

2D BIN PACKING PROBLEM AND APPLICATION IN MARITIME TRANSPORTATION

Phung The Huan*, Hoang Thi Canh, Vu Duc Thai, Bui Ngoc Tuan

TNU - University of Information and Communication Technology

ARTICLE INFO	ABSTRACT
Received: 08/3/2021 Revised: 31/5/2021 Published: 31/5/2021	Along with the strong development of the economy and the current commerce, the transport industry is also making great strides in both quantity and quality. One of the most important factors influencing the productivity of the transport process is the process of packing and dispatching to the transport. In this article we present the Bin Packing 2D problem and the areas of the problem application. Bin Packing is a problem that items of different volumes must be packed into a finite number of bins. To answer this question, this article has studied some of the methods used through research and synthesis based on articles of reputable publishers to filter out original articles and the most influential articles to survey and analyze. The article gives the advantages and disadvantages of each method and suggests future development directions.
KEYWORDS Transport Planning Optimization Packing Sorting	

VỀ BÀI TOÁN BIN PACKING 2D VÀ ỨNG DỤNG TRONG VẬN TẢI HÀNG HẢI

Phùng Thế Huân*, Hoàng Thị Cảnh, Vũ Đức Thái, Bùi Ngọc Tuấn

Trường Đại học Công nghệ Thông tin và Truyền thông – ĐH Thái Nguyên

THÔNG TIN BÀI BÁO	TÓM TẮT
Ngày nhận bài: 08/3/2021 Ngày hoàn thiện: 31/5/2021 Ngày đăng: 31/5/2021	Cùng với sự phát triển mạnh mẽ của nền kinh tế và giao thương hàng hóa hiện nay, ngành vận tải cũng đang có những bước tiến vượt bậc về cả số lượng và chất lượng. Một trong những yếu tố quan trọng ảnh hưởng đến năng suất của quá trình vận tải chính là quá trình đóng gói và điều phối hàng hóa vào các phương tiện vận chuyển một cách phù hợp. Trong bài báo này chúng tôi trình bày về bài toán Bin Packing 2D và các lĩnh vực ứng dụng bài toán. Bin Packing là dạng bài toán đóng thùng sao cho tất cả các đồ vật có thể tích khác nhau được đóng gói vào số lượng thùng sử dụng là ít nhất. Để trả lời cho câu hỏi này, bài báo này đã nghiên cứu về một số phương pháp đã từng sử dụng thông qua việc khảo cứu và tổng hợp dựa trên các bài báo của các nhà xuất bản uy tín để lọc ra các bài báo gốc và các bài có ảnh hưởng nhất để khảo sát và phân tích. Bài báo đã đưa ra các ưu nhược điểm trong từng phương pháp và đề xuất hướng phát triển trong tương lai.
TỪ KHÓA Vận tải Lập kế hoạch Tối ưu hóa Đóng thùng Sắp xếp	

DOI: <https://doi.org/10.34238/tnu-jst.3839>

* Corresponding author. Email: pthuan@ictu.edu.vn

1. Giới thiệu

Vận tải là quá trình di chuyển các vật thể từ một vị trí ban đầu đến vị trí khác, vận tải có vai trò và ý nghĩa quan trọng trong đời sống sinh hoạt, sản xuất của con người [1]. Các hoạt động sơ khai về vận tải đã được hình thành trong xã hội nguyên thủy như khuân, vác, nâng,... Sau này, khi các hình thái kinh tế trở nên đa dạng và phức tạp thì các hình thức vận tải càng được cải tiến và đa dạng hóa. Trong khoảng thời gian gần đây, quá trình vận tải đơn thuần ban đầu đã hình thành nên các dịch vụ vận tải, ngành vận tải (*logistics*) đã trở thành ngành kinh tế - kỹ thuật quan trọng để gắn kết quá trình sản xuất và quá trình giao lưu, phân phối hàng hóa.

Trong nhiều thập kỷ gần đây, container [2] là một đơn vị trọng tải thiết yếu có tầm quan trọng trong vận tải hàng hóa đường biển quốc tế. Với việc ngày càng gia tăng số lượng cảng biển và số lượng container tại mỗi cảng đã đưa đến sự cạnh tranh trong vận tải đường biển. Vấn đề nghiên cứu để cải thiện những hoạt động tại cảng container hiện nay đang được nhiều nhà khoa học quan tâm. Vấn đề này rất khó thực hiện nếu các phương án không được nghiên cứu, thử nghiệm hiệu quả sử dụng các phương pháp tối ưu hóa thích hợp. Trên thế giới và Việt Nam hiện nay, vấn đề khai thác cảng biển, rút ngắn thời gian chờ tàu và bốc dỡ hàng hóa đang nhận được sự quan tâm từ nhiều nhà nghiên cứu. Với sự phát triển của giao thương hàng hoá, số lượng tàu và container cập cảng là rất lớn. Hàng loạt các yếu tố liên quan đến vấn đề khai thác cảng biển như: bến bãi, thiết bị, nhân lực, công nghệ bốc xếp, chủng loại hàng hóa, hạ tầng kết nối với các loại hình, phương thức vận tải khác, thủ tục hành chính,...

Trong bài báo này, chúng tôi giới thiệu một số hướng tiếp cận liên quan đến tối ưu công nghệ sắp xếp hàng hóa phục vụ cho vận tải tại cảng biển, từ đó đề xuất một số hướng phát triển cho các nghiên cứu tiếp theo. Đảm bảo khai thác bến bãi một cách hiệu quả, giảm thiểu thời gian chờ và tiết kiệm chi phí cho tàu, đảm bảo hàng hoá được lưu thông thuận lợi,... Lợi ích của việc tìm cách sắp xếp hợp lý số lượng hàng hóa vào một container sao cho số lượng hàng hoá nhiều nhất là vấn đề được các công ty vận tải rất quan tâm. Chính vì thế bài báo này chúng tôi đã tổng quan lại các vấn đề liên quan đến bài toán Bin Packing 2D, tìm hiểu các hướng giải quyết bài toán và đưa ra đề xuất về một phương pháp tối ưu để giải bài toán này nhằm mục đích nâng cao hiệu quả của quá trình sắp xếp hàng hóa tại cảng biển.

Cũng trong nghiên cứu này, chúng tôi tập trung tìm hiểu bài toán Bin Packing 2D (*bài toán đóng thùng 2 chiều*) [3] và đưa ra một số cách giải quyết bài toán dựa trên kỹ thuật quy hoạch ràng buộc. Kỹ thuật này phù hợp để giải quyết các bài toán có không gian tìm kiếm lớn và các biến trong bài toán đó phụ thuộc vào nhau theo một hay nhiều quy luật. Bài toán đóng thùng có nhiều ứng dụng trong thực tiễn. Trong bài toán xếp gỗ, các miếng gỗ nguyên tấm có kích thước cố định và cần xếp các miếng gỗ đó ra thành các miếng gỗ nhỏ hơn để làm nhà. Yêu cầu đặt ra là hãy đưa các miếng gỗ nguyên tấm đó thành các mẫu nhỏ sao cho số miếng gỗ nguyên tấm được sử dụng là ít nhất? Hoặc trong một lĩnh vực ứng dụng khác như trong việc lưu dữ liệu trong máy tính, dữ liệu là một tập các file cần được lưu trữ trên một tập các thiết bị nhớ giống nhau. Cần phải lưu trữ sao cho một file chỉ được nằm trong một thiết bị nhớ và số các thiết bị nhớ cần dùng là ít nhất. Trong bài toán vận tải sắp xếp hàng hóa tại cảng biển [4], [5] cần xếp một dãy các đồ vật (*hàng hóa, đồ đạc, container,...*) lên các phương tiện vận chuyển (*xe tải, toa xe lửa,...*). Biết rằng, mỗi vật có kích thước và khối lượng xác định, các phương tiện vận chuyển giống nhau cùng có sức chứa và trọng tải xác định. Cần sắp xếp số hàng hóa **sao cho** số phương tiện vận tải cần dùng là ít nhất, đảm bảo các đồ vật đều nằm trong khoang chứa và tổng trọng lượng của chúng không vượt quá trọng tải của phương tiện,... và còn nhiều các lĩnh vực khác như trong bài toán cắt vải, bài toán cắt sắt, bài toán về lắp đặt cột viễn thông, bài toán sắp xếp lịch chương trình truyền hình. Tuy nhiên, một câu hỏi đặt ra cho các nghiên cứu về chủ đề này là làm thế nào để thu được kết quả tối ưu nhất?

Để trả lời cho câu hỏi này, bài báo đã tìm hiểu các hướng giải quyết bài toán Bin Packing 2D hiện nay. Sau đó, tìm hiểu về một số phương pháp cụ thể trong các hướng này để từ đó đưa ra các

ưu nhược điểm trong từng phương pháp và đề xuất hướng phát triển trong tương lai. Phương pháp nghiên cứu sử dụng là khảo cứu và tổng hợp dựa trên các bài báo truy xuất từ Google Scholar của các nhà xuất bản uy tín như IEEE, Elsevier, Springer,... để lọc ra các bài báo gốc và các bài có ảnh hưởng nhất để khảo sát và phân tích. Nghiên cứu này có ý nghĩa khảo cứu cho các bài báo chuyên sâu về sau.

Các phần tiếp theo của bài báo được trình bày như sau: Trong phần II chúng tôi trình bày về phương pháp nghiên cứu, trong đó giới thiệu về bài toán đóng thùng 2 chiều và các lĩnh vực ứng dụng. Cũng trong phần II, chúng tôi trình bày một số hướng tiếp cận để giải quyết bài toán đóng thùng 2 chiều và đánh giá độ phức tạp của từng phương pháp. Trong phần III trình bày các kết quả nghiên cứu và các thảo luận liên quan. Phần kết luận, đánh giá, đề xuất hướng phát triển trong thời gian tới sẽ được trình bày trong phần IV.

2. Phương pháp nghiên cứu

2.1. Bài toán đóng thùng

2.1.1. Phát biểu bài toán

Bài toán đóng thùng [5] (bin packing problem)

Cho một dãy các đồ vật $L = (a_1, a_2, \dots, a_n)$ và các thùng giống nhau có cùng sức chứa B . Kích thước của đồ vật a_i là s_i lớn hơn 0 và không vượt quá sức chứa của thùng ($0 \leq s_i \leq B \forall i = 1, 2, \dots, n$). Yêu cầu của bài toán: hãy xếp tất cả các đồ vật trên vào các thùng chứa sao cho số thùng sử dụng là ít nhất.

Một cách đầy đủ thì bài toán vừa phát biểu ở trên được gọi là bài toán đóng thùng dạng cơ bản (*classical bin packing problem*), để phân biệt với một số dạng tổng quát hơn của bài toán đóng thùng. Bài toán đóng thùng dạng cơ bản đã được chứng minh là một bài toán *NP - khó*.

Mô hình quy hoạch nguyên của bài toán

Để thấy số thùng cần sử dụng không vượt quá n , đưa vào các biến số x_{ij} với ý nghĩa như sau:

$$x_{ij} = \begin{cases} 1 & \text{nếu đồ vật } i \text{ được xếp vào thùng } j \\ 0 & \text{nếu ngược lại} \end{cases}$$

$$i, j = 1, 2, \dots, n$$

Khi đó bài toán đóng thùng có thể được phát biểu dưới dạng bài toán quy hoạch nguyên tuyến tính như sau:

$$\text{Tìm cực tiểu: } m = \sum_{j=1}^n \text{sign} \left(\sum_{i=1}^n x_{ij} \right) \text{ với điều kiện: } \sum_{j=1}^n x_{ij} = 1, \quad i = \overline{1, n} \quad (1);$$

$$\sum_{j=1}^n x_{ij} \leq B, \quad i = \overline{1, n} \quad (2); \quad x_{ij} \in \{0, 1\}, \quad i = \overline{1, n}; \quad j = \overline{1, n} \quad (3)$$

Điều kiện (1) nghĩa là mỗi đồ vật phải được xếp vào đúng một và chỉ một thùng. Điều kiện (2) nghĩa là tổng kích thước các đồ vật được xếp vào một thùng không vượt quá sức chứa của thùng đó.

Đánh giá hiệu quả thuật toán xấp xỉ giải bài toán đóng thùng

Bài toán đóng thùng là bài toán *NP - khó* nên phần nhiều các thuật toán giải là thuật toán xấp xỉ. Để đánh giá hiệu quả của một thuật toán xấp xỉ A cho bài toán đóng thùng thường sử dụng các chỉ số hiệu năng tiệm cận tồi nhất R_A^∞ (*asymptotic worst-case performance ratio*) và chỉ số hiệu năng tuyệt đối tồi nhất R_A (*absolute worst-case performance ratio*) được định nghĩa như sau:

Cho dãy đồ vật L . Gọi $A(L)$ là số thùng thuật toán A sử dụng để xếp dãy đồ vật L và $OPT(L)$ là số thùng tối ưu cho dãy đồ vật L . Đặt $R_A(L) = \frac{A(L)}{OPT(L)}$

Định nghĩa 2.1. Chỉ số hiệu năng tiệm cận tối nhất R_A^∞ là tỉ số tiệm cận giữa kết quả của thuật toán A đối với kết quả tối ưu của bài toán trong tình huống tối nhất.

$$R_A^\infty = \inf\{r \geq 1: \exists N_0 \in \mathbb{Z}^+ \text{ để } R_A(L) \leq r \forall L \text{ thỏa mãn điều kiện } OPT(L) \geq N_0\}$$

Định nghĩa 2.2. Chỉ số hiệu năng tuyệt đối tối nhất là tỉ số tuyệt đối giữa kết quả hoạt động của thuật toán A với kết quả tối ưu của bài toán trong tình huống tối nhất.

$$R_A = \inf\{r \geq 1 \text{ sao cho } R_A(L) \leq r \text{ với mọi } L\}$$

Trong một số tình huống có thể biết thêm một thông tin là các đồ vật đều có kích thước không vượt quá một giá trị xác định nào đó. Gọi α là cận trên của kích thước của các đồ vật, tức là $s(a_i) \leq \alpha \forall i = 1, 2, \dots, n$. Khi đó có:

$$R_A(\alpha) = \inf\{r \geq 1 \text{ sao cho } R_A(L) \leq r \text{ với mọi } L \text{ và } s(a_i) \leq \alpha \forall a_i \in L\}$$

$$R_A^\infty(\alpha) = \inf\{r \geq 1: \exists N_0 \in \mathbb{Z}^+ \text{ để } R_A(L) \leq r \forall L \text{ thỏa } OPT(L) \geq N_0 \text{ và } s(a_i) \leq \alpha \forall a_i \in L\}$$

R_A^∞ và R_A là tiêu chuẩn quan trọng để đánh giá hiệu quả của các thuật toán xếp xi.

2.1.2. Các biến thể của bài toán đóng thùng

Ngoài dạng cơ bản, bài toán đóng thùng còn có một số biến thể khác. Về cơ bản các dạng bài toán biến thể này giống với bài toán cơ bản nhưng có thay đổi (hoặc thêm vào) một số điều kiện. Dưới đây là một số dạng biến thể quan trọng của bài toán đóng thùng:

1) Hãy xếp các đồ vật sao cho kích thước các đồ vật hoặc tổng số lượng các đồ vật trong thùng là lớn nhất khi số lượng thùng sử dụng và dung lượng các thùng không đổi.

2) Tìm kích thước chung tối thiểu của m thùng chứa để có thể xếp được tất cả các đồ vật.

3) Các đồ vật lần lượt xuất hiện theo thứ tự cho trước và đồ vật trước phải được xếp vào thùng xong thì đồ vật sau mới xuất hiện (bài toán đóng thùng trực tuyến – online bin packing).

4) Cho trước một hằng số thực $r \in [0, 1]$. Cần xếp các đồ vật sao cho số thùng sử dụng là ít nhất và mỗi thùng nếu được sử dụng thì tổng dung lượng các đồ vật phải đạt ít nhất là $r \times$ dung lượng thùng.

5) Cho trước một hằng số nguyên $k > 0$. Cần xếp các đồ vật sao cho số thùng sử dụng là ít nhất và số lượng đồ vật được xếp vào một thùng không được quá k .

6) Mỗi đồ vật có một khoảng thời gian tồn tại nhất định đánh dấu bởi một thời điểm bắt đầu và một thời điểm kết thúc. Trước khoảng thời gian tồn tại của mình, mỗi đồ vật chưa cần phải xếp vào thùng nào cả. Trong khoảng thời gian tồn tại của mình thì đồ vật phải được xếp vào một thùng nào đó. Sau khoảng thời gian tồn tại của mình thì đồ vật không cần phải được xếp vào thùng nữa nên có thể được lấy ra để nhường khoảng trống cho đồ vật khác. Bài toán này gọi là bài toán đóng thùng động (dynamic bin packing).

Sắp xếp các đồ vật sao cho số thùng sử dụng là ít nhất và một số đồ vật nhất định không được xếp chung với nhau vào một thùng.

7) Dạng đối ngẫu của bài toán đóng thùng là bài toán phủ thùng (bin covering problem) được phát biểu như sau:

Cho các đồ vật $L = (a_1, a_2, \dots, a_n)$, với kích thước của mỗi đồ vật a_i là s_i , $i = 1, 2, \dots, n$. Hãy xếp tất cả các đồ vật trên vào các thùng chứa sao cho:

i. $s_i \leq 1 \forall i$

ii. Dung lượng của các thùng là tùy ý

iii. Số lượng thùng có tổng dung lượng các đồ vật lớn hơn hoặc bằng 1 là nhiều nhất.

8) Bài toán đóng thùng với dung lượng thùng chứa thay đổi (variable-sized bin packing):

Cho dãy các đồ vật $L = (a_1, a_2, \dots, a_n)$, kích thước đồ vật a_i là $s_i \in [0, 1]$;

($i = 1, 2, \dots, n$). Có r loại thùng chứa khác nhau: $B_1, B_2, \dots, B_r$, dung lượng thùng chứa loại j là B_j ($j = 1, 2, \dots, r$) và thỏa mãn: $1 = s(B_1) > s(B_2) > \dots > s(B_r)$. Hãy xếp tất cả các đồ vật trên vào các thùng chứa sao cho tổng dung lượng các thùng được sử dụng là nhỏ nhất.

Để thấy nếu $r = 1$ thì bài toán trở thành bài toán đóng thùng dạng cơ bản.

9) Bài toán đóng thùng đa chiều (*multi-dimensional bin packing*)

Trong bài toán đóng thùng d chiều (d là một số nguyên dương, $d \geq 1$), có các thùng chứa là các siêu cúb d chiều kích thước là $B_1, B_2, \dots, B_d$ và dãy các đồ vật $a_1, a_2, \dots, a_n$, trong đó đồ vật a_i có kích thước là $s_1(a_i) \times s_2(a_i) \times \dots \times s_d(a_i)$. $s_j(a_i)$ là kích thước chiều thứ j của đồ vật a_i ; $0 < s_j(a_i) \leq B_j$; $1 \leq i \leq n$; $1 \leq j \leq d$.

Một cách xếp các đồ vật gọi là hợp lệ nếu mỗi đồ vật a_i sẽ được gán một vector tọa độ $p(a_i) = (x_1(a_i), x_2(a_i), \dots, x_d(a_i))$ sao cho:

(1) $x_j(a_i) \geq 0$; $1 \leq i \leq n$; $1 \leq j \leq d$.

(2) $x_j(a_i) + s_j(a_i) \leq B_j$; $1 \leq i \leq n$; $1 \leq j \leq d$.

(3) $(x_j(a_i); x_j(a_i) + s_j(a_i)) \cap (x_j(a_k); x_j(a_k) + s_j(a_k)) = \emptyset$; $1 \leq i \neq k \leq n$; $1 \leq j \leq d$

Điều kiện (1) và (2): tất cả các đồ vật đều nằm gọn trong không gian của thùng chứa.

Điều kiện (3): trong mọi thùng chứa, các đồ vật không được xếp chồng lên nhau.

Các điều kiện trên áp dụng với bài toán đóng thùng đa chiều hình học (*geometric multi-dimension bin packing*), tức là mỗi chiều kích thước là một chiều không gian. Nếu mỗi chiều kích thước không phải là một chiều không gian, ví dụ chiều thứ nhất biểu diễn độ dài, chiều thứ 2 biểu diễn khối lượng, chiều thứ 3 biểu diễn thời gian, ... thì điều kiện ràng buộc chỉ là tổng kích thước các đồ vật theo một chiều sẽ không vượt quá dung lượng chiều đó của thùng chứa:

$$\sum_{i=1}^n s_i(a_i) \leq B, \quad j = \overline{1, d}$$

Yêu cầu của bài toán là hãy xếp các đồ vật hợp lệ sao cho số lượng thùng cần sử dụng là ít nhất.

Để thấy với $d = 1$ thì bài toán suy biến về bài toán đóng thùng dạng cơ bản.

2.1.3. Ứng dụng của bài toán đóng thùng

Không chỉ đóng vai trò quan trọng trong nghiên cứu lý thuyết, đặc biệt là các nghiên cứu về thuật toán xấp xỉ, bài toán đóng thùng còn có nhiều ứng dụng trong thực tiễn.

Dưới đây là một số ứng dụng trong đó bài toán đóng thùng đóng vai trò là bài toán chính:

1) Bài toán xẻ gỗ (*stock cutting*): Cho các miếng gỗ xẻ nguyên tấm có chiều ngang cố định và chiều dài giống nhau bằng C . Cần chia các miếng gỗ xẻ nguyên tấm đó thành các mẫu nhỏ $\{a_i\}$ hơn dùng để làm nhà. Hãy chia các miếng gỗ nguyên tấm đó thành các mẫu $\{a_i\}$ sao cho số miếng gỗ nguyên tấm phải dùng là ít nhất. Cũng thấy dạng tương tự khi phải chia các cuộn dây cáp, ống dẫn nước nguyên vẹn, ... thành các đoạn nhỏ để lắp đặt cho một hệ thống nào đó và cần chia khéo sao cho đỡ lãng phí nhất.

2) Lưu dữ liệu trên máy tính (*memory allocation*): Các dữ liệu là một tập hợp các file cần được lưu trên một tập hợp các thiết bị nhớ giống nhau (đĩa CD, đĩa cứng, đĩa mềm, hoặc bộ nhớ bán dẫn USB, ...). Cần phải lưu các file trên các thiết bị nhớ sao cho 1 file chỉ được nằm trong 1 thiết bị nhớ và số thiết bị nhớ cần dùng là ít nhất.

3) Bài toán vận tải (*transportation*): Cần xếp một dãy các đồ vật (*đồ đạc, hàng hóa, ...*) $\{a_i\}$ lên các phương tiện vận tải (*xe tải, toa xe lửa, ...*). Biết rằng mỗi đồ vật có kích thước (3 chiều) và trọng lượng xác định. Các phương tiện vận tải là giống nhau có cùng sức chứa và trọng tải xác định. Cần xếp các đồ vật lên các phương tiện vận tải sao cho số phương tiện cần dùng là ít nhất mà vẫn đảm bảo các đồ vật đều nằm gọn trong khoang chứa và tổng trọng lượng của chúng không vượt quá trọng tải của phương tiện vận tải. Bài toán này có thể quy về bài toán đóng thùng 4 chiều với 3 chiều đầu là 3 chiều không gian, chiều thứ 4 là trọng lượng.

4) Xếp lịch chương trình truyền hình (*television programming*): thông thường các mẫu chương trình quảng cáo được xếp vào các khoảng thời gian xen kẽ trong các chương trình truyền hình chính. Các khoảng thời gian này có độ dài cố định (*ví dụ 5 phút*). Yêu cầu là hãy xếp các mẫu quảng cáo vào những khoảng thời gian này sao cho số khoảng thời gian cần sử dụng là ít nhất.

Ngoài ra trong một số ứng dụng, bài toán đóng thùng đóng vai trò là bài toán phụ hỗ trợ việc giải tối ưu bài toán chính, ví dụ bài toán lập lộ trình vận chuyển có hạn chế khả năng (*capacitated vehicle routing problem*) trong đó phải bố trí các xe chở hàng có trọng tải hữu hạn đi chuyển hàng cho các vị khách sao cho mọi khách hàng đều nhận được hàng theo đơn đặt hàng và chi phí vận chuyển là thấp nhất. Thông thường đội xe sẽ chia thành các nhóm nhỏ, mỗi nhóm gồm một vài xe đi phục vụ theo một tuyến đường. Trên mỗi tuyến đường, cần xếp các hàng hóa của các vị khách trên tuyến đường đó lên các xe sao cho số xe trong nhóm cần dùng là ít nhất. Đó chính là bài toán đóng thùng.

2.2. Tổng quan các phương pháp giải bài toán đóng thùng

Bài toán đóng thùng là một bài toán tối ưu tổ hợp nổi tiếng và có nhiều thuật toán để giải, đặc biệt là các thuật toán xấp xỉ. Bài toán đóng thùng đóng vai trò quan trọng trong việc nghiên cứu các thuật toán xấp xỉ khi được chọn là một trong những bài toán đầu tiên để nghiên cứu xây dựng các thuật toán xấp xỉ và đánh giá hiệu năng của chúng, ví dụ như đánh giá hiệu quả thuật toán xấp xỉ trong thời gian tối nhất, đánh giá cận dưới cho hiệu năng các thuật toán tức thì (*online algorithms*), phân tích hiệu năng thuật toán dựa trên xác suất thống kê,...

Phần dưới đây sẽ trình bày một số thuật toán quan trọng để giải bài toán đóng thùng, qua đó giới thiệu được phần nào các kết quả nghiên cứu hiện nay về bài toán đóng thùng.

2.2.1. Các thuật toán trực tiếp

2.2.1.1. Khái niệm thuật toán trực tiếp

Một thuật toán được gọi là thuật toán trực tiếp (*online algorithm*) khi nó phải thực hiện xếp đồ vật đang xét vào thùng chứa mà không được biết bất kì thông tin gì về các đồ vật xuất hiện sau nó. Đây là một ràng buộc thường gặp trong thực tế, ví dụ như người bán hàng thì không biết khách hàng tiếp theo sẽ có yêu cầu như thế nào.

Các thuật toán online phổ biến bao gồm Next Fit, First Fit và Best Fit.

2.2.1.2. Thuật toán Next Fit

Mô tả quy tắc xếp các đồ vật của Next Fit (NF)

NF sẽ đặt đồ vật đang xét vào ngay thùng đang mở (*trong NF có 1 và chỉ 1 thùng được mở tại một thời điểm*) và cũng là thùng tiếp nhận đồ vật vừa được xét gần đây nhất nếu như thùng đó còn đủ chỗ trống. Ngược lại, nó sẽ đóng thùng đó lại, mở thùng mới rồi đặt đồ vật vào đó. Sau đó tiếp tục xét đến đồ vật tiếp theo. Quá trình bắt đầu với đồ vật đầu tiên và kết thúc khi tất cả các đồ vật đều đã được xếp hết vào thùng.

Next Fit có thể được mô tả bằng đoạn giả ngôn ngữ sau:

```
bin_index = 1; // chỉ số cho biết đã có bao nhiêu thùng đã được sử dụng
item_index = 1;
// chỉ số cho biết đang xét đồ vật thứ bao nhiêu
Open B[bin_index]; // mở thùng đầu tiên để tiếp nhận đồ vật
active_bin = B[bin_index];
// B[] là một mảng chứa các thùng được sử dụng trong thuật toán
while (item_index <= n)
{
    active_item = a[item_index];
    if (Size(active_item) <= FreeSize(active_bin)) Pack active_item into active_bin;
    else{
        Close active_bin; bin_index ++; Open B[bin_index];
        active_bin = B[bin_index];
    }
    item_index ++;
}
```

```
Pack active_item into active_bin;
}
item_index++; } // while
Để thấy NF có độ phức tạp tính toán là  $O(n)$ .
```

2.2.1.3. Thuật toán First Fit

Mô tả quy tắc xếp đồ vật của First Fit (FF):

Thuật toán FF sẽ xếp đồ vật đang xét vào thùng đầu tiên (tức là thùng có chỉ số nhỏ nhất) trong số những thùng đã mở và còn đủ chỗ để chứa đồ vật đó. Nếu trong số những thùng đã mở không có thùng nào còn đủ chỗ trống để xếp đồ vật đang xét thì FF sẽ mở một thùng mới và đặt đồ vật vào đó. Sau đó chuyển sang xét đồ vật tiếp theo. Quá trình đóng thùng kết thúc khi đã xếp hết tất cả các đồ vật vào thùng chứa. Để thấy FF không đóng một thùng nào và số lượng thùng ở trạng thái hoạt động là không cố định.

Đoạn giả ngôn ngữ sau mô tả hoạt động của FF:

```
bin_index = 1; // chỉ số cho biết đã có bao nhiêu thùng đã được sử dụng item_index = 1;
// chỉ số cho biết đang xét đồ vật thứ bao nhiêu Open B[bin_index];
// mở thùng đầu tiên để tiếp nhận đồ vật
// B[] là một mảng chứa các thùng được sử dụng trong thuật toán while (item_index <= n)
{
    test = false; count = 1; do{
        if (Size(a[item_index] <= FreeSize(B[count])){
            Pack a[item_index] into B[count]; test = true;
        }else count++;
    }while (test == false) && (count <= bin_index); if (test == false){
        bin_index++; Open B[bin_index];
        Pack a[item_index] into B[bin_index];
    }
    item_index++;
}
```

Độ phức tạp tính toán của thuật toán FF là $O(n \log n)$.

2.2.1.4. Thuật toán Best Fit

Mô tả xếp đồ vật

Quy tắc đóng thùng của Best Fit (BF) khá đơn giản, BF sẽ xếp đồ vật đang xét vào thùng còn ít chỗ trống nhất trong số những thùng đã mở và còn đủ chỗ trống để tiếp nhận đồ vật. Nếu không có thùng đã mở nào có đủ chỗ để tiếp nhận đồ vật đang xét, BF sẽ mở một thùng mới và đặt đồ vật vào đó. Sau đó chuyển sang xét thùng tiếp theo. Quá trình kết thúc khi tất cả các đồ vật đều đã được xếp hết vào thùng. Như vậy BF ưu tiên tìm thùng “vừa vặn” (fit) với đồ vật nhất.

BF có thể được mô tả qua đoạn giả ngôn ngữ sau::

```
bin_index = 1; // chỉ số cho biết đã có bao nhiêu thùng đã được sử dụng item_index = 1;
// chỉ số cho biết đang xét đồ vật thứ bao nhiêu Open B[bin_index];
// mở thùng đầu tiên để tiếp nhận đồ vật
// B[] là một mảng chứa các thùng được sử dụng trong thuật toán while (item_index <= n)
{
    index = bin_index + 1;
    fit = 1 - Size(a[item_index]); for(j = 1; j <= bin_index; j++){
        temp = FreeSize(B[j]) - Size(a[item_index]); if (0 <= temp) && (temp < fit){
            fit = temp; index = j;
        }
    }
}
```

```

if (index < bin_index + 1)
    Pack a[item_index] into B[index];
else{
    bin_index++;
    Open B[bin_index];
    Pack a[item_index] into B[bin_index];
}
item_index++;
}

```

Thuật toán BF đã được chứng minh là có hiệu quả tương đương với FF.

Ngoài các thuật toán NF, BF và FF vừa đề cập ở trên, còn có một số thuật toán online khác như *Group- X Fit (GXF)*, thuật toán First Fit cải tiến (*Refined First Fit*),... Tuy nhiên chúng tôi sẽ không trình bày những thuật toán này ở đây.

2.2.2. Các phương pháp không trực tiếp: First Fit Decreasing và Best Fit Decreasing

Khác với ràng buộc trong các thuật toán online, trong thuật toán không trực tiếp (*off-line algorithm*), được biết trước toàn bộ danh sách các đồ vật trước khi xếp chúng vào thùng và điều này cho phép thực hiện một số thao tác tiền xử lý để nâng cao hiệu quả công việc.

Mục này sẽ đề cập đến 2 thuật toán off-line phổ biến và cũng rất hiệu quả là First Fit Decreasing (FFD) và Best Fit Decreasing (BFD). Về cơ chế hoạt động của FFD và BFD thì rất đơn giản, chỉ việc sắp xếp dãy các đồ vật cho trước theo thứ tự giảm dần về kích thước, sau đó áp dụng thuật toán online tương ứng (FF cho FFD hoặc BF cho BFD) để xếp các đồ vật vào thùng.

Chi phí thời gian cho FFD và BFD đều là $\Omega(n \log n)$, bởi vì việc sắp xếp theo chiều tăng dần về kích thước dãy n đồ vật cần thời gian là $\Omega(n \log n)$, còn việc xếp các đồ vật vào thùng theo quy tắc BF hoặc FF đều cùng yêu cầu thời gian là $O(n \log n)$ như đã nói ở phần trước.

3. Kết quả và bàn luận

Vấn đề tối ưu trong ngành vận tải ngày càng được nhiều nhà khoa học quan tâm nghiên cứu, điều đó giúp thúc đẩy quá trình lưu thông hàng hóa, tiết kiệm được chi phí vận chuyển cho các doanh nghiệp vận tải. Trong bài báo này, chúng tôi đã đưa ra vấn đề tối ưu trong quá trình vận tải, cụ thể là tối ưu sắp xếp trong bài toán Bin Packing 2D và các lĩnh vực ứng dụng của bài toán. Bài toán Bin Packing 2D là bài toán tối ưu tổ hợp có độ phức tạp cao, có nhiều thuật toán để giải, đặc biệt là các thuật toán xấp xỉ. Trong đó chúng tôi đã đưa ra một số hướng để giải quyết bài toán này: Các thuật toán trực tiếp (*Next Fit*, *First Fit* và *Best Fit*), các phương pháp không trực tiếp (*First Fit Decreasing*, *Best Fit Decreasing*), cũng như giả ngôn ngữ và đánh giá độ phức tạp của các thuật toán.

Đối với bài toán tối ưu sắp xếp container tại cảng biển có thể quy về bài toán Bin Packing 2D, trong đó chính là quá trình tối ưu sắp xếp container tại kho bãi. Khi có một đoàn tàu chở hàng cập cảng, sau khi hoàn thành các thủ tục hành chính được tiến hành bốc dỡ hàng hoá. Hàng hoá rất đa dạng bao gồm các sản phẩm xuất nhập khẩu như các sản phẩm công - nông nghiệp, các sản phẩm điện tử - điện lạnh,... Hàng được chứa trong các container hàng, có các đặc tính như kích thước, trọng lượng, tính chất,... riêng. Sau đó, các container hàng sẽ được các cần trục bốc lên các xe chở hàng chuyên dụng và sắp xếp vào kho hàng (*sân bãi*) theo từng khu vực riêng biệt, tùy thuộc vào đặc tính. Tiếp theo, dựa vào nhu cầu của tàu hàng mà các container hàng xuất sẽ được lấy ra và sắp xếp lên tàu. Mặt khác, một số lượng lớn container hàng sẽ được cho lên các xe đầu kéo (*xe ô tô container*) để chuyển hàng đi trên đất liền. Do đó bài toán tối ưu sắp xếp container tại cảng biển cũng là một vấn đề đáng được quan tâm nghiên cứu.

4. Kết luận

Trong nghiên cứu này chúng tôi nêu và giải quyết bài toán Bin Packing 2D bằng các phương pháp xấp xỉ. Tuy nhiên trong thực tế còn một số hướng nghiên cứu và giải quyết với nhiều

phương pháp khác như sử dụng tính toán mềm, sử dụng tính toán tiến hóa, giải thuật di truyền,... Đây là những hướng quan tâm nghiên cứu trong thời gian sắp tới.

Mặt khác, trong bài báo mới chỉ đưa ra và giải quyết bài toán Bin Packing 2D, trong thời gian tới chúng tôi sẽ nghiên cứu và giải quyết bài toán Bin Packing 3D [6]. Đây là bài toán tối ưu với không gian 3 chiều, với độ phức tạp tính toán lớn. Lời giải của bài toán này nhằm áp dụng cho bài toán tối ưu sắp xếp container tại cảng biển.

Lời cảm ơn

Nghiên cứu này được tài trợ bởi Trường Đại học Công nghệ Thông tin và Truyền thông – Đại học Thái Nguyên trong đề tài mã số T2020-07-18.

TÀI LIỆU THAM KHẢO/ REFERENCES

- [1] D. Topolšek, K. Čižiūnienė, and T. C. Ojsteršek, "Defining transport logistics: a literature review and practitioner opinion based approach," *Transport*, vol. 33, no. 5, pp. 1196-1203, 2018.
- [2] T. E. Notteboom, "Container shipping and ports: an overview," *Review of network economics*, vol. 3, no. 2, pp. 86-106, 2004.
- [3] J. F. Gonçalves and M. G. Resende, "A biased random key genetic algorithm for 2D and 3D bin packing problems," *International Journal of Production Economics*, vol. 145, no. 2, pp. 500-510, 2013.
- [4] N. Kemme, *Design and operation of automated container storage systems*. Springer Science & Business Media, 2012.
- [5] C. Blum and V. Schmid, "Solving the 2D bin packing problem by means of a hybrid evolutionary algorithm," *Procedia Computer Science*, vol. 18, pp. 899-908, 2013.
- [6] Y. Wu, W. Li, M. Goh, and R. de Souza, "Three-dimensional bin packing problem with variable bin height," *European journal of operational research*, vol. 202, no. 2, pp. 347-355, 2010.