Thúc, Lập trình tiến hoá.
- [3]. Glover, F. and Laguna, M., (1997). Tabu Search, Kluwer Academic Publishers, Boston; www.dei.unipd.it/~tabu_search_glover_laguna.pdf
- [4]. Eckart Zitzler, K. Deb, Lothar Thiele, (2000), Evolutionary Computation 8(2), 173-195.
- [5]. K. Deb (1999), Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems, Evolutionary Computation, 7(3), 205-230.
- [6]. K. Deb, A. Pratap, S. Moitra (2000), Mechanical Component Design for Multiple Objectives Using Elitist Non-Dominated Sorting GA, Technical Report No. 200002.
- [7]. Laguna, M. (2005), Implementing and Testing the Tabu Cycle and Conditional Probability Methods, <http://leeds-faculty.colorado.edu/laguna/articles/tabucycle.html>
- [8]. Oliver Schuetze, Gustavo Sanchez, Carlos A. Coello Coello, (2008) A new memetic strategy for the numerical treatment of multi-objective optimization problems, Gecco'08.

SUMMARY

COMBINE GENETIC ALGORITHM AND TABU SEARCH TO SOLVE MULTI-OBJECTIVE OPTIMIZATION PROBLEMS

Vu Manh Xuan*

College of Education - TNU

Many decision problems in the real life are multi-objective ones. The natural characteristics of multi-objective decision are: there are more than one objectives, and the estates of every objective is not in the same and there's conflict between them. Therefore, multi-objective optimization problems (MOPs) is always the hot and hard point of optimization. The aim of MOPs is to generate a list of solutions for problems called the pareto set. Evolutionary algorithms can efficiently solve MOPs by obtaining diverse and near-optimal solution sets. In this paper, we present methods to combine Genetic Algorithm and Tabu Search to solve multi-objective optimization problems. The result of those methods was verified by testing some concrete problems.

Keyword: *Genetic Algorithm; Tabu Search; Multi-objective Optimization*

Ngày nhận bài: 25/7/2012, ngày phản biện: 08/8/2012, ngày duyệt đăng: 10/12/2012

* Tel: 0912 700396, Email: vmxuan08@gmail.com