Các biến thể

### Vấn đề về cái túi đa mục tiêu

### Tóm tắt: Bài toán knapsack (KP) và phiên bản đa chiều của nó (MKP) là những bài toán cơ bản trong tối ưu hóa tổ hợp. Trong bài báo này, chúng tôi xem xét phần mở rộng đa mục tiêu của chúng (MOKP và MOMKP), với mục đích là thu được hoặc ước tính tập hợp các giải pháp hiệu quả. Trong bước đầu tiên, chúng tôi phân loại và mô tả ngắn gọn các tác phẩm hiện có, về cơ bản dựa trên việc sử dụng siêu mô phỏng. Trong bước thứ hai, chúng tôi đề xuất sự điều chỉnh của tìm kiếm cục bộ Pareto hai giai đoạn (2PPLS) với độ phân giải của MOMKP. Với mục đích này, chúng tôi sử dụng một vùng lân cận quy mô rất lớn (VLSN) trong giai đoạn thứ hai của phương pháp, đó là tìm kiếm cục bộ Pareto. Chúng tôi so sánh kết quả của mình với kết quả hiện đại nhất và chúng tôi cho thấy rằng chúng tôi thu được kết quả chưa từng đạt được trước đây bằng phương pháp phỏng đoán, đối với các trường hợp khách quan. Cuối cùng, chúng tôi xem xét phần mở rộng cho các trường hợp ba mục tiêu.

<https://arxiv.org/ftp/arxiv/papers/1007/1007.4063.pdf?fbclid=IwAR22ua9kDOdnPkl2Ifoz0vpeV9zqAiJCgJvxD4VwLkQpfn0Mm3Cfd63G-pc>

1. **Vấn đề về cái túi đa chiều**

Tóm tắt: Bài toán gói dữ liệu đa chiều (MDKP) là một bài toán gói hàng có nhiều ràng buộc tài nguyên. Cả phiên bản chung và phiên bản 0-1 của bài toán này đều có nhiều ứng dụng thực tế. MDKP được biết là NP-cứng mạnh. Trong bài báo này, chúng tôi đề xuất một phương pháp heuristic mới giống như tham lam, chủ yếu dành cho MDKP chung, nhưng bản thân nó cũng hiệu quả cho MDKP 0-1. Phương pháp heuristic của chúng tôi khác với phương pháp heuristics giống như tham lam hiện có ở hai khía cạnh. Đầu tiên, phương pháp phỏng đoán hiện tại dựa vào mức tiêu thụ tổng hợp tài nguyên của mỗi mặt hàng để đưa ra quyết định lựa chọn mặt hàng, trong khi phương pháp phỏng đoán của chúng tôi sử dụng khả năng hiệu quả, được định nghĩa là số lượng bản sao tối đa của một mặt hàng có thể được chấp nhận nếu toàn bộ túi đựng được sử dụng cho việc đó một mình mặt hàng, làm tiêu chí để đưa ra quyết định lựa chọn mặt hàng. Thứ hai, các phương pháp khác chỉ tăng giá trị của mỗi biến quyết định lên một đơn vị, trong khi phương pháp heuristic của chúng tôi thêm các biến quyết định vào giải pháp theo lô và do đó cải thiện đáng kể hiệu quả tính toán cho các bài toán quy mô lớn. Chúng tôi thực hiện các nghiên cứu số chuyên sâu về các vấn đề thử nghiệm được tạo ngẫu nhiên với nhiều cài đặt tham số và các vấn đề điểm chuẩn từ các tài liệu liên quan. Chúng tôi chứng minh rằng phương pháp heuristic mới cải thiện đáng kể hiệu quả tính toán của các phương pháp hiện có và tạo ra các giải pháp mạnh mẽ và gần như tối ưu. Phương pháp heuristic mới tỏ ra đặc biệt hiệu quả đối với các bài toán có ba lô chiều cao với số lượng biến quyết định từ nhỏ đến trung bình, thường được coi là MDKP “khó” và không có phương pháp heuristic hiệu quả về mặt tính toán nào để xử lý các vấn đề như vậy.

[**https://www.math.wsu.edu/math/faculty/lih/MDKP.pdf?fbclid=IwAR2kHl06an2IyvLAvwcOAs8inDgg9\_Kk6J5sauJQBQQLCgbS0\_HOVRo8LHw**](https://www.math.wsu.edu/math/faculty/lih/MDKP.pdf?fbclid=IwAR2kHl06an2IyvLAvwcOAs8inDgg9_Kk6J5sauJQBQQLCgbS0_HOVRo8LHw)

### Nhiều vấn đề về ba lô

Bài toán Multiple Knapsack (MKP) là bài toán gán một tập hợp con gồm n mục cho m túi đựng riêng biệt, sao cho tổng lợi nhuận của các mặt hàng đã chọn là tối đa, mà không vượt quá dung lượng của mỗi túi. Bài toán có một số ứng dụng trong hải quân cũng như quản lý tài chính. Một thuật toán chính xác mới cho MKP được trình bày, được thiết kế đặc biệt để giải quyết các trường hợp vấn đề lớn. Thuật toán nhánh và giới hạn đệ quy áp dụng thư giãn thay thế để lấy giới hạn trên, trong khi giới hạn dưới thu được bằng cách tách nghiệm thay thế thành m knapsacks bằng cách giải một loạt Bài toán về tổng phụ. Một thuật toán lập trình động có thể phân tách mới được trình bày cho giải pháp của Bài toán tổng hợp con và chúng tôi cũng sử dụng thuật toán này để thắt chặt các giới hạn dung lượng nhằm đạt được giới hạn trên tốt hơn. Thuật toán đã phát triển được so sánh với thuật toán mtm của Martello và Toth, cho thấy những lợi ích của cách tiếp cận mới. Một kết quả đáng ngạc nhiên là các trường hợp lớn với n = 100 000 mục có thể được giải quyết trong vòng chưa đầy một giây và thuật toán có hiệu suất ổn định ngay cả đối với các trường hợp có hệ số trong phạm vi vừa phải lớn.

[**https://www.sciasedirect.com/science/article/abs/pii/S0377221798001209?fbclid=IwAR3JqvEb\_o9-\_g9ZAEpANFwl6z52jASNaZFLAV88rrWSdMA82v\_WdcrgxBQ#:~:text nhiều%20**](https://www.sciencedirect.com/science/article/abs/pii/S0377221798001209?fbclid=IwAR3JqvEb_o9-_g9ZAEpANFwl6z52jASNaZFLAV88rrWSdMA82v_WdcrgxBQ#:~:text=The%20Multiple%20Knapsack%20Problem%20)

### Bài toán ba lô bậc hai

### Bài toán ba lô bậc hai (QKP) , được giới thiệu lần đầu tiên vào thế kỷ 19, [[1]](https://en.wikipedia.org/wiki/Quadratic_knapsack_problem?fbclid=IwAR2U3UCFuy1J8LJb_LCk6TLXzhqVwTKRJB7xgntaWqyH6U4ryx9911MZZJ8#cite_note-1) là một phần mở rộng của [bài toán ba lô](https://en.wikipedia.org/wiki/Knapsack_problem) cho phép các số hạng bậc hai trong hàm mục tiêu: Cho một tập hợp các mục, mỗi mục có trọng số, giá trị và phần dư lợi nhuận có thể kiếm được nếu hai mục được chọn, xác định số lượng các mục để bao gồm trong một bộ sưu tập mà không vượt quá sức chứa của [ba lô](https://en.wikipedia.org/wiki/Knapsack) , để tối đa hóa lợi nhuận tổng thể. Thông thường, các bài toán về bao gói bậc hai đi kèm với hạn chế về số lượng bản sao của mỗi loại vật phẩm: 0 hoặc 1. Loại QKP đặc biệt này tạo thành bài toán bao gói [bậc hai 0-1](https://en.wikipedia.org/wiki/0-1_quadratic_knapsack_problem) , được Gallo et al lần đầu tiên thảo luận. [[2]](https://en.wikipedia.org/wiki/Quadratic_knapsack_problem?fbclid=IwAR2U3UCFuy1J8LJb_LCk6TLXzhqVwTKRJB7xgntaWqyH6U4ryx9911MZZJ8#cite_note-2) Bài toán ba lô bậc hai 0-1 là một biến thể của các bài toán ba lô, kết hợp các tính năng của bài toán ba lô không giới hạn, bài toán ba lô 0-1 và bài toán ba lô bậc hai.

### <https://en.wikipedia.org/wiki/Quadratic_knapsack_problem?fbclid=IwAR2U3UCFuy1J8LJb_LCk6TLXzhqVwTKRJB7xgntaWqyH6U4ryx9911MZZJ8>

### Bài toán tổng tập hợp con

Bài **toán tổng tập hợp con** (SSP) là một [bài toán quyết định](https://en.wikipedia.org/wiki/Decision_problem) trong [khoa học máy tính](https://en.wikipedia.org/wiki/Computer_science) . Trong công thức tổng quát nhất của nó, có một tập [hợp](https://en.wikipedia.org/wiki/Multiset) {\ displaystyle S} của các số nguyên và một target-sum {\ displaystyle T} , và câu hỏi đặt ra là quyết định xem có tập hợp con nào của các số nguyên tổng chính xác bằng {\ displaystyle T} hay không *.* [[1]](https://en.wikipedia.org/wiki/Subset_sum_problem?fbclid=IwAR0UkiNO6XymsBXbp6YNn_4QSczM3-wPIMvWUN8VpBuVobgeS_VKuraeUDQ#cite_note-kleinberg2006p491-1) Vấn đề được biết là NP. Hơn nữa, một số biến thể bị hạn chế của nó cũng là [NP-hoàn chỉnh](https://en.wikipedia.org/wiki/NP-completeness) , ví dụ: [[1]](https://en.wikipedia.org/wiki/Subset_sum_problem?fbclid=IwAR0UkiNO6XymsBXbp6YNn_4QSczM3-wPIMvWUN8VpBuVobgeS_VKuraeUDQ#cite_note-kleinberg2006p491-1)

* Biến thể trong đó tất cả các đầu vào đều dương.
* Biến thể trong đó đầu vào có thể dương hoặc âm và {\ displaystyle T = 0} . Ví dụ: với tập hợp {\ displaystyle \ {- 7, -3, -2,9000,5,8 \}} , câu trả lời là *có* vì tập con {\ displaystyle \ {- 3, -2,5 \} } tổng bằng 0.
* Biến thể trong đó tất cả các đầu vào đều dương và tổng mục tiêu chính xác bằng một nửa tổng của tất cả các đầu vào, tức là {\ displaystyle T = {\ frac {1} {2}} (a\_ {1} + \ dot + a\_ { n})} . Trường hợp đặc biệt này của SSP được gọi là sự [cố phân vùng](https://en.wikipedia.org/wiki/Partition_problem) .

SSP cũng có thể được coi là một [bài toán tối ưu hóa](https://en.wikipedia.org/wiki/Optimization_problem) : tìm một tập hợp con có tổng của nó nhiều nhất là *T* , và tùy thuộc vào đó, càng gần *T càng tốt* . Nó là [NP-khó](https://en.wikipedia.org/wiki/NP-hard) , nhưng có một số thuật toán có thể giải quyết nó một cách hợp lý nhanh chóng trong thực tế.

SSP là một trường hợp đặc biệt của [bài toán knapsack](https://en.wikipedia.org/wiki/Knapsack_problem) và bài toán [tổng nhiều tập hợp con](https://en.wikipedia.org/wiki/Multiple_subset_sum) .

[**https://en.wikipedia.org/wiki/Subset\_sum\_problem?fbclid=IwAR0UkiNO6XymsBXbp6YNn\_4QSczM3-wPIMvWUN8VpBuVobgeS\_VKuraeUDQ**](https://en.wikipedia.org/wiki/Subset_sum_problem?fbclid=IwAR0UkiNO6XymsBXbp6YNn_4QSczM3-wPIMvWUN8VpBuVobgeS_VKuraeUDQ)

1. **Vấn đề về cái túi hình học**

Mục tiêu của bài báo này là trình bày một nghiên cứu thực nghiệm về Bài toán cái túi hình học (GKP) với mục tiêu thu được các giải pháp tối ưu có thể chứng minh được. Chúng tôi giới thiệu mô hình Lập trình tuyến tính số nguyên cho GKP và áp dụng nó cho hàng trăm trường hợp của hai lớp: một bao gồm các điểm được tạo đồng nhất với các giá trị được gán ngẫu nhiên; và một cấu trúc khác bao gồm các điểm phân lớp lồi với phân bố giá trị thiên về việc tập trung các điểm có giá trị âm vào các lớp trong cùng. Các bài kiểm tra thử nghiệm được sử dụng để hướng dẫn lựa chọn các tham số đầu vào để tránh tạo ra các trường hợp nhỏ. Các thử nghiệm của chúng tôi cho thấy rằng lớp phân lớp khó được giải quyết đến mức tối ưu hơn đáng kể, trong thực tế, vì ngay cả những trường hợp có ít nhất 35 điểm cũng không thể được giải quyết trong vòng 5 phút trong thời gian CPU.

[**http://www.cs.umanitoba.ca/~cccg2018/papers/session5B-p2.pdf**](http://www.cs.umanitoba.ca/~cccg2018/papers/session5B-p2.pdf)