

Thuật toán tiến hóa đa nhiệm tự thích ứng

Lê Tiến Thành

Giáo viên hướng dẫn: **PGS.TS Huỳnh Thị Thanh Bình**

Trường Đại học Bách Khoa Hà Nội

Mục lục

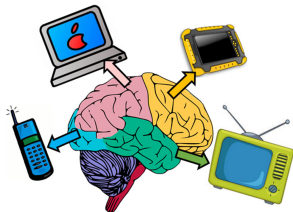
- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - Mô hình Multi-Armed Bandits
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

Mục lục

- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - Mô hình Multi-Armed Bandits
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

Ý tưởng chính của tiến hóa đa nhiệm

Ý tưởng

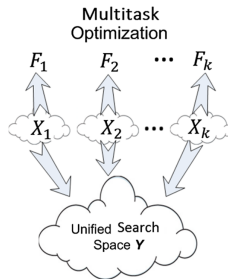


Hình 1: Khả năng đa nhiệm của con người

Xu hướng

Thuật toán mô phỏng trí thông minh trong tự nhiên.

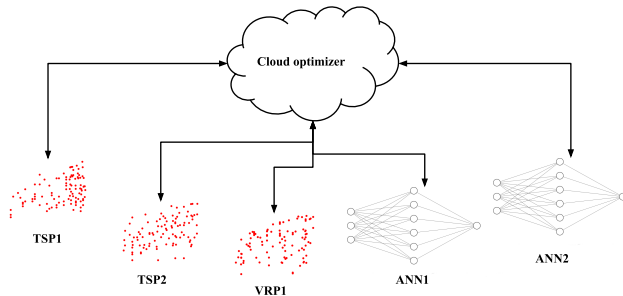
Thuật toán tiến hóa đa nhiệm



Hình 2: Tiến hóa đa nhiệm mô phỏng lại khả năng giải quyết nhiều việc cùng một thời điểm của con người

Ứng dụng nổi bật của tiến hóa đa nhiệm

Tối ưu tại cloud computing



Các nghiên cứu đã áp dụng tiến hóa đa nhiệm

- Huynh Thi Thanh Binh, Pham Dinh Thanh, Tran Ba Trung, and Le Phuong Thao. "Effective multifactorial evolutionary algorithm for solving the cluster shortest path tree problem". In: *2018 IEEE Congress on Evolutionary Computation (CEC)*. IEEE. 2018, pp. 1–8
- Rohitash Chandra, Abhishek Gupta, Yew-Soon Ong, and Chi-Keong Goh. "Evolutionary multi-task learning for modular knowledge representation in neural networks". In: *Neural Processing Letters* 47.3 (2018), pp. 993–1009

Câu hỏi nghiên cứu còn tồn tại

Các tồn tại của các nghiên cứu trước

- Ứng dụng hướng đến làm thuật toán tối ưu trên cloud.
- Cloud có **số lượng người dùng lớn**.
- Chỉ thử nghiệm và chứng minh tính hiệu quả trên tập hợp **2 đến 3 tác vụ**.

Yêu cầu

- Thiết kế thuật toán chạy tốt với số lượng lớn tác vụ.
- Thuật toán có ít tham số.

Hướng giải quyết cụ thể

- Thiết kế **cấu trúc mới** cho tiến hóa đa nhiệm phù hợp với tối ưu nhiều tác vụ.
- Áp dụng mô hình **Multi-Armed Bandits**, học trên dữ liệu hàm mục tiêu để ghép cặp các tác vụ

Mục lục

- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - Mô hình Multi-Armed Bandits
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

Mục lục

- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - Mô hình Multi-Armed Bandits
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

Thuật toán tiến hóa đa nhiệm - 1

Phát biểu bài toán tối ưu đa nhiệm tổng quát

- **Cho:** K bài toán tối ưu.
- Tác vụ T_k ứng với việc giải bài toán thứ k .
- T_k có không gian tìm kiếm \mathcal{X}_k , hàm mục tiêu $f_k : \mathcal{X}_k \rightarrow \mathbb{R}$
- **Yêu cầu:** tìm $\{x_1^*, x_2^*, \dots, x_{K-1}^*, x_K^*\} = \operatorname{argmin}\{f_1(x), f_2(x), \dots, f_{K-1}(x), f_K(x)\}$ với x_k^* là nghiệm tối ưu toàn cục của T_k .

Tính chất đặt biệt của tiến hóa đa nhiệm

Việc giải tác vụ T_k có thể có ảnh hưởng tốt giúp giải $T_{k'}, k' \neq k$ tối ưu hơn.

Thuật toán tiền hóa đa nhiệm - 2

Không gian biểu diễn chung

- **Cho:** K tác vụ với chiều lần lượt là $\{D_1, D_2, \dots, D_K\}$.
- **Biểu diễn chung:** $D_{unified} = \max\{D_1, D_2, \dots, D_K\}$.

Không gian biểu diễn chung trong tối ưu số thực

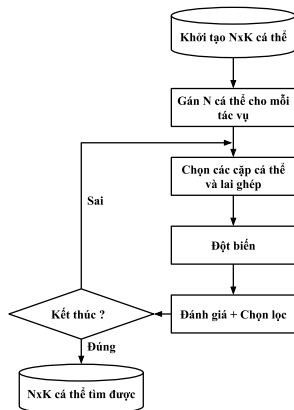
- Quy ước là $[0, 1]^{D_{unified}}$.

Cách sử dụng không gian biểu diễn chung

- Thực hiện toán tử tiền hóa trên cá thể với chiều $D_{unified}$.
- Đánh giá trên cá thể với chiều $\{D_1, D_2, \dots, D_K\}$.

Thuật toán tiến hóa đa nhiệm - 3

MFEA - Multifactorial Evolutionary Algorithm



Hình 3: Khung thuật toán MFEA

Skill factor

Skill factor τ_i của cá thể i^{th} là index của tác vụ trong K tác vụ, mà cá thể i thuộc về.

Algorithm 1 Lai ghép trong MFEA

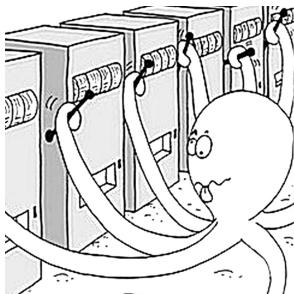
- 1: Lấy ngẫu nhiên hai cá thể cha mẹ p_a và p_b từ P
- 2: if $\tau_a == \tau_b$ then
- 3: $[c_a, c_b] \leftarrow$ Lai ghép cùng tác vụ giữa p_a và p_b
- 4: Gán skill factor τ_a cho c_a và c_b
- 5: else if $rand \leq rmp$ then
- 6: $[c_a, c_b] \leftarrow$ Lai ghép khác tác vụ giữa p_a và p_b
- 7: Gán ngẫu nhiên skill factor τ_a or τ_b cho từng con sinh ra
- 8: end if

Mục lục

- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - **Mô hình Multi-Armed Bandits**
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

Mô hình Multi-Armed Bandits - 1

Nguồn gốc tên gọi Multi-Armed Bandits



Hình 4: Lựa chọn chơi máy nào, thứ tự thế nào, mỗi máy bao lần, thắng càng nhiều tiền càng tốt

Algorithm 2 Mô hình Multi-armed bandits

- 1: **Cho:** K lựa chọn, T vòng.
 - 2: **for** vòng thứ $t \in \{1, \dots, T\}$ **do**
 - 3: Chọn lựa chọn a_t ;
 - 4: Nhận về phần thưởng $r_t \in [0, 1]$ cho lựa chọn a_t ;
 - 5: **end for**
-

Mô hình Multi-Armed Bandits - 2

Ứng dụng của MAB



Hình 5: Lựa chọn nguồn tin và đề xuất tin tức sao cho người dùng xem nhiều nhất



Hình 6: Lựa chọn kênh đầu tư sao cho lãi nhất

→ Trong tiến hóa đa nhiệm, cũng có thể áp dụng MAB để chọn tác vụ ghép cặp sao cho kết quả tối ưu tối nhất.

Mục lục

- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - Mô hình Multi-Armed Bandits
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

MFEA - giải số lượng tác vụ nhỏ

MFEA

- Abhishek Gupta, Yew-Soon Ong, and Liang Feng. “Multifactorial evolution: toward evolutionary multitasking”. In: *IEEE Transactions on Evolutionary Computation* 20.3 (2015), pp. 343–357
- **Vấn đề:** Lai ghép khác tác vụ một cách ngẫu nhiên.

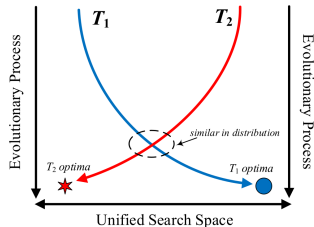
MFEA-II

- Kavitesh Kumar Bali, Yew-Soon Ong, Abhishek Gupta, and Puay Siew Tan. “Multifactorial evolutionary algorithm with online transfer parameter estimation: MFEA-II”. In: *IEEE Transactions on Evolutionary Computation* 24.1 (2019), pp. 69–83
- **Ý tưởng:** Giải quyết bài toán con (tối ưu hàm lồi), tối ưu $RMP \in \mathbb{R}^{K \times K}$ sao cho con sinh ra từ quần thể cha mẹ và RMP giống với con được chọn lọc sang thế hệ tiếp theo nhất.
- **Vấn đề:** Thuật toán chạy chậm, khi phải giải quyết K^2 bài toán con.

MFEA - giải số lượng tác vụ lớn

GMFEA

- Jing Tang, Yingke Chen, Zixuan Deng, Yanping Xiang, and Colin Paul Joy. “A Group-based Approach to Improve Multifactorial Evolutionary Algorithm.”. In: *IJCAI*. 2018, pp. 3870–3876
- Với mỗi tác vụ, chọn một vài cá thể tốt nhất làm đại diện, gộp lại thành một tập dữ liệu
- Dùng thuật toán K-Means, nhóm dữ liệu trên lại.
- Các tác vụ ở cùng một cụm mới được trao đổi thông tin cho nhau.
- **Vấn đề:** trường hợp quần thể cùng phân phối nhưng cần đi về khác hướng (Hình 7).



Hình 7: Hai tác vụ cùng vị trí quần thể, khác vị trí cực trị.

MFEA - giải số lượng tác vụ lớn

SBSGA

- Rung-Tzuo Liaw and Chuan-Kang Ting. “Evolutionary manytasking optimization based on symbiosis in biocoenosis”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. 01. 2019, pp. 4295–4303
- Trao đổi thông tin giữa các tác vụ bằng việc trao cá thể (swap) thay vì lai ghép.
- Lưu lại số lần trao đổi thành công T^{pos} (con được chọn lọc vào quần thể mới) và thất bại T^{neg} .
- Xác suất trao đổi tính bằng công thức:

$$RMP_{i,j} = \frac{T^{pos}}{T^{pos} + T^{neg}} \quad (1)$$

- **Vấn đề:** Phương pháp ghép cặp tự thiết kế không dựa trên lý thuyết.

MFEA - giải số lượng tác vụ lớn

MaTGA

- Yongliang Chen, Jinghui Zhong, Liang Feng, and Jun Zhang. “An adaptive archive-based evolutionary framework for many-task optimization”. In: *IEEE Transactions on Emerging Topics in Computational Intelligence* 4.3 (2019), pp. 369–384
- Lưu một phần quần thể của từng tác vụ qua nhiều thế hệ.
- Tính khoảng cách KL Divergence giữa các tập quần thể đã lưu.
- **Vấn đề:** Có nhiều tham số, thời gian tính toán chậm.

Mục lục

- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - Mô hình Multi-Armed Bandits
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

Cơ chế chọn tác vụ hỗ trợ - MAB

Mô hình hóa việc chọn tác vụ để lai ghép bằng MAB

Định nghĩa (Lựa chọn)

Với mỗi tác vụ T_k , sẽ có $K - 1$ lựa chọn, tương ứng với $K - 1$ tác vụ $T_{k'}$ mà $k' \in \{1, \dots, K\}$ và $k' \neq k$.

Định nghĩa (Phần thưởng)

Sau khi tác vụ $T_{k'}$ được lựa chọn để ghép cặp với tác vụ T_k , phần thưởng của việc chọn tác vụ $T_{k'}$ được định nghĩa như sau:

$$r(k, k') = \begin{cases} 1 & \text{nếu } f_k(c) < f_k(p), \exists p \in P^k \\ 0 & \text{trong các trường hợp khác.} \end{cases} \quad (2)$$

- c là con sinh ra trong quá trình lai ghép khác tác vụ
- $f_k(\cdot)$ là hàm đánh giá của tác vụ T_k

Cách giải bài toán con chọn tác vụ hỗ trợ - KLUCB

Giả định

- **Phần thưởng:** Biến ngẫu nhiên với giá trị $\{0, 1\}$
- **Giả định:** Phần thưởng sinh từ phân phối Bernoulli chưa biết trước.

Cách giải - KLUCB

$$k' = \operatorname{argmax}_j \mu(j) + \frac{1 + t \times \log^2(t)}{N(j)} \quad (3)$$

- $\mu(j)$ là giá trị trung bình ước lượng được của phần thưởng khi lựa chọn j
- $N(j)$ là tổng số lần thuật toán đã lựa chọn j
- t là tổng số của tất cả các lần lựa chọn

Tham khảo lý thuyết tại

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020

Khung giải thuật đề xuất - Ma2BEA

Algorithm 3 Many-task Multi-armed Bandit Evolutionary Algorithm (Ma²BEA) trong mỗi thế hệ của tác vụ T_k

- 1: Khởi tạo quần thể con $P_{(c)}^k = \emptyset$
 - 2: **while** số con sinh ra $< N$ **do**
 - 3: Chọn ngẫu nhiên cá thể cha p_a từ P^k
 - 4: **if** $\text{rand}(0, 1) < \text{rmp}$ **then**
 - 5: Chọn tác vụ $T_{k'}$ sử dụng phương trình Kullback–Leibler UCB (KL-UCB) trong Công thức 3
 - 6: Chọn ngẫu nhiên cá thể mẹ p_b từ $P^{k'}$
 - 7: $c = \text{Lai ghép khác tác vụ giữa } p_a \text{ và } p_b$
 - 8: **else**
 - 9: Chọn ngẫu nhiên cá thể mẹ p_b từ P^k
 - 10: $c = \text{Lai ghép cùng tác vụ giữa } p_a \text{ và } p_b$
 - 11: **end if**
 - 12: $c = \text{Đột biến } c$
 - 13: Đánh giá cá thể con c
 - 14: Cập nhật lại ước lượng $\mu(k')$ và số lượng $N(k')$ cho KL-UCB nếu c được sinh ra từ việc *Lai ghép khác tác vụ*
 - 15: $P_{(c)}^k = P_{(c)}^k \cup \{c\}$
 - 16: **end while**
 - 17: $P^k \leftarrow \text{Chọn } N \text{ cá thể tốt nhất từ } P^k \cup P_{(c)}^k \text{ để tạo lại } P^k \text{ cho thế hệ tiếp theo}$
-

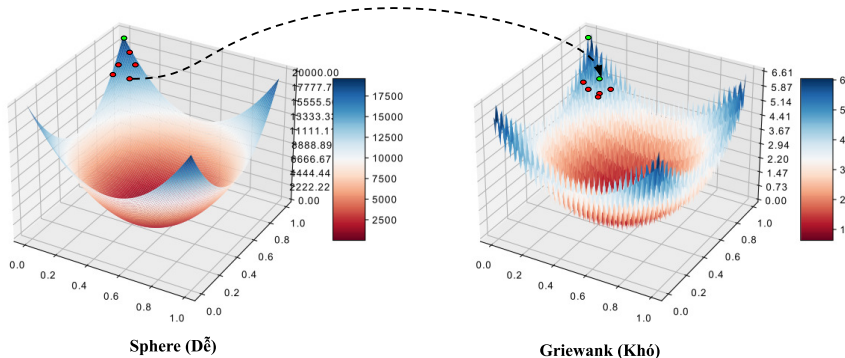
Cấu trúc cập nhật tuần tự - 1

Algorithm 4 Giải mã của toàn bộ Ma^2BEA

```
1: for  $k \in \{1, \dots, K\}$  do
2:   Khởi tạo ngẫu nhiên  $N \sim \mathbb{R}^{D_{\text{unified}}}$  cá thể để tạo ra quần thể  $P^k$ 
3:   Đánh giá  $P^k$  trên  $f_k$ 
4: end for
5: while điều kiện kết thúc chưa thỏa mãn do
6:   for  $k \in$  hoán vị ngẫu nhiên của  $(1, \dots, K)$  do
7:     Thực hiện Thuật toán 3 cho tác vụ  $T_k$ 
8:   end for
9: end while
```

Cấu trúc cập nhật tuần tự - 2

Ví dụ cho tác dụng của cập nhật tuần tự trong Ma^2BEA



Hình 8: Cập nhật tuần tự trên hai tác vụ có địa hình hàm mục tiêu tương đồng. Trục tung, trục hoành là giá trị của nghiệm. Trục thẳng đứng là giá trị hàm mục tiêu.

Tóm tắt đóng góp

Các đề xuất tại Ma²BEA

- Mô hình hóa ghép cặp trao đổi tri thức giữa các tác vụ trên mô hình MAB, giải bằng KLUCB.
- Thay cấu trúc của MFEA bằng cấu trúc trao đổi tuần tự.

So với MFEA

- SBSGA, MaTGA, Ma²BEA đều trội hơn ở cơ chế trao đổi tri thức tự thích ứng.

So với MaTGA

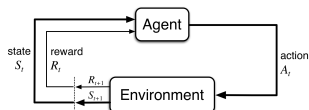
- Ma²BEA nhanh hơn, vì chỉ phân tích dữ liệu hàm mục tiêu ($\sim \mathbb{R}^{N \times K}$), thay vì phân tích dữ liệu lịch sử quần thể ($\sim \mathbb{R}^{D_{unified} \times N \times K}$).
- Không phụ thuộc biểu diễn quần thể.
- Ma²BEA ít tham số hơn, không cần chọn lưu lại quá khứ bao nhiêu cá thể như MaTGA.

So với SBSGA

- Cách ghép cặp của Ma²BEA có nền tảng lý thuyết hơn so với SBSGA

Áp dụng - Tối ưu nhiều mạng nơ-ron

Bài toán học tăng cường



Hình 9: Tương tác giữa tác tử (Agent) và môi trường (Environment) trong mô hình Markov Decision Process.

Lý do áp dụng tiến hóa đa nhiệm cho học tăng cường

- Hàm mục tiêu không có gradient.
- Có nhiều môi trường tương tự nhau, việc giải cùng nhau hy vọng sẽ cho kết quả tối ưu hơn.

Giải nhiều bài toán học tăng cường

Cho: K môi trường $\{E_1, \dots, E_K\}$

$$\underset{\theta_k}{\text{maximize}} F(\theta_k; E_k) = \sum_{t=1}^T r_t \quad (4)$$

- θ_k : tham số mô hình đưa quyết định (mạng nơ-ron)
- r_t : phần thưởng tại thời điểm t

Mục lục

- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - Mô hình Multi-Armed Bandits
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

Mục lục

- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - Mô hình Multi-Armed Bandits
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

Dữ liệu thử nghiệm và cài đặt các thuật toán

Mô tả bộ dữ liệu MaTO-10

- Bộ MaTO-10^a gồm 10 tác vụ tương ứng với 10 hàm số thực 25 hoặc 50 biến.
- T_1, \dots, T_4 dễ, T_6, \dots, T_{10} khó.
- Biết trước T_1 hỗ trợ T_5 , T_2 hỗ trợ T_6 , T_3, T_4 hỗ trợ T_7 , T_4 hỗ trợ T_9 .
- Phục vụ phân tích mô đun ghép cặp các tác vụ hoạt động đúng hay không.

^aYongliang Chen, Jinghui Zhong, Liang Feng, and Jun Zhang. “An adaptive archive-based evolutionary framework for many-task optimization”. In: *IEEE Transactions on Emerging Topics in Computational Intelligence* 4.3 (2019), pp. 369–384.

Tham số của MFEA/EBSGA/Ma²BEA

- Kích thước quần thể: 100
- rmp: 0.3
- sbxdi: 2
- pmdi: 5

Tham số của MaTGA

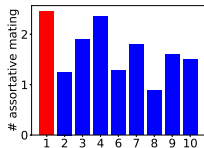
Dùng tham số mặc định của tác giả ^a

Kết quả tối ưu

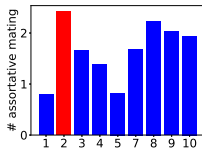
Task	Ma ² BEA	MFEA	MaTGA	EBSGA
T_1	3.72E-05	1.31E+00 (−)	2.45E-04 (−)	5.58E-04 (−)
T_2	6.48E-06	1.27E+00 (−)	4.75E-04 (−)	6.10E-05 (−)
T_3	4.35E-07	1.17E+00 (−)	0.00E+00 (\approx)	2.00E-06 (−)
T_4	7.09E-13	3.37E+00 (−)	1.00E-06 (−)	1.60E-05 (−)
T_5	1.74E+01	8.15E+02 (−)	2.16E-02 (+)	3.29E-02 (+)
T_6	7.03E-04	1.99E+01 (−)	3.61E-03 (−)	8.59E-04 (\approx)
T_7	1.00E-02	1.05E+01 (−)	5.57E-04 (+)	1.60E-03 (+)
T_8	1.89E+02	2.34E+03 (−)	3.40E-02 (−)	4.73E-02 (−)
T_9	4.00E-07	4.11E+02 (−)	4.52E-03 (−)	4.58E-03 (−)
T_{10}	2.74E+01	1.54E+01 (+)	1.57E+01 (+)	3.41E+01 (−)

Bảng 1: Giá trị trung bình hàm mục tiêu của các thuật toán tiến hóa đa nhiệm sau 30 lần chạy độc lập. (−, +, and \approx thể hiện thuật toán kém đáng kể, tốt đáng kể hoặc tương đương với Ma²BEA, kiểm định bằng Wilcoxon signed-rank test $\alpha = 0.05$)

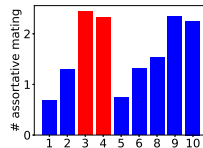
Kết quả ghép cặp



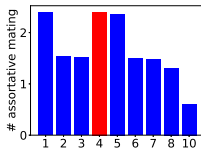
(a) Tác vụ 5



(b) Tác vụ 6



(c) Tác vụ 7



(d) Tác vụ 9

Hình 10: Số lần trung bình mỗi thể hệ mỗi tác vụ chọn ghép cặp với tác vụ khác trong Ma^2BEA . Tác vụ hỗ trợ thật được hiển thị bằng màu đỏ.

Mục lục

- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - Mô hình Multi-Armed Bandits
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

Dữ liệu thử nghiệm

Mô tả bộ dữ liệu MaTO-50

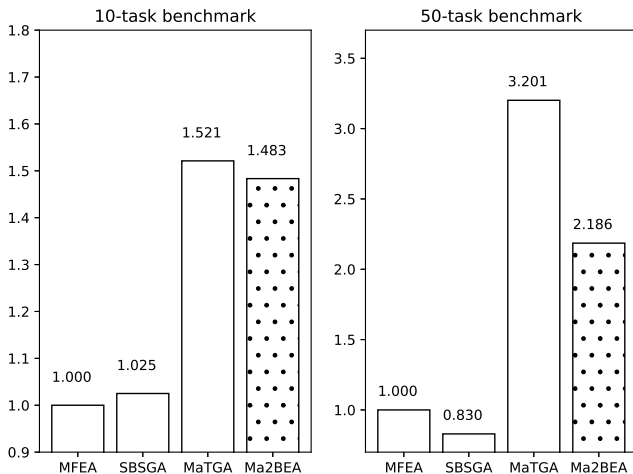
- 10 bộ, mỗi bộ 50 tác vụ tương ứng với 50 hàm số thực 50 biến.
- Không biết trước mối quan hệ giữa các hàm.
- Đa dạng hơn MaTO-10.
- Cuộc thi của hội thảo WCCI/GECCO 2020.

Kết quả tối ưu

Benchmark	Ma ² BEA	MFEA	MaTGA	EBSGA
B_1	50 (50)	0 (0)	0 (0)	0 (0)
B_2	50 (50)	0 (0)	0 (0)	0 (0)
B_3	8 (0)	0 (0)	42 (8)	0 (0)
B_4	17 (9)	0 (0)	0 (0)	33 (33)
B_5	50 (50)	0 (0)	0 (0)	0 (0)
B_6	50 (50)	0 (0)	0 (0)	0 (0)
B_7	3 (0)	0 (0)	14 (2)	33 (28)
B_8	30 (28)	0 (0)	0 (0)	20 (20)
B_9	42 (42)	0 (0)	0 (0)	8 (8)
B_{10}	8 (6)	2 (0)	11 (10)	29 (23)

Bảng 2: Bảng thể hiện số lần một thuật toán giải tốt nhất và tốt nhất đáng kể trên 10 tập hàm đánh giá, mỗi tập có 50 hàm đánh giá của bộ dữ liệu MaTO-50, thống kê lại sau 30 lần chạy độc lập.

So sánh thời gian chạy



Hình 11: Thời gian chạy của các thuật toán tiến hóa đa nhiệm, lấy thời gian chạy của Multifactorial Evolutionary Algorithm (MFEA) trên cũng một một trường thực nghiệm, cùng ngôn ngữ lập trình làm chuẩn

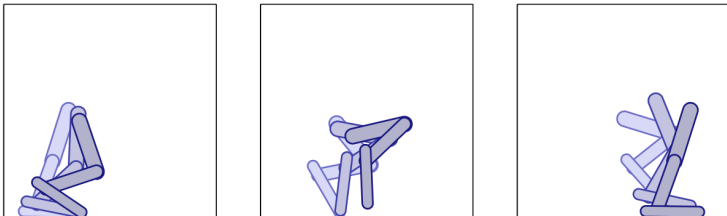
Mục lục

- 1 Giới thiệu
- 2 Cơ sở lý thuyết
 - Thuật toán tiến hóa đa nhiệm
 - Mô hình Multi-Armed Bandits
- 3 Các nghiên cứu liên quan
- 4 Thuật toán đề xuất
- 5 Kết quả thực nghiệm
 - 10 tác vụ
 - 50 tác vụ
 - Mujoco Multitask

Dữ liệu thử nghiệm

Dữ liệu thử nghiệm

- Sử dụng bộ Mujoco Multitask^a.
- Thực nghiệm trên hai tập dữ liệu *Hopper-gravity* và *Hopper-size*



Hình 12: Minh họa cách hoạt động của robot Hopper [7].

^aPeter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. “Deep reinforcement learning that matters”. In: *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

Cài đặt thực nghiệm

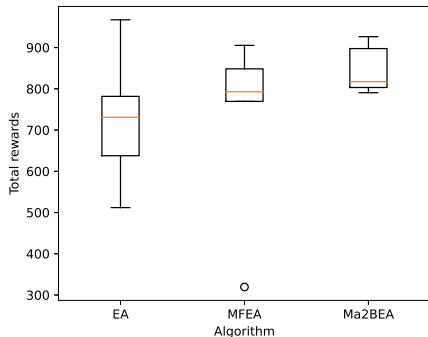
Môi trường

- **Trạng thái:** $s \in \mathbb{R}^{11}$
- **Hành động:** $a \in \mathbb{R}^3$

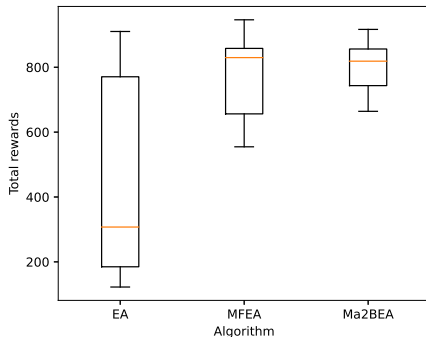
Mô hình ra quyết định cần tối ưu

- **Đầu vào:** là 11 chiều
- **Lớp ẩn:** chứa 16 nút
- **Activation:** tanh
- **Đầu ra:** là 3 chiều
- **Không gian biểu diễn chung:** $[0, 1]^{243}$.
- **Giải mã:** dẫn từ khoảng $[0, 1]$ đến $[-5, 5]$.
- $(T, N) = (100, 10)$

Kết quả tối ưu



(a) Hopper – gravity



(b) Hopper – size

Hình 13: Biểu đồ hộp (minh họa các giá trị nhỏ nhất, quartile thứ nhất, median, quartile thứ ba, giá trị lớn nhất) của các hàm mục tiêu của bộ Mujoco Multitask, giải bởi Evolutionary Algorithm (EA), MFEA, và Ma²BEA. Giá trị được thống kê ở biểu đồ hộp là giá trị hàm mục tiêu tại thể hệ cuối cùng của các tác vụ khác nhau, giải bởi từng thuật toán trên từng bộ dữ liệu.

Bài báo trong quá trình học

Nghiên cứu chính của luận văn

- Le Tien Thanh, La Van Cuong, Ta Bao Thang, and Huynh Thi Thanh Binh. "Multi-Armed Bandits for Many-task Evolutionary Optimization". In: *2021 IEEE Congress on Evolutionary Computation (CEC)*. IEEE. 2021, pp. 1–8

Thực tập nghiên cứu - NII

- Le Van An, Le Tien Thanh, Nguyen Phi Le, Huynh Thi Thanh Binh, Akerkar Rajendra, and Yusheng Ji. "GCRINT: Network Traffic Imputation Using Graph Convolutional Recurrent Neural Network". In: *2021 IEEE International Conference on Communications*. IEEE. 2021, pp. 1–8
- Le Van An, Le Tien Thanh, Nguyen Phi Le, Huynh Thi Thanh Binh, and Yusheng Ji. "Multi-time-step Segment Routing based Traffic Engineering Leveraging Traffic Prediction". In: *2021 IFIP/IEEE International Symposium on Integrated Network Management*. IEEE. 2021, pp. 1–8

Cảm ơn thầy cô đã lắng nghe!