

**ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

NGUYỄN HỮU MÙI

**THUẬT TOÁN VÀ
CÁC BÀI TOÁN LỊCH BIỂU**

**Chuyên ngành: Khoa học máy tính
Mã số: 62 48 01 01**

TÓM TẮT LUẬN ÁN TIẾN SĨ CÔNG NGHỆ THÔNG TIN

Hà Nội - Năm 2013

**Công trình này được hoàn thành tại Trường Đại học Công nghệ
Đại học Quốc gia Hà Nội**

NGƯỜI HƯỚNG DẪN KHOA HỌC:

- 1. PGS. TSKH Vũ Đình Hòa**
- 2. PGS. TS Hoàng Xuân Huấn**

Phản biện 1: PGS. TS Đoàn Văn Ban

Phản biện 2: PGS. TS Huỳnh Quyết Thắng

Phản biện 3: PGS. TS Đỗ Trung Tuấn

**Luận án được bảo vệ trước hội đồng cấp Đại học Quốc gia chấm
luận án tiến sĩ tại Phòng 212, Nhà E3, Trường Đại học Công
Nghệ.**

Vào hồi 9h00, ngày 25 tháng 9 năm 2013

Có thể tìm hiểu luận án tại:

Thư viện Quốc gia Việt Nam.

Trung tâm Thông tin - Thư viện, Đại học Quốc gia Hà Nội.

MỞ ĐẦU

▪ Lý do chọn đề tài

Lập lịch là một trong những chủ đề quan trọng thuộc lĩnh vực vận trù học xuất hiện từ đầu những năm 1950. Mục tiêu chính của lập lịch là phân phối tài nguyên dùng chung một cách hiệu quả nhất cho các tác vụ đồng thời trong toàn bộ thời gian xử lý. Một mô hình chung nhất về lập lịch đó là bài toán lập lịch job shop (Job shop Scheduling Problem - JSP), bài toán này thuộc lớp *NP-hard* (*NP* là lớp các bài toán quyết định có thể giải quyết trong thời gian đa thức trên máy Turing không đơn định). JSP cũng là một trong những bài toán được nghiên cứu nhiều nhất và là một mô hình phát triển tốt về lý thuyết lập lịch. Ngoài ra, một động lực khác giúp cho JSP được thúc đẩy mạnh mẽ là bởi các ứng dụng của nó trong thực tiễn cuộc sống và sản xuất.

Đã có nhiều giải pháp được đề xuất cho bài toán lập lịch job shop. Tuy nhiên, cho tới nay chưa có một tiếp cận nào đã đề xuất giải quyết triệt để bài toán này. Một số vấn đề liên quan tới việc giải quyết bài toán JSP còn tồn tại như sau:

1. Các chuẩn thiết kế thử nghiệm để đánh giá một cách chính xác các thuật toán mới được đề xuất.
2. Tính hội tụ của các thuật toán mới được đề xuất chưa được chứng minh dựa trên cơ sở toán học.
3. Phương pháp luận cho việc kết hợp các kỹ thuật tìm kiếm khác nhau để tạo ra một giải pháp mạnh cho JSP còn chưa được nghiên cứu một cách đầy đủ.

Ở nước ta, việc nghiên cứu về bài toán lập lịch job shop vẫn chưa phát triển. Trong những năm gần đây đã xuất hiện một số báo cáo khoa học nghiên cứu về JSP. Tuy nhiên, kết quả đạt được chưa tương xứng với tầm quan trọng của bài toán này.

Vì những lý do trên, luận án chọn đề tài "*Thuật toán và các bài toán lịch biểu*".

▪ **Mục tiêu của luận án**

Luận án tập trung nghiên cứu một số vấn đề chủ yếu sau đây:

1. Phân tích các tiếp cận đã đề xuất để giải quyết JSP trong những năm qua để thấy được ưu điểm, nhược điểm của mỗi giải pháp. Trên cơ sở đó đề xuất một số hướng nghiên cứu bài toán này.

2. Đề xuất một thuật toán di truyền lai mới cho JSP và song song hóa thuật toán nhằm khắc phục độ phức tạp tính toán vốn có của các bài toán JSP cỡ lớn.

3. Chứng minh tính hội tụ của thuật toán di truyền lai với mã hóa tự nhiên cho JSP mà luận án đề xuất.

▪ **Ý nghĩa khoa học và ý nghĩa thực tiễn của đề tài**

✓ **Ý nghĩa khoa học**

Những đóng góp chính của luận án về khoa học:

1. Nghiên cứu về tổng quan của bài toán: Phân tích, đánh giá, so sánh các tiếp cận đã áp dụng cho các bài toán lập lịch job shop. Trên cơ sở đó đề xuất một số hướng nghiên cứu để giải quyết bài toán này.

2. Nghiên cứu và đề xuất một thuật toán di truyền lai mới kết hợp thuật toán di truyền với các kỹ thuật tìm kiếm khác cho bài toán lập lịch job shop. Thuật toán được song song hóa để giảm thời gian chạy máy cho bài toán.

3. Chứng minh tính hội tụ của thuật toán di truyền lai mới với mã hóa tự nhiên cho bài toán lập lịch job shop sử dụng một công cụ toán học là lý thuyết xích Markov. Qua đó chúng tỏ độ tin cậy của thuật toán mà luận án đề xuất.

✓ Ý nghĩa thực tiễn

1. Luận án có thể được sử dụng để xây dựng giáo trình cho môn chuyên đề tự chọn ở bậc đại học ngành công nghệ thông tin.

2. Luận án có thể được sử dụng làm tài liệu tham khảo cho các sinh viên đại học và học viên cao học ngành công nghệ thông tin làm đề tài về thuật toán di truyền và ứng dụng.

3. Nếu được đầu tư về tài chính và nhân lực, luận án có thể được áp dụng cho các bài toán trong thực tiễn về qui hoạch và tối ưu.

▪ Bố cục của luận án

Luận án được trình bày trong 155 trang, với 39 hình vẽ, 20 bảng, 82 tài liệu tham khảo. Ngoài phần mở đầu, kết luận và phụ lục, luận án được bố cục thành 4 chương như sau:

Chương 1. Tổng quan về thuật toán di truyền và bài toán lập lịch job shop

Chương 2. Hai bài toán con của bài toán lập lịch job shop

Chương 3. Một thuật toán di truyền lai mới cho bài toán lập lịch job shop

Chương 4. Phân tích tính hội tụ của thuật toán di truyền lai mới cho bài toán lập lịch job shop

CHƯƠNG 1. TỔNG QUAN VỀ THUẬT TOÁN DI TRUYỀN VÀ BÀI TOÁN LẬP LỊCH JOB SHOP

1.1. Thuật toán di truyền cổ điển

Thuật toán di truyền (Genetic Algorithm - GA) phỏng theo các quá trình sinh học trong tự nhiên để tối ưu hóa các hàm mục tiêu. GA được đề xuất và nghiên cứu có hệ thống lần đầu tiên bởi John Holland và các cộng sự tại trường đại học Michigan vào năm 1975. Sau đó, GA được phát triển rất nhanh cả về chiều rộng lẫn chiều sâu, cả về lý thuyết lẫn ứng dụng trong thực tiễn. Hiện nay, GA đã được nghiên cứu ở hầu hết các quốc gia trên thế giới và đặc biệt phát triển mạnh ở Mỹ, Trung Quốc, Nhật Bản, Hàn Quốc,... Lý thuyết GA đã được ứng dụng thành công trong rất nhiều lĩnh vực khác nhau như sinh học, khoa học máy tính, kỹ thuật lai ghép, xử lý ảnh,...

- **Cấu trúc của thuật toán di truyền cổ điển**

- ✓ **Mã hóa lời giải**

Trong GA cổ điển, mỗi cá thể được mã hóa bởi một chuỗi nhị phân. Mỗi vị trí trên chuỗi được gọi là một gen và nhận một trong hai giá trị 0 hoặc 1.

- ✓ **Toán tử trao đổi chéo**

Toán tử trao đổi chéo thực hiện trên hai cá thể cha để tạo ra một hoặc hai cá thể con mới bằng cách trao đổi các đoạn gen tương ứng của các lời giải cha. Có một số cách trao đổi chéo sau đây: Trao đổi chéo một điểm, trao đổi chéo hai điểm, trao đổi chéo đồng nhất.

- ✓ **Toán tử đột biến**

Toán tử đột biến sửa đổi một số gen trong một cá thể cha được chọn một cách ngẫu nhiên bằng cách thay đổi các gen có giá trị 0 thành 1 và ngược lại.

✓ **Toán tử chọn lọc**

Cơ chế chọn lọc thực hiện theo nguyên lý bánh xe xổ số. Mỗi cá thể trong quần thể có một xác suất chọn lọc được tính theo công thức: $p_i = \text{eval}(v_i) / F$. Trong đó, $\text{eval}(v_i)$ là giá trị của hàm thích nghi của cá thể v_i , F là tổng các giá trị thích nghi của quần thể.

▪ **Một thủ tục đơn giản cho thuật toán di truyền cổ điển**

Procedure GA

Begin

$t = 0$

Khởi tạo $P(t)$

Đánh giá $P(t)$

While (not điều kiện dừng) do

Begin

Xây dựng tập lời giải trung gian $P'(t)$ từ $P(t)$

Đánh giá $P'(t)$

Chọn lọc $P(t + 1)$ từ $P'(t)$

$t = t + 1$

Đánh giá $P(t)$

End

End

1.2. Các lớp bài toán P , NP , NPC và NP -hard

Trong mục này, luận án trình bày ngắn gọn các khái niệm cơ bản về lớp các bài toán P , NP , NPC và NP -hard.

1.3. Tổng quan về bài toán lập lịch job shop (JSP)

▪ Mô tả bài toán lập lịch job shop tổng quát

Cho một tập n công việc $\{J_i\} 1 \leq i \leq n$, mỗi công việc bao gồm m thao tác được xử lý ở trên một tập m máy $\{M_j\} 1 \leq j \leq m$ và thỏa mãn các ràng buộc sau đây:

1. Mỗi công việc phải được xử lý ở trên mỗi máy theo một trình tự cho trước của các thao tác. Trình tự thực hiện các thao tác của mỗi công việc lần lượt trên các máy được gọi là tuần tự công nghệ.

2. Tại một thời điểm mỗi máy chỉ có thể xử lý nhiều nhất là một công việc.

3. Mỗi máy M_j tùy ý đều có khả năng xử lý một công việc J_i nào đó, phần công việc J_i được xử lý trên máy M_j được gọi là thao tác O_{ij} .

4. Mỗi thao tác O_{ij} phải được xử lý một cách liên tục ở trên máy M_j (không bị ngắt khi đang xử lý).

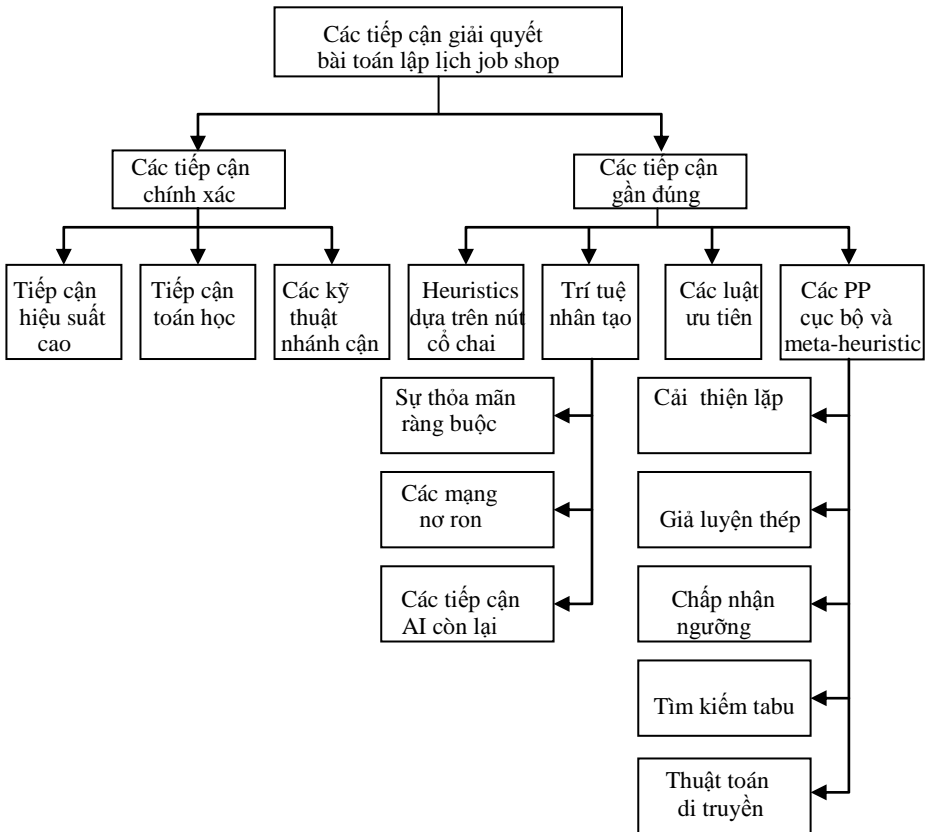
5. Thời gian bắt đầu xử lý và thời gian hoàn thành việc xử lý thao tác O_{ij} được ký hiệu lần lượt là s_{ij} và c_{ij} . Thời gian xử lý thao tác O_{ij} được ký hiệu là p_{ij} .

6. Thời gian hoàn thành việc xử lý tất cả các công việc được gọi là makespan và được ký hiệu là C_{max} .

Việc giải quyết JSP là xác định một lịch biểu sao cho makespan là nhỏ nhất có thể.

▪ Các tiếp cận đã được đề xuất để giải quyết JSP

Sơ đồ tổng kết các tiếp cận chủ yếu đã được áp dụng cho bài toán lập lịch job shop được trình bày trong hình 1.7.



Hình 1.7 - Các tiếp cận cho JSP

- **Một số vấn đề còn tồn tại và các đề xuất**

- ✓ **Một số vấn đề còn tồn tại**

Một số vấn đề cần phải được thảo luận chi tiết và nghiên cứu đầy đủ hơn sau đây:

1. Việc đánh giá kết quả của các giải pháp khác nhau đòi hỏi phải chính xác, đặc biệt các kết quả có liên quan tới heuristic. Các thiết kế thử nghiệm hiện có còn nhiều bất cập, các thiết kế thử

nghiệm cần phải nghiêm ngặt hơn, các bài toán test chuẩn cần phải nhiều hơn và đa dạng hơn.

2. Cần phải chứng tỏ thuộc tính hội tụ tới tối ưu toàn cục của các tiếp cận gần đúng được đề xuất mới cho JSP trên cơ sở lý thuyết.

3. Chưa có một phương pháp hình thức được đề xuất cho việc kết hợp hiệu quả các kỹ thuật tìm kiếm với nhau. Các nghiên cứu nên đưa ra cách thức kết hợp như thế nào để có một tiếp cận mới mạnh hơn.

4. Vấn đề thứ tư là sự phức tạp của các bài toán job shop cỡ lớn với không gian lời giải quá lớn của nó. Do vậy, nên tập trung vào các kỹ thuật tìm kiếm song song trên nhiều vùng lân cận của không gian lời giải.

✓ Một số đề xuất

Luận án xin đề xuất một số hướng nghiên cứu sau:

1. Các phương pháp mới cho JSP nên tập trung vào các kỹ thuật gần đúng để tránh thời gian tăng theo hàm số mũ khi cỡ bài toán tăng theo tuyến tính và nên là các giải pháp lai pha trộn một số kỹ thuật tìm kiếm khác nhau.

2. Các phương pháp mới này nên tích hợp một cách hợp lý các ưu điểm nổi trội của mỗi phương pháp thành phần sao cho phù hợp với đặc thù của mỗi bài toán cần giải quyết.

3. Các phương pháp mới nên tập trung vào các tiếp cận giàu tiềm năng mà chưa được khai thác chẳng hạn như mạng nơ ron.

4. Trong tìm kiếm cục bộ, nên sử dụng khối tới hạn trong các cấu trúc vùng lân cận để tạo ra các di chuyển một cách hiệu quả nhất.

5. Các phương pháp mới nên áp dụng các luật sinh lịch biểu tích cực để hạn chế không gian tìm kiếm và dẫn dắt tới các lời giải tốt hơn.

CHƯƠNG 2. HAI BÀI TOÁN CON CỦA BÀI TOÁN LẬP LỊCH JOB SHOP

Trong thực tiễn, chúng ta gặp nhiều trường hợp các bài toán cần giải quyết chỉ thỏa mãn một số ràng buộc của bài toán lập lịch job shop. Đối với các bài toán con này, cách giải quyết đơn giản hơn bài toán lập lịch job shop rất nhiều. Chương này trình bày hai bài toán con của bài toán lập lịch job shop thường gặp trong thực tiễn sản xuất và đề xuất một thuật toán di truyền với mã hóa tự nhiên cho chúng.

2.1. Bài toán lập lịch flow shop hoán vị

- **Mô tả bài toán**

Bài toán lập lịch flow shop hoán vị (Permutation Flow shop Scheduling Problem - PFSP) là bài toán có n công việc (J_1, J_2, \dots, J_n) được xử lý trên m máy (M_1, M_2, \dots, M_m) và có các đặc trưng sau:

1. Mỗi công việc J_i ($i = 1, \dots, n$) có m thao tác, thao tác thứ j phải được xử lý ở trên máy M_j ($j = 1, \dots, m$). Một công việc chỉ có thể bắt đầu được xử lý ở trên máy M_j nếu nó được hoàn thành việc xử lý ở trên máy M_{j-1} và máy M_j rỗi. không có khoảng thời gian dừng khi chuyển từ máy này sang máy khác.

2. Trình tự xử lý các công việc ở trên tất cả các máy là như nhau. Tức là, nếu một công việc có thứ tự xử lý thứ i ở trên máy M_1 thì nó cũng có thứ tự xử lý thứ i ở trên các máy còn lại.

3. Thao tác của công việc J_i được xử lý ở trên máy M_j được ký hiệu là O_{ij} và có thời gian xử lý cho trước là p_{ij} .

4. Khoảng thời gian kể từ khi bắt đầu xử lý các công việc cho tới khi hoàn thành việc xử lý tất cả các công việc được gọi là makespan của bài toán và được ký hiệu là C_{max} .

Việc giải quyết PFSP là xác định một lịch biểu sao cho makespan là nhỏ nhất.

- **Cách tính thời gian hoàn thành trong một lịch biểu hoán vị**

Gọi j_1, j_2, j_3, j_4, j_5 là một hoán vị của J_1, J_2, J_3, J_4, J_5 và C_{jj_k} là thời gian hoàn thành thao tác thứ j của công việc j_k . Chúng ta có công thức tính thời gian hoàn thành như sau:

$$C_{jj_k} = \begin{cases} \sum_{i=1}^j p_{ij_k} & \text{nếu } j_k = j_1 \\ \sum_{i=1}^k p_{ij_k} & \text{nếu } j = 1 \\ \max(C_{j-1j_k}, C_{jj_{k-1}}) + p_{jj_k} & \{ \text{trường hợp còn lại} \} \end{cases}$$

- **Thuật toán Johnson cho PFSP 2 máy và PFSP 3 máy**

Bài toán lập lịch flow shop hoán vị 2 máy và 3 máy thỏa mãn một số điều kiện nhất định, có thể áp dụng thuật toán Johnson [52] được đề xuất vào năm 1954 để tìm ra lời giải tối ưu thực sự.

- **Một thuật toán di truyền mã hóa tự nhiên cho bài toán lập lịch flow shop hoán vị tổng quát**

Thuật toán Johnson chỉ áp dụng được cho các PFSP 2 máy hoặc 3 máy thỏa mãn ràng buộc (2). Như vậy, với các PFSP tổng quát không thể giải được bằng thuật toán Johnson [52]. Chúng ta phải dùng các phương pháp gần đúng để giải quyết chúng. Trong mục này luận án đề xuất một thuật toán di truyền mới áp dụng cho PFSP. Thuật toán này đã được cài đặt và chạy trên các bài toán test và cho các kết quả rất tốt (tr. 67-73).

2.2. Bài toán lập lịch flow shop

- **Mô tả bài toán**

Bài toán lập lịch flow shop (flow shop scheduling problem - FSP) cũng là bài toán con của JSP nhưng là trường hợp tổng quát hơn bài toán lập lịch flow shop hoán vị. Bài toán này được mô tả tương tự như PFSP, chỉ khác ở chỗ thứ tự xử lý các công việc ở trên mỗi máy có thể khác nhau.

- **Một thuật toán di truyền mã hóa tự nhiên cho bài toán lập lịch flow shop tổng quát**

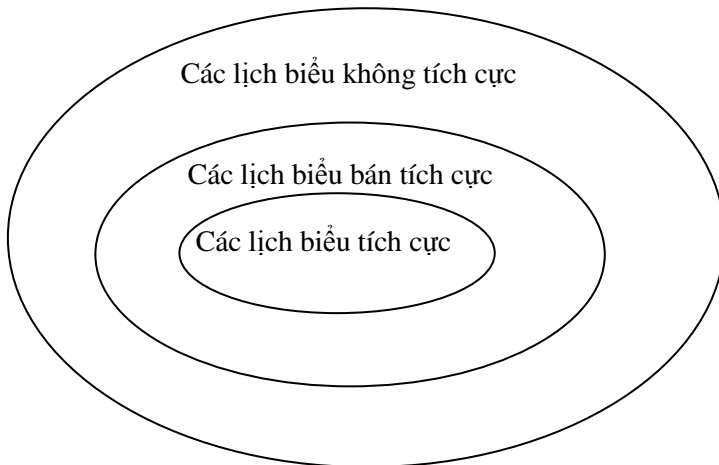
Trong mục này luận án đề xuất một thuật toán di truyền mới áp dụng cho FSP. Thuật toán này đã được cài đặt và chạy trên các bài toán test và cho các kết quả rất tốt (tr. 74-81).

CHƯƠNG 3. MỘT THUẬT TOÁN DI TRUYỀN LAI MỚI CHO BÀI TOÁN LẬP LỊCH JOB SHOP

Trong chương này, luận án đề xuất một thuật toán di truyền lai mới cho bài toán lập lịch job shop. Thuật toán đã được cài đặt và chạy thử nghiệm. Kết quả thử nghiệm đã được so sánh với kết quả được công bố gần đây để chứng tỏ tính vượt trội của thuật toán do luận án đề nghị. Để rút ngắn thời gian chạy máy, thuật toán đã được song song hóa và được chạy thử nghiệm trên các bài toán test chuẩn cho kết quả tốt.

3.1. Các lịch biểu tích cực và bán tích cực

Theo B. Giffler và Thompson [36], không gian các lịch biểu có thể của JSP bao gồm 3 lớp lịch biểu:



Hình 3.1 - Các loại lịch biểu

3.2. Các luật ưu tiên của Giffler và Thompson

- **Thuật toán GT**

Tập luật ưu tiên do Giffler và Thompson [36] đề nghị còn được gọi là thuật toán GT. Thuật toán GT là một trong các công trình sớm nhất về các luật ưu tiên, thuật toán này rất quan trọng và được sử dụng rộng rãi trong lập lịch. Cho tới nay, thuật toán GT vẫn được xem như là nền tảng cho các luật ưu tiên khác. Tầm quan trọng của nó xuất phát từ thực tế đó là nó sinh ra các lịch biểu tích cực.

- **Thuật toán GT áp dụng cho JSP để sinh ra các lịch biểu tích cực**

Một bài toán lập lịch job shop được cho bởi ma trận tuần tự công nghệ $\{T_{ik}\}$ và ma trận thời gian xử lý $\{p_{ik}\}$, đây là 2 ma trận dữ liệu vào của bài toán lập lịch job shop cần giải quyết. Một lịch biểu tích cực có thể được sinh ra bằng cách sử dụng thuật toán GT.

3.3. Một thuật toán di truyền lai mới cho bài toán lập lịch job shop

Trong mục này luận án đề xuất một thuật toán mới cho JSP. Thuật toán có một số cải tiến mới trong việc dùng thuật toán di truyền để giải JSP sau đây:

1. Mã hoá các thao tác của một lịch biểu bởi các số tự nhiên. Cách mã hóa này tạo điều kiện thuận lợi cho việc thực thi các toán tử di truyền và đơn giản hóa trong cài đặt chương trình.

2. Sử dụng chiến lược “đột biến lai”. Vì một trong những điểm yếu của GA là không phù hợp cho việc điều chỉnh các lời giải khi rất gần với lời giải tối ưu vì toán tử trao đổi chéo thường phá vỡ sự điều chỉnh này. Cải tiến này được đưa vào có tác dụng tinh chỉnh các lời

giải hướng tới lời giải tối ưu, nó đặc biệt hữu ích khi một cá thể cha tham gia đột biến mà cá thể này đã gần chạm tới lời giải tối ưu của bài toán.

3. Toán tử trao đổi chéo được thực hiện trên 3 cá thể cha. Cải tiến này có tác dụng tạo ra một cá thể con mang nhiều thuộc tính của các cá thể cha khác nhau, từ đó tăng cường sự khám phá không gian tìm kiếm.

- **Thuật toán tiến hóa**

Procedure NHGA_JSP

Begin

t = 0

Khởi tạo P(t) {hàm InitPopulation}

Đánh giá P(t)

Chọn cá thể tinh hoa

While (not điều kiện dừng) do

Begin

t = t + 1;

Thực hiện phép trao đổi chéo {hàm InitCrossOver3}

Thực hiện phép đột biến {hàm Mutation}

Đánh giá độ thích nghi của mỗi cá thể

Thực hiện chọn lọc {hàm Select}

Xác định cá thể có độ thích nghi cao nhất

Thực hiện sao chép {hàm Copy}

End

End

3.4. Song song hóa thuật toán di truyền lai mới cho bài toán lập lịch job shop

▪ Mô tả thuật toán

Trong giải thuật song song hóa này, luận án áp dụng hình thức song song dữ liệu, bằng cách chia dữ liệu thành nhiều phần, mỗi phần sẽ do một bộ xử lý thực thi. Mô hình máy được áp dụng cho song song hóa là mô hình Master-Slave. Bảng 3.3 nêu một số nhiệm vụ chính của Master và các Slave.

Bảng 3.3 - Nhiệm vụ của Master và Slave

Master	Slave
<ul style="list-style-type: none">- Khởi tạo môi trường để các tiến trình giao tiếp với nhau.- Truyền các tham số: cỡ quần thể, xác suất trao đổi chéo, xác suất đột biến, số thế hệ cho các Slave.	<ul style="list-style-type: none">- Thực hiện thuật toán tuần tự <i>NHGA_JSP</i>
<ul style="list-style-type: none">- Nhận các kết quả từ Slave gửi về	<ul style="list-style-type: none">- Xác định cá thể có độ thích nghi cao nhất gửi về cho Master
<ul style="list-style-type: none">- Lựa chọn kết quả tốt nhất trong các kết quả nhận về từ các Slave- Gửi trở lại cho các Slave làm cá thể tinh hoa	

▪ Thủ tục di truyền song song cho JSP

Procedure PGA_JSP

Begin

Master:

Mở kênh truyền thông và khởi tạo các tuyến đoạn

Gửi các tham số: cỡ quần thể, xác suất trao đổi chéo, xác suất đột biến, số thế hệ cho các Slave

Các Slave:

$t = 0$

Khởi tạo $P(t)$ {hàm InitPopulation}

Đánh giá $P(t)$

Chọn cá thể tốt nhất và gửi về Master

Master:

Chọn cá thể tốt nhất trong các cá thể vừa nhận và gửi trở lại cho các Slave làm cá thể tinh hoa

While (not điều kiện dừng) do

Begin

$t = t + 1$;

Các Slave:

Thực hiện trao đổi chéo {hàm InitCrossOver3}

Thực hiện đột biến {hàm Mutation}

Đánh giá độ thích nghi của mỗi cá thể

Thực hiện chọn lọc {hàm Select}

Xác định độ thích nghi cao nhất

Thực hiện sao chép

Chọn cá thể tốt nhất gửi về Master

Master:

Chọn cá thể tốt nhất trong các cá thể vừa nhận và gửi trở lại cho các Slave làm cá thể tinh hoa

End

End

3.5. Kết quả thử nghiệm

Dựa vào thuật toán NHGA_JSP đề xuất trong mục 3.3, luận án đã cài đặt một chương trình chạy thử nghiệm trên máy PC với bộ vi xử lý có tốc độ 2.8 GHz, hệ điều hành Windows. Kết quả chạy thử nghiệm trên các bài toán test được đề xuất bởi S. Lawrence (1984), Trường Đại Học Quản trị công nghiệp, Đại học Carnegie-Mellon, Pittsburgh, Pennsylvania. Các bài toán test này được đề xuất để thử nghiệm các kỹ thuật lập lịch heuristic. Kết quả chạy thử nghiệm được thống kê trong bảng 3.4.

Bảng 3.4 - Kết quả chạy thử nghiệm trên các bài toán test của Lawrence

(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
Bài toán test	Số công việc	Số máy	Cỡ quần thể	p_c	p_m	Thời gian TB (s)	Kết quả chạy	Tối ưu của BT
LA01	10	5	100	0.8	0.1	100	666	666
LA02	10	5	200	0.8	0.1	120	655	655
LA03	10	5	100	0.8	0.1	150	597	597
LA04	10	5	100	0.8	0.1	200	590	590
LA05	10	5	100	0.8	0.1	250	593	593
LA06	15	5	200	0.8	0.1	250	926	926
LA07	15	5	200	0.8	0.1	80	890	890
LA08	15	5	200	0.8	0.1	20	863	863
LA09	15	5	200	0.8	0.1	250	951	951
LA10	15	5	200	0.8	0.1	250	958	958
LA11	20	5	200	0.8	0.1	150	1222	1222
LA12	20	5	200	0.8	0.1	100	1039	1039

LA13	20	5	200	0.8	0.1	150	1150	1150
LA14	20	5	200	0.8	0.1	150	1292	1292
LA15	20	5	200	0.8	0.1	150	1207	1207
LA16	10	10	300	0.8	0.1	950	945	945
LA17	10	10	300	0.8	0.1	950	794	794
LA18	10	10	300	0.8	0.1	950	848	848
LA19	10	10	300	0.8	0.1	950	842	842
LA20	10	10	300	0.8	0.1	950	907	907
LA21	15	10	400	0.8	0.1	2500	1055	?
LA22	15	10	300	0.8	0.1	2150	927	927
LA23	15	10	300	0.8	0.1	2150	1032	1032
LA24	15	10	400	0.8	0.1	2500	940	?
LA25	15	10	400	0.8	0.1	2500	978	?
LA26	20	10	300	0.8	0.1	2450	1218	1218
LA27	20	10	400	0.8	0.1	3250	1270	?
LA28	20	10	300	0.8	0.1	2450	1216	1216
LA29	20	10	400	0.8	0.1	2450	1190	?
LA30	20	10	300	0.8	0.1	2450	1355	1355
LA31	30	10	300	0.8	0.1	2600	1784	1784
LA32	30	10	300	0.8	0.1	2600	1850	1850
LA33	30	10	300	0.8	0.1	3000	1719	1719
LA34	30	10	300	0.8	0.1	3900	1721	1721
LA35	30	10	300	0.8	0.1	4100	1888	1888
LA36	15	15	300	0.8	0.1	3950	1275	1268
LA37	15	15	300	0.8	0.1	3950	1415	1397
LA38	15	15	400	0.8	0.1	4250	1210	?
LA39	15	15	300	0.8	0.1	3950	1240	1233
LA40	15	15	400	0.8	0.1	4250	1235	?

Bảng thống kê cho thấy đa số các bài toán test của Lawrence đều tìm được lời giải tối ưu thực sự trong khoảng thời gian chạy máy trung bình không dài (cột 7). Ở đây, các bài toán test đều được chạy với số lần lặp 100 lần và chạy thử 10 lần. Cột 8 là các kết quả chạy thuật toán do luận án đề xuất, cột 9 là kết quả tối ưu thực sự của bài toán. Các vị trí có dấu ? là ký hiệu cho biết các bài toán này cho tới nay chưa biết lời giải tối ưu thực sự của chúng.

Bảng 3.5 - So sánh kết quả chạy thử nghiệm với kết quả của các tác giả người Italy

Bài toán test	n	m	Kết quả tối ưu	Kết quả chạy các thuật toán				Thời gian chạy trung bình			
				GA-ACO	GA	ACO	NH GA	GA-ACO	GA	ACO	NH GA
LA01	10	5	666	666	675	669	666	183	143	171	100
LA02	10	5	655	688	712	693	655	221	125	322	120
LA03	10	5	597	626	644	642	597	290	125	497	150
LA04	10	5	590	611	628	625	590	312	139	313	200
LA07	15	5	890	894	939	908	890	110	92	71	80
LA08	15	5	863	863	872	865	863	13	42	63	20
LA15	20	5	1207	1246	1284	1249	1207	360	189	184	150

Để chứng tỏ tính vượt trội của thuật toán mà luận án đề xuất. Kết quả chạy thử nghiệm của luận án được so sánh với các kết quả chạy thử nghiệm các thuật toán GA-ACO, GA, ACO của các tác giả Andrea Rossi và Elena Boschi người Italy [5] được công bố năm 2010. Thuật toán do luận án đề xuất và các thuật toán của các tác giả người Italy đều được cài đặt, chạy trên máy PC tốc độ 2.8 GHz và hệ

điều hành Windows. Các bài toán thử nghiệm được chọn trong bộ test của Lawrence.

Trong bảng thống kê so sánh có hai phần: phần kết quả tính toán và phần thời gian chạy máy trung bình cho kết quả tính toán. Trong các thuật toán được đề nghị của các tác giả người Italy, thuật toán GA-ACO là tốt nhất cả về kết quả tính toán lẫn thời gian chạy máy. Thuật toán NHGA do luận án đề xuất được so sánh với thuật toán GA-ACO. Bảng so sánh cho thấy kết quả tính toán của NHGA tốt hơn của GA-ACO, đồng thời thời gian chạy máy cũng nhanh hơn.

CHƯƠNG 4. PHÂN TÍCH TÍNH HỘI TỤ CỦA THUẬT TOÁN DI TRUYỀN LAI MỚI CHO BÀI TOÁN LẬP LỊCH JOB SHOP

Các giải pháp gần đúng được đề xuất trong những năm qua chỉ được đánh giá thông qua kết quả thử nghiệm. Tính hội tụ tới tối ưu toàn cục của các giải pháp mới không được chứng minh dựa trên cơ sở lý thuyết. Trong chương này, luận án phân tích các thuộc tính hội tụ của thuật toán do luận án đề nghị trong chương 3 bằng cách áp dụng các tính chất của xích Markov. Trên cơ sở phân tích xích Markov của thuật toán di truyền, luận án chứng tỏ rằng thuật toán được đề nghị trong chương 3 hội tụ tới tối ưu toàn cục.

4.1. Lý thuyết Xích Markov

Nếu một tiến trình mà sự tiến triển của nó trong tương lai chỉ phụ thuộc vào hiện tại và hoàn toàn độc lập với quá khứ (tính không nhớ), thì nó có tính chất Markov. Một quá trình ngẫu nhiên $X(t)$ có tính chất Markov được gọi là quá trình Markov. Nếu không gian trạng thái S gồm một số hữu hạn hoặc vô hạn đếm được các trạng thái thì quá trình Markov $X(t)$ được gọi là một xích Markov.

4.2. Xích Markov Ergodic

Một xích Markov được gọi là xích Markov Ergodic nếu từ một trạng thái gốc bất kỳ, ta có thể di chuyển đến mọi trạng thái khác trong không gian trạng thái (không nhất thiết phải sau 1 bước). Như vậy, một xích Markov có ma trận chuyển là ma trận chính quy (primitive) là xích Markov Ergodic.

4.3. Phân tích tính hội tụ của thuật toán di truyền lai tuần tự cho bài toán lập lịch job shop

- **Phân tích tính hội tụ của thuật toán tiến hóa truyền thống**

Thuật toán tiến hóa truyền thống có thể được mô tả thông qua một xích Markov với trạng thái của quần thể nằm trong không gian trạng thái và ma trận xác suất chuyển trạng thái P dương. Ta suy ra thuật toán này chính là một xích Markov Ergodic. Qua phân tích sử dụng lý thuyết xích Markov đã chứng tỏ rằng thuật toán di truyền truyền thống không hội tụ tới tối ưu toàn cục (tr. 117-123).

- **Phân tích tính hội tụ của thuật toán di truyền với cá thể tinh hoa và toán tử sao chép**

Thuật toán di truyền với cá thể tinh hoa và toán tử sao chép được đề xuất trong mục 3.3. Trên cơ sở phân tích xích Markov của thuật toán di truyền với cá thể tinh hoa và toán tử sao chép, luận án đã chứng tỏ rằng thuật toán được đề nghị trong chương 3 hội tụ tới tối ưu toàn cục.

KẾT LUẬN

Trong thời gian qua, với sự nỗ lực của bản thân và được sự hướng dẫn tận tình của hai cán bộ hướng dẫn, luận án đã hoàn thành các mục tiêu đặt ra ban đầu. Các kết quả cụ thể mà luận án đạt được như sau:

1. Nghiên cứu tổng quan về bài toán lập lịch job shop: Phân tích đánh giá, so sánh các giải pháp đã áp dụng cho các bài toán lập lịch job shop. Trên cơ sở đó đề xuất một số hướng nghiên cứu để giải quyết bài toán này.

2. Đề xuất một thuật toán di truyền lai mới kết hợp thuật toán di truyền với các kỹ thuật tìm kiếm khác cho bài toán lập lịch job shop. Trong phương pháp đề xuất này, có một số cải tiến trong các công đoạn: Mã hóa lời giải, toán tử đột biến và toán tử trao đổi chéo. Phương pháp đề xuất này đã được cài đặt và chạy thử nghiệm trên các bài toán chuẩn cho kết quả tốt. Kết quả đã được so sánh kết quả với các giải pháp trước đó để chứng tỏ tính vượt trội của nó.

3. Đề xuất một thuật toán di truyền lai song song cho bài toán lập lịch job shop, thuật toán đã được cài đặt và chạy thử nghiệm cho kết quả tốt và rút ngắn được nhiều lần thời gian thực thi với cùng bộ tham số và dữ liệu vào trong thuật toán tuần tự. Kết quả này đã được chuyển thành bài báo tham gia hội nghị quốc tế về “xử lý tín hiệu số và công nghệ thông tin - ISSPIT” 2012.

4. Chứng minh tính hội tụ tới tối ưu toàn cục của thuật toán di truyền lai mới với mã hóa tự nhiên cho bài toán lập lịch job shop. Kết quả này đã chuyển thành bài báo tham gia hội nghị quốc tế về “xử lý tín hiệu số và công nghệ thông tin - ISSPIT” 2012.

DANH MỤC CÔNG TRÌNH KHOA HỌC CỦA TÁC GIẢ LIÊN QUAN ĐẾN LUẬN ÁN

1. Nguyễn Hữu Mùi, Vũ Đình Hoà (2009), "Solving the permutation flow shop scheduling problem by genetic algorithms", *Journal of Science of HNUE* Vol. 54 (1), pp. 40-45.

2. Nguyễn Hữu Mùi, Vũ Đình Hoà (2009), "Solving the flow shop scheduling problem by genetic algorithms", *Journal of Science of HNUE* Vol. 54 (6), pp. 35-41.

3. Nguyễn Hữu Mùi, Vũ Đình Hoà (2010), "Active schedules and a new hybrid genetic algorithm for the job shop scheduling problem", *VNU Journal of Science, Mathematics - Physics* Vol. 26 (4), pp. 213-221.

4. Nguyễn Hữu Mùi, Vũ Đình Hoà (2010), "Solving the job shop scheduling problem by genetic algorithm", *Addendum Proceedings IEEE RIVF 2010*, pp. 29-32.

5. Nguyễn Hữu Mùi, Vũ Đình Hoà (2011), "Giải bài toán lập lịch job shop bằng thuật toán di truyền", *Kỷ yếu hội thảo quốc gia lần thứ XIII, một số vấn đề chọn lọc của công nghệ thông tin và truyền thông*, tr. 71-82.

6. Nguyễn Hữu Mùi, Vũ Đình Hoà (2011), "Một thuật toán di truyền lai mới cho bài toán lập lịch công việc", *Kỷ yếu hội nghị khoa học công nghệ quốc gia lần thứ V*, tr. 239-249.

7. Nguyễn Hữu Mùi, Vũ Đình Hoà (2012), "Một thuật toán di truyền hiệu quả cho bài toán lập lịch job shop", *Tạp chí khoa học và công nghệ, Viện Khoa học và Công nghệ Việt nam* Tập 50 (5), tr. 565-577.

8. Nguyễn Hữu Mùi - Vũ Đình Hoà - Lục Trí Tuyên (2012), "A Parallel Genetic Algorithm for the Job Shop Scheduling Problem", Proceedings IEEE ISSPIT 2012, Published online.

9. Nguyễn Hữu Mùi - Vũ Đình Hoà - Lục Trí Tuyên (2012), "Convergence Analysis of the New Hybrid Genetic Algorithm for the Job Shop Scheduling Problem", Proceedings IEEE ISSPIT 2012, Published online.