

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



Nguyễn Hồng Anh Tấn

**ỨNG DỤNG TRÍ TUỆ NHÂN TẠO HỖ TRỢ SẮP XẾP LỊCH HỌC
TẠI HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT
(Theo định hướng ứng dụng)

HÀ NỘI - 2024

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



Nguyễn Hồng Anh Tấn

**ỨNG DỤNG TRÍ TUỆ NHÂN TẠO HỖ TRỢ SẮP XẾP LỊCH HỌC
TẠI HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

Chuyên ngành: HỆ THỐNG THÔNG TIN

Mã số: 8.48.01.04

ĐỀ CƯƠNG ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. PHAN LÝ HUỖNH

HÀ NỘI - 2024

LỜI CAM ĐOAN

Học viên Nguyễn Hồng Anh Tấn, mã học viên B22CHIS039 xin cam đoan đề án tốt nghiệp là công trình nghiên cứu của riêng học viên dưới sự hướng dẫn của TS. Phan Lý Huỳnh. Tất cả những tham khảo trong đề án tốt nghiệp bao gồm hình ảnh, bảng biểu, số liệu, và các câu từ trích dẫn – đều được ghi rõ ràng và đầy đủ nguồn gốc trong danh mục tài liệu tham khảo.

Hà Nội, ngày 01 tháng 05 năm 2024

Tác giả đề án tốt nghiệp ký và ghi rõ họ tên

Nguyễn Hồng Anh Tấn - B22CHIS039

LỜI CẢM ƠN

Em xin chân thành cảm ơn giảng viên hướng dẫn TS. Phan Lý Huỳnh đã giúp đỡ và định hướng cho em trong toàn bộ quá trình học tập và thực hiện đề án tốt nghiệp.

Dưới sự hướng dẫn của TS. Phan Lý Huỳnh, em đã cố gắng hoàn thành tốt nhất có thể đề án tốt nghiệp này. tuy nhiên trong quá trình thực hiện không thể tránh được những thiếu sót, em rất mong nhận được sự góp ý của các thầy/cô trong hội đồng để em hoàn thiện hơn đề án tốt nghiệp này.

Em xin chân thành cảm ơn!

Hà Nội, ngày 01 tháng 05 năm 2024

Học viên thực hiện

Nguyễn Hồng Anh Tấn - B22CHIS039

MỤC LỤC

| | |
|--|------|
| LỜI CAM ĐOAN..... | I |
| LỜI CẢM ƠN..... | II |
| DANH MỤC VIẾT TẮT..... | VII |
| DANH SÁCH BẢNG..... | VIII |
| DANH SÁCH HÌNH VẼ..... | IX |
| MỞ ĐẦU..... | 1 |
| CHƯƠNG 1. TỔNG QUAN VỀ BÀI TOÁN LẬP LỊCH HỌC VÀ CÁC PHƯƠNG PHÁP TIẾP CẬN..... | 3 |
| 1.1 Đặt vấn đề..... | 3 |
| 1.2 Mô tả bài toán..... | 3 |
| 1.3 Các phương pháp tiếp cận..... | 5 |
| 1.3.1 Giải thuật tham lam (Greedy Algorithm)..... | 5 |
| 1.3.2 Giải thuật leo đồi (Hill Climbing Algorithm)..... | 6 |
| 1.3.3 Giải thuật luyện kim (Simulated Annealing - SA)..... | 6 |
| 1.3.4 Giải thuật tối ưu đàn kiến (Ant Colony Optimization - ACO)..... | 7 |
| 1.3.5 Giải thuật di truyền (Genetic Algorithm - GA)..... | 8 |
| 1.4 Kết luận..... | 9 |
| CHƯƠNG 2. NGHIÊN CỨU ĐÁNH GIÁ GIẢI THUẬT DI TRUYỀN..... | 11 |

| | |
|---|----|
| 2.1 Khái niệm..... | 11 |
| 2.1.1 Khởi tạo quần thể..... | 12 |
| 2.1.2 Chọn lọc (selection)..... | 12 |
| 2.1.3 Lai ghép (crossover)..... | 14 |
| 2.1.4 Đột biến (mutation)..... | 15 |
| 2.2 Xây dựng mô hình..... | 18 |
| 2.2.1 Ràng buộc cứng..... | 19 |
| 2.2.2 Ràng buộc mềm..... | 20 |
| 2.2.3 Hàm thích nghi..... | 22 |
| 2.2.4 Biểu diễn nhiễm sắc thể..... | 22 |
| 2.2.5 Khởi tạo quần thể..... | 23 |
| 2.2.6 Toán tử chọn lọc..... | 24 |
| 2.2.7 Toán tử đột biến..... | 25 |
| 2.2.8 Cải tiến..... | 25 |
| 2.2.8.1 Cải tiến lai ghép..... | 25 |
| 2.2.8.2 Cải tiến đột biến..... | 26 |
| 2.3 Đánh giá..... | 27 |
| 2.4 Kết luận..... | 29 |
| CHƯƠNG 3. XÂY DỰNG ỨNG DỤNG XẾP THỜI KHÓA BIỂU BÁN TỰ ĐỘNG..... | 30 |

| | |
|---|----|
| 3.1 Phân tích thiết kế..... | 30 |
| 3.1.1 Danh mục Use-case..... | 30 |
| 3.2 Kiến trúc tổng quan..... | 34 |
| 3.3 Công nghệ..... | 35 |
| 3.3.1 ReactJS (Client)..... | 35 |
| 3.3.2 NestJS (Backend)..... | 36 |
| 3.3.3 Keycloak (IAM)..... | 37 |
| 3.4 Chức năng chính..... | 38 |
| 3.4.1 Rà soát nhu cầu học phần..... | 39 |
| 3.4.1.1 Chương trình đào tạo..... | 39 |
| 3.4.1.2 Kết quả học tập học phần..... | 40 |
| 3.4.1.3 Rà soát nhu cầu..... | 41 |
| 3.4.2 Phân công giảng dạy..... | 43 |
| 3.4.3 Thiết lập danh mục phòng học..... | 44 |
| 3.4.4 Thiết lập danh mục môn học..... | 45 |
| 3.4.5 Xếp thời khóa biểu bán tự động..... | 46 |
| 3.4.5.1 Cấu hình tham số giải thuật..... | 46 |
| 3.4.5.2 Xếp thời khóa biểu tự động..... | 47 |
| 3.5 Kết luận..... | 50 |
| KẾT LUẬN..... | 51 |

| | |
|-------------------------|----|
| TÀI LIỆU THAM KHẢO..... | 52 |
|-------------------------|----|

DANH MỤC VIẾT TẮT

| Viết tắt | Tiếng Anh | Tiếng Việt |
|------------|-------------------------|----------------------|
| <i>SA</i> | Simulated Annealing | Giải thuật luyện kim |
| <i>ACO</i> | Ant Colony Optimization | Tối ưu đàn kiến |
| <i>GA</i> | Genetic Algorithm | Giải thuật di truyền |
| <i>NST</i> | | Nhiễm sắc thể |
| <i>AI</i> | Artificial intelligence | Trí tuệ nhân tạo |
| <i>ML</i> | Machine learning | Học máy |

DANH SÁCH BẢNG

| | |
|---|----|
| Bảng 1.1: Bảng so sánh các giải thuật xếp lịch..... | 9 |
| Bảng 2.1: Thông tin chi tiết của các nhiệm sắc thể..... | 13 |
| Bảng 2.2: Bảng trọng số ràng buộc mềm..... | 28 |
| Bảng 2.3: Bảng đánh giá hiệu quả giải thuật..... | 28 |
| Bảng 3.1: Danh mục Use-case..... | 30 |
| Bảng 3.2: Bảng công việc xếp thời khóa biểu..... | 38 |

DANH SÁCH HÌNH VẼ

| | |
|--|----|
| Hình 2.1 Cách thức hoạt động của bánh xe roulette..... | 13 |
| Hình 2.2: Hai kỹ thuật lai ghép phổ biến của giải thuật di truyền:..... | 15 |
| Hình 2.3: Toán tử đột biến thay đổi một hoặc nhiều gen của các nhiễm sắc thể con sau khi quá trình lai ghép diễn ra..... | 16 |
| Hình 2.4: Sơ đồ tổng quan của giải thuật di truyền..... | 17 |
| Hình 2.5: Biểu diễn nhiễm sắc thể dưới dạng mảng ba chiều..... | 23 |
| Hình 2.6: Điền lần lượt các lớp học vào các tiết ngẫu nhiên..... | 24 |
| Hình 3.1: Mô hình hoạt động của hệ thống..... | 35 |
| Hình 3.2: Chương trình đào tạo khóa ngành..... | 40 |
| Hình 3.3: Điểm học phần của sinh viên..... | 41 |
| Hình 3.4: Cấu hình thông tin rà soát học phần..... | 42 |
| Hình 3.5: Bảng thống kê nhu cầu học phần..... | 43 |
| Hình 3.6: Chức năng phân công giảng viên của khoa bộ môn..... | 44 |
| Hình 3.7: Thông tin phòng học..... | 45 |
| Hình 3.8: Các thông tin của môn học..... | 46 |
| Hình 3.9: Cấu hình tham số giải thuật di truyền..... | 47 |
| Hình 3.10: Dashboard xếp thời khóa biểu..... | 48 |
| Hình 3.11: Danh sách các tuần học trong kỳ..... | 48 |
| Hình 3.12: Danh sách các lớp tín chỉ chưa được xếp đủ lịch học..... | 49 |

| | |
|---|----|
| Hình 3.13: Giao diện xếp thời khóa biểu thủ công..... | 49 |
| Hình 3.14: Các thao tác xếp lịch..... | 50 |

MỞ ĐẦU

Bài toán xếp thời khoá biểu là một bài toán phổ biến trong các tổ chức giáo dục nói chung và các trường Đại học nói riêng. Thời khoá biểu có thể bao gồm lịch giảng dạy, đào tạo bồi dưỡng của giảng viên; lịch học, lịch thực hành, lịch thi của sinh viên, các sự kiện trong trường. Mục đích của việc xếp thời khoá biểu là xây dựng ra một kế hoạch học tập được sắp xếp một cách hợp lý, không gây xung đột, và đảm bảo nhiều yếu tố khác như cân bằng về thời gian giảng dạy/học tập của giảng viên/sinh viên.

Bài toán xếp thời khoá biểu ràng buộc bởi nhiều yếu tố, bao gồm các ràng buộc của nhà quản trị, của người dạy, người học, ... và chúng thường xung đột với nhau. Điều đó khiến việc giải quyết bài toán khá phức tạp và tốn nhiều thời gian. Vì vậy các giải pháp sắp xếp thời khoá biểu tự động hoặc bán tự động là thiết thực và giúp giảm thiểu công sức của cán bộ, chuyên viên tại các tổ chức giáo dục trong việc xây dựng thời khoá biểu cho các học kỳ.

Có nhiều phương pháp đã được nghiên cứu để giải quyết bài toán xếp thời khoá biểu. Trong đó các phương pháp ứng dụng trí tuệ nhân tạo có thể được xem xét để nghiên cứu do các hiệu quả của chúng so với các phương pháp khác.

Bên cạnh đó, sắp xếp thời khoá biểu tự động/bán tự động cũng là một chức năng quan trọng trong các hệ thống quản lý đào tạo, trong đó có hệ thống quản lý đào tạo số đang được phát triển tại Học viện Công nghệ Bưu chính Viễn thông. Vì vậy đề tài “**Ứng dụng trí tuệ nhân tạo hỗ trợ sắp xếp lịch học tại Học viện Công nghệ Bưu chính Viễn thông**” được xây dựng với mục đích nghiên cứu và phát triển giải pháp xây dựng thời khoá biểu tự động/bán tự động có hiệu quả cao cho hệ thống quản lý đào tạo số của Học viện.

Có nhiều phương pháp đã được nghiên cứu để giải quyết bài toán xếp thời khoá biểu như các phương pháp tối ưu toán học, giải thuật metaheuristic, giải thuật

tô màu đồ thị... [1]. Trong số đó, giải thuật di chuyển là một kỹ thuật tìm kiếm metaheuristic được mô phỏng theo quá trình chọn lọc tự nhiên. Ưu điểm của giải thuật di chuyển bao gồm khả năng tìm kiếm trên không gian tìm kiếm lớn, khả năng xử lý song song, khả năng tối ưu hoá toàn cục [2]. Bên cạnh đó, giải thuật này cũng có một số nhược điểm như độ phức tạp lớn trong việc tính toán, khả năng xảy ra hội tụ sớm, khó tìm kiếm chính xác tối ưu toàn cục [2]. Các biến thể của giải thuật di chuyển đã được nghiên cứu để khắc phục các nhược điểm trên [3]. Một biến thể của giải thuật với các tiêu chí phù hợp sẽ được lựa chọn để áp dụng giải quyết bài toán xếp thời khoá biểu.

Đề án nghiên cứu có mục tiêu tìm ra giải pháp sắp xếp thời khoá biểu ứng dụng trí tuệ nhân tạo, từ đó phát triển ứng dụng sắp xếp thời khoá biểu tự động/bán tự động.

Đề án nghiên cứu giải pháp sắp xếp thời khoá biểu dựa trên giải thuật di chuyển. Bên cạnh đó, đề án nghiên cứu biến thể của giải thuật di chuyển có khả năng tối ưu cho bài toán xếp thời khoá biểu. Mô hình thời khoá biểu được xây dựng dựa trên thời khoá biểu giảng dạy theo học chế tín chỉ tại Học viện Công nghệ Bưu chính Viễn thông. Kết quả nghiên cứu sẽ được áp dụng trong việc xây dựng mô hình sắp xếp thời khoá biểu tự động/bán tự động trong hệ thống quản lý đào tạo số đang được phát triển tại Học viện.

CHƯƠNG 1. TỔNG QUAN VỀ BÀI TOÁN LẬP LỊCH HỌC VÀ CÁC PHƯƠNG PHÁP TIẾP CẬN

1.1 Đặt vấn đề

Ngày nay các trường Đại học thường tổ chức đào tạo theo hệ đào tạo tín chỉ, thay thế cho hệ đào tạo niên chế. Trong khi hệ đào tạo niên chế quản lý sinh viên tập trung theo lớp hành chính và dựa trên các lớp hành chính để xây dựng lịch học, hệ đào tạo tín chỉ lại coi sinh viên là đơn vị quản lý cơ sở và lập lịch học kết hợp giữa nhu cầu và chương trình đào tạo của sinh viên. Do đó việc sắp xếp lịch học đối với hệ thống đào tạo tín chỉ có nhiều ràng buộc hơn. Bên cạnh đó, lịch học có thể thay đổi liên tục nhưng vẫn phải đảm bảo các điều kiện ràng buộc về cơ sở vật chất, quỹ thời gian của người dạy và người học. Những điều trên khiến cho việc xếp lịch học là một bài toán phức tạp và tốn nhiều thời gian.

Vì vậy, việc phát triển một công cụ xây dựng lịch học theo hệ đào tạo tín là điều cần thiết đối với công việc quản lý đào tạo của các trường Đại học.

1.2 Mô tả bài toán

Theo quy chế đào tạo của hệ đào tạo tín chỉ áp dụng cho trường Đại học, sinh viên được phép tự do lựa chọn môn học và thời gian học theo nhu cầu cá nhân của mình. Trước khi bắt đầu mỗi học kỳ, phòng đào tạo tiến hành xây dựng kế hoạch giảng dạy. Căn cứ để lập kế hoạch giảng dạy gồm:

- Chương trình đào tạo toàn khóa của từng chuyên ngành đã được Hiệu trưởng ký duyệt.
- Số lượng sinh viên thuộc các khóa - ngành tương ứng có đủ điều kiện theo học các môn học dự kiến giảng dạy trong năm học.

- Dự tính số lượng sinh viên sẽ học lại mỗi môn học và số lớp có thể mở tương ứng.
- Ý kiến phản hồi và yêu cầu từ các khoa và sinh viên.

Kế hoạch giảng dạy bao gồm :

- Chương trình đào tạo của từng khóa-ngành.
- Thông tin chi tiết về từng môn học.
- Số lớp dự kiến mở cho mỗi môn học trong học kỳ.
- Tiến độ học tập của năm học.

Kế hoạch này được gửi đến từng khoa và các đơn vị liên quan. Các khoa góp ý cho bản kế hoạch giảng dạy và thông qua hệ thống cố vấn học tập thu thập ý kiến từ sinh viên để bổ sung, điều chỉnh kế hoạch.

Dựa trên kế hoạch giảng dạy, các khoa phân công giảng viên phụ trách giảng dạy từng lớp tín chỉ, kể cả việc mời giảng khi cần thiết. Các giảng viên được phân công giảng dạy có thể nêu kèm theo các yêu cầu riêng, nếu có.

Sau khi nhận được danh sách phân công giảng viên và các ý kiến phản hồi từ các đơn vị, căn cứ vào kế hoạch giảng dạy, phòng Đào tạo sẽ tiến hành phân lớp, lập và công bố Thời khóa biểu. Thời khóa biểu này được công bố làm cơ sở để sinh viên đăng ký môn học.

Dựa vào quy mô của chương trình đào tạo, số lượng lớp tín chỉ trong một kỳ học có thể biến đổi từ vài trăm đến vài ngàn lớp. Việc tạo thời khóa biểu cho một lượng lớn các lớp tín chỉ có thể dẫn đến việc sinh viên gặp khó khăn khi đăng ký các môn học do trùng lịch học.

Phát biểu bài toán như sau:

Mỗi trường có một danh sách lớp học, danh sách phòng học, danh sách các tiết học trong một ngày, danh sách các thứ học trong một tuần.

Mỗi lớp học ứng với một môn học, được phân công cho giảng viên giảng dạy thuộc khoa, bộ môn quản lý môn học đó.

Mỗi lớp học có danh sách các buổi học căn cứ theo số tín chỉ của môn học và số tiết cần giảng dạy.

Cần tìm một phương án sắp xếp các buổi học của các lớp vào các phòng học tại các tiết học của một ngày trong tuần. Việc sắp xếp lịch học cần thỏa mãn các điều kiện bắt buộc (ràng buộc cứng) và tối ưu hóa tối đa các điều kiện không bắt buộc (ràng buộc mềm) [1].

1.3 Các phương pháp tiếp cận

Dưới đây là một số phương pháp được sử dụng để giải quyết bài toán xếp thời khóa biểu, dựa trên khảo sát của Tan, J. S. và các cộng sự [2]:

1.3.1 Giải thuật tham lam (Greedy Algorithm)

Giải thuật tham lam là một phương pháp giải quyết bài toán dựa trên việc lựa chọn bước đi tốt nhất tại mỗi bước mà không cần xem xét toàn bộ bài toán. Tư tưởng chính của giải thuật này là luôn chọn giải pháp tốt nhất hiện tại với hy vọng rằng nó sẽ dẫn đến kết quả tốt nhất toàn cục.

Một giải thuật tham lam điển hình có ba thành phần chính:

1. Lựa chọn tham lam: Ở mỗi bước, chọn phương án tốt nhất tại thời điểm đó.
2. Tính chất không đổi: Sau mỗi bước chọn, bài toán con mới vẫn giữ nguyên tính chất như bài toán ban đầu.
3. Giải quyết bài toán con: Tiếp tục áp dụng chiến lược tham lam để giải quyết bài toán con còn lại.

Giải thuật tham lam có thể áp dụng vào bài toán xếp thời khóa biểu bằng cách chọn các lớp học vào các khoảng thời gian sao cho tối ưu nhất tại mỗi bước. Chẳng hạn, nó có thể bắt đầu bằng việc sắp xếp các lớp học có số lượng sinh viên đông nhất vào các khung giờ có sẵn trước, sau đó tiếp tục với các lớp nhỏ hơn. Tuy nhiên, giải pháp này không phải lúc nào cũng tối ưu toàn cục vì nó không xem xét toàn bộ bài toán từ đầu đến cuối.

1.3.2 Giải thuật leo đồi (Hill Climbing Algorithm)

Giải thuật leo đồi là một phương pháp tìm kiếm dựa trên việc bắt đầu từ một điểm khởi đầu và liên tục di chuyển đến trạng thái láng giềng tốt hơn, cho đến khi không còn trạng thái láng giềng nào tốt hơn điểm hiện tại. Điều này tương tự như việc leo lên đỉnh đồi, luôn di chuyển theo hướng lên cao cho đến khi đạt đỉnh.

Quá trình này bao gồm:

1. Khởi tạo: Bắt đầu từ một trạng thái ban đầu ngẫu nhiên.
2. Đánh giá: Đánh giá các trạng thái láng giềng.
3. Lựa chọn: Chọn trạng thái láng giềng tốt nhất để di chuyển đến.
4. Lặp lại: Lặp lại quá trình cho đến khi không có cải thiện nào nữa.

Trong bài toán xếp thời khóa biểu, giải thuật leo đồi có thể bắt đầu với một phân bố thời khóa biểu ngẫu nhiên và sau đó cải thiện bằng cách hoán đổi các lớp học, sao cho giảm thiểu xung đột (ví dụ như tránh trùng lặp phòng học, thời gian, giảng viên). Tuy nhiên, phương pháp này dễ mắc kẹt tại các cực tiểu cục bộ và không đảm bảo tìm ra giải pháp tối ưu toàn cục.

1.3.3 Giải thuật luyện kim (Simulated Annealing - SA)

Giải thuật luyện kim dựa trên quá trình tôi luyện kim loại, trong đó kim loại được đun nóng đến nhiệt độ cao rồi làm nguội từ từ để đạt được cấu trúc tinh thể tối

ưu. Tương tự, SA bắt đầu từ một giải pháp ngẫu nhiên và thực hiện các thay đổi ngẫu nhiên với xác suất chấp nhận phụ thuộc vào nhiệt độ, từ đó giảm dần nhiệt độ theo thời gian.

Các bước chính của SA bao gồm:

1. Khởi tạo: Bắt đầu từ một giải pháp ngẫu nhiên.
2. Đột biến và chấp nhận: Thực hiện thay đổi ngẫu nhiên giải pháp hiện tại và quyết định chấp nhận giải pháp mới dựa trên xác suất tính theo hàm nhiệt độ.
3. Giảm nhiệt độ: Giảm nhiệt độ từ từ theo một lịch trình làm nguội.
4. Lặp lại: Lặp lại quá trình cho đến khi nhiệt độ giảm về 0 hoặc đạt đến số lần lặp nhất định.

SA có thể được sử dụng để xếp thời khóa biểu bằng cách bắt đầu từ một giải pháp ban đầu và thực hiện các thay đổi nhỏ, chẳng hạn như hoán đổi thời gian của các lớp học. Với xác suất chấp nhận các giải pháp tạm thời kém hơn, SA có thể vượt qua các cực tiểu cục bộ và tìm kiếm giải pháp tối ưu hơn, làm cho nó phù hợp với các bài toán có không gian tìm kiếm phức tạp như xếp thời khóa biểu.

1.3.4 Giải thuật tối ưu đàn kiến (Ant Colony Optimization - ACO)

Giải thuật tối ưu đàn kiến là một phương pháp dựa trên hành vi tìm đường của các con kiến trong tự nhiên. Các con kiến khi di chuyển để tìm thức ăn sẽ để lại dấu vết pheromone, và các kiến khác có xu hướng đi theo những dấu vết này, dẫn đến việc tìm ra con đường ngắn nhất một cách tự nhiên.

Các bước chính trong ACO bao gồm:

1. Khởi tạo: Đặt các kiến tại các điểm khởi đầu khác nhau.
2. Di chuyển và đặt pheromone: Các kiến di chuyển đến các điểm khác nhau dựa trên một xác suất tỉ lệ thuận với lượng pheromone trên đường đi.

3. Cập nhật pheromone: Lượng pheromone trên các đường đi được cập nhật dựa trên chất lượng của giải pháp mà các kiến tìm được.
4. Lặp lại: Lặp lại quá trình cho đến khi đạt được kết quả tối ưu.

ACO có thể được áp dụng để tìm kiếm các phân bố thời khóa biểu tối ưu bằng cách mô phỏng các con kiến di chuyển giữa các lựa chọn lớp học và thời gian. Mỗi kiến sẽ thử một giải pháp xếp thời khóa biểu và để lại dấu vết pheromone, từ đó các kiến khác sẽ dần dần tìm ra các giải pháp tối ưu hơn dựa trên dấu vết này. Điều này giúp tìm ra các giải pháp tốt một cách hiệu quả.

1.3.5 Giải thuật di truyền (Genetic Algorithm - GA)

Giải thuật di truyền mô phỏng quá trình chọn lọc tự nhiên, sử dụng các khái niệm như gen, lai ghép, và đột biến để phát triển các giải pháp tốt hơn qua các thế hệ.

Các bước chính của GA bao gồm:

1. Khởi tạo quần thể: Tạo ra một quần thể các cá thể ngẫu nhiên.
2. Đánh giá: Đánh giá độ thích nghi của mỗi cá thể trong quần thể.
3. Chọn lọc: Chọn các cá thể tốt nhất để làm cha mẹ dựa trên độ thích nghi.
4. Lai ghép và đột biến: Lai ghép các cá thể cha mẹ để tạo ra thế hệ con mới, và thực hiện đột biến ngẫu nhiên.
5. Lặp lại: Lặp lại quá trình cho đến khi đạt được giải pháp tối ưu hoặc sau một số thế hệ nhất định.

GA có thể áp dụng bằng cách mã hóa mỗi thời khóa biểu dưới dạng một cá thể trong quần thể. Các cá thể này sẽ trải qua các quá trình chọn lọc, lai ghép và đột biến để tạo ra các thời khóa biểu mới, tốt hơn. GA có thể tìm ra các giải pháp tối ưu bằng cách kết hợp các đặc điểm tốt từ các cá thể khác nhau qua nhiều thế hệ.

Bảng 1: Bảng so sánh các giải thuật xếp lịch

| Giải thuật | Ưu điểm | Nhược điểm |
|-------------------|--|---|
| Tham lam | Đơn giản, hiệu quả cho bài toán nhỏ | Chất lượng giải pháp thấp |
| Leo đồi | Tìm kiếm giải pháp tốt hơn giải thuật tham lam | Dễ mắc kẹt tối ưu cục bộ, tốn thời gian tính toán |
| Luyện kim | Thoát tối ưu cục bộ tốt, tìm kiếm tối ưu cho bài toán nhỏ | Phức tạp, phụ thuộc vào tham số, tốn thời gian tính toán |
| ACO | Khả năng tìm kiếm giải pháp tốt, tính linh hoạt cao | Khó khăn mô hình hóa, yêu cầu dữ liệu, kết quả không tối ưu |
| Di truyền | Tối ưu hóa cao, hiệu quả, linh hoạt, xử lý thông tin phi số, dễ triển khai | Có thể mắc kẹt tối ưu cục bộ, phụ thuộc cài đặt |

1.4 Kết luận

Giải thuật Tham lam là giải thuật đơn giản nhất và dễ triển khai nhất, nhưng đối với các bài toán phức tạp như xếp thời khóa biểu, giải thuật Tham lam dễ bị rơi vào trường hợp tối ưu cục bộ do chỉ quan tâm đến kết quả có lợi trước mắt, dẫn đến không đưa ra được phương án tối ưu toàn cục.

Tương tự, giải thuật Leo đồi cũng dễ mắc phải trường hợp tối ưu cục bộ. Ngoài ra giải thuật leo đồi có tốc độ hội tụ chậm do quá trình “leo đồi” không thể nhảy tới các khu vực tìm kiếm xa hơn mà không cần phải đi qua các giải pháp tệ hơn.

Giải thuật Luyện kim tuy có thể thoát được tối ưu cục bộ, nhưng độ tối ưu của kết quả bị phụ thuộc vào các tham số đầu vào, và thời gian tính toán cao.

Giải thuật Tối ưu đàn kiến và giải thuật Di truyền là hai giải thuật mới và có nhiều tiến bộ trong việc tìm kiếm tối ưu và phù hợp với các bài toán có độ phức tạp cao giống như bài toán xếp thời khóa biểu. Giải thuật Di truyền dễ dàng hơn trong việc mô hình hóa và cài đặt so với giải thuật Tối ưu đàn kiến và dễ kết hợp với các giải thuật khác trong việc tìm kiếm tối ưu hơn. Do đó giải thuật Di truyền được chọn để nghiên cứu áp dụng giải quyết bài toán xếp thời khóa biểu.

CHƯƠNG 2. NGHIÊN CỨU ĐÁNH GIÁ GIẢI THUẬT DI TRUYỀN

Chương này sẽ trình bày các khái niệm của giải thuật di truyền và cách xây dựng mô hình giải thuật để giải quyết bài toán xếp thời khóa biểu.

2.1 Khái niệm

Giải thuật di truyền được lấy cảm hứng từ lý thuyết tiến hóa của Darwin [3] [4], trong đó sự sống sót của sinh vật có khả năng thích nghi cao hơn và gen của chúng được mô phỏng lại. Giải thuật di truyền là một thuật toán dựa trên quần thể (population-based). Mỗi giải pháp tương ứng với một nhiễm sắc thể (chromosome) và mỗi tham số đại diện cho một gen. Giải thuật di truyền đánh giá mức độ thích hợp của từng cá thể trong quần thể bằng cách sử dụng hàm thích nghi (mục tiêu). Để cải thiện các giải pháp kém, các giải pháp tốt nhất được chọn ngẫu nhiên bằng toán tử chọn lọc (ví dụ: bánh xe roulette). Toán tử này có nhiều khả năng chọn giải pháp tốt nhất hơn vì xác suất tỷ lệ thuận với mức độ phù hợp. Điều làm tăng khả năng tránh tối ưu cục bộ là xác suất chọn được các giải pháp kém. Điều này có nghĩa là nếu các giải pháp tốt bị mắc kẹt trong một giải pháp cục bộ, chúng có thể được rút ra bằng các giải pháp khác [5].

Giải thuật di truyền có tính chất ngẫu nhiên, vì vậy độ tin cậy của nó có thể bị nghi ngờ. Điều làm cho thuật toán này trở nên đáng tin cậy và có khả năng ước tính mức tối ưu toàn cục cho một bài toán là quá trình duy trì các giải pháp tốt nhất trong mỗi thế hệ và sử dụng chúng để cải thiện các giải pháp khác. Như vậy, toàn bộ quần thể sẽ trở nên tốt hơn qua từng thế hệ. Sự lai chéo (crossover) giữa các nhiễm sắc thể giúp trao đổi thông tin giữa các giải pháp với nhau mà giải pháp còn lại không có [5].

Giải thuật di truyền cũng được hưởng lợi từ toán tử đột biến (mutation). Toán tử này thay đổi ngẫu nhiên các gen trong nhiễm sắc thể, giúp duy trì sự đa dạng của các cá thể trong quần thể và làm mở rộng không gian tìm kiếm của giải thuật di truyền. Giống như trong tự nhiên, toán tử đột biến có thể mang lại một giải pháp tốt hơn đáng kể và dẫn các giải pháp khác hướng tới mức tối ưu toàn cục [5].

2.1.1 Khởi tạo quần thể

Giải thuật di truyền bắt đầu với một quần thể ngẫu nhiên. Quần thể này có thể được tạo ra từ phân bố ngẫu nhiên Gaussian để tăng tính đa dạng. Quần thể này bao gồm nhiều giải pháp đại diện cho nhiễm sắc thể của các cá thể. Mỗi nhiễm sắc thể có một tập hợp các biến mô phỏng các gen. Mục tiêu chính trong bước khởi tạo là phổ biến các giải pháp xung quanh không gian tìm kiếm một cách đồng nhất nhất có thể để tăng tính đa dạng của dân số và có cơ hội tốt hơn để tìm thấy các khu vực đầy hứa hẹn.

2.1.2 Chọn lọc (selection)

Chọn lọc tự nhiên là nguồn cảm hứng chính cho toán tử chọn lọc của giải thuật di truyền. Trong tự nhiên, những cá thể khỏe mạnh nhất có cơ hội kiếm được thức ăn và giao phối cao hơn. Điều này khiến gen của chúng đóng góp nhiều hơn vào việc tạo ra thế hệ tiếp theo của cùng loài. Từ ý tưởng đơn giản này, giải thuật di truyền sử dụng bánh xe roulette để gán xác suất cho các cá nhân và chọn họ để tạo ra thế hệ tiếp theo tỷ lệ thuận với các giá trị thích hợp (khách quan) của họ. Hình 2.1 minh họa một ví dụ về bánh xe roulette dành cho năm nhiễm sắc thể. Thông tin chi tiết về những nhiễm sắc thể này được trình bày trong bảng 2.1.

Hình 1 Cách thức hoạt động của bánh xe roulette**Bảng 2: Thông tin chi tiết của các nhiễm sắc thể**

| Nhiễm sắc thể | Giá trị thích nghi | % trên tổng |
|---------------|--------------------|-------------|
| 1 | 0.01 | 1 |
| 2 | 0.12 | 15 |
| 3 | 0.25 | 31 |
| 4 | 0.07 | 9 |
| 5 | 0.36 | 44 |
| Tổng | 0.81 | 100 |

Có thể thấy rằng nhiễm sắc thể tốt nhất (#5) có phần chia lớn nhất trong bánh xe roulette, trong khi nhiễm sắc thể kém nhất (#1) có phần chia thấp nhất. Cơ chế này mô phỏng quá trình chọn lọc tự nhiên của cá thể khỏe mạnh nhất trong tự nhiên. Vì bánh xe roulette là một toán tử ngẫu nhiên nên những nhiễm sắc thể kém có ít khả năng tham gia vào việc sản sinh ra thế hệ tiếp theo. Tuy nhiên việc loại bỏ các

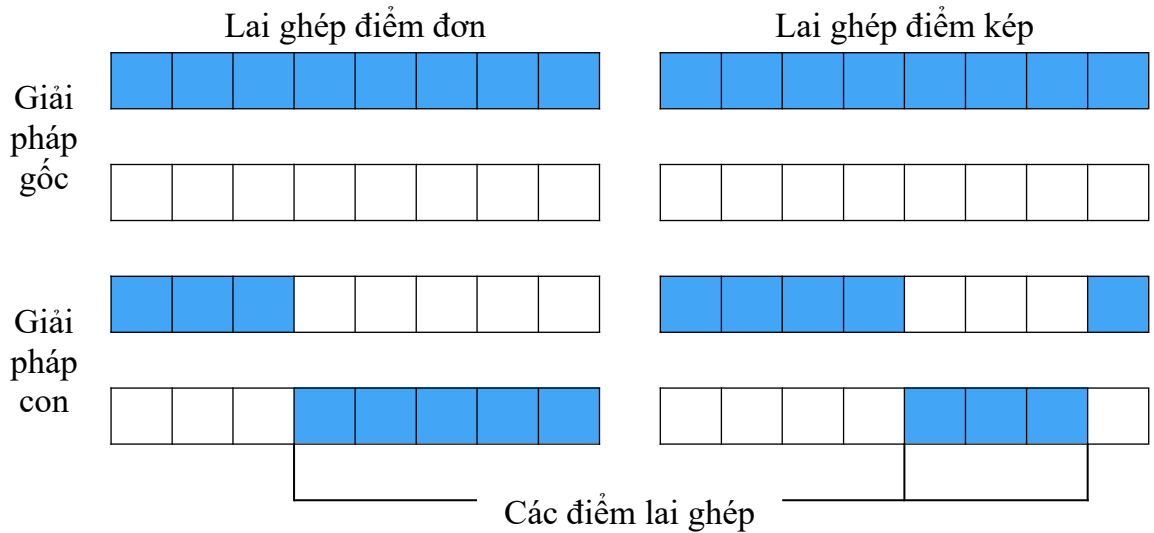
nhiễm sắc thể có độ thích nghi không tốt đôi khi có thể làm giảm sự đa dạng của quần thể.

Bánh xe roulette là một trong nhiều toán tử chọn lọc trong tài liệu [6] [7] [8]. Một số toán tử chọn lọc khác có thể kể đến như:

- Chọn lọc Boltzmann (Boltzmann selection) [9]
- Chọn lọc cạnh tranh (Tournament selection) [10]
- Chọn lọc thứ hạng (Rank selection) [11]
- Chọn lọc trạng thái ổn định (Steady state selection) [12]
- Chọn lọc cắt ngắn (Truncation selection) [13]

2.1.3 Lai ghép (crossover)

Sau khi chọn các cá thể bằng toán tử chọn lọc, chúng sẽ được sử dụng để tạo ra thế hệ nhiễm sắc thể mới. Trong tự nhiên, các nhiễm sắc thể trong gen của nam và nữ được kết hợp với nhau để tạo ra nhiễm sắc thể mới. Điều này được mô phỏng bằng cách kết hợp hai giải pháp (giải pháp gốc) được chọn bởi bánh xe roulette để tạo ra hai giải pháp mới (giải pháp con) trong giải thuật di truyền. Có nhiều kỹ thuật khác nhau cho toán tử lai ghép, trong đó có hai kỹ thuật (điểm đơn và điểm kép [16]) được hiển thị trong Hình 2.2.



Hình 2: Hai kỹ thuật lai ghép phổ biến của giải thuật di truyền:

lai ghép điểm đơn và lai ghép điểm kép

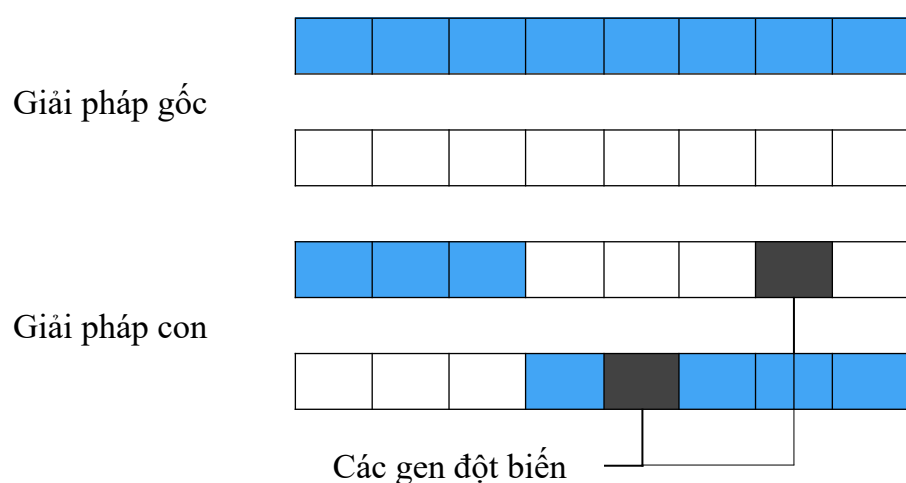
Trong phép lai ghép điểm đơn, nhiễm sắc thể của hai giải pháp gốc được hoán đổi trước và sau một điểm. Còn trong lai ghép điểm kép, nhiễm sắc thể sẽ được hoán đổi giữa hai điểm lai ghép. Các kỹ thuật lai ghép khác có thể kể đến là:

- Uniform crossover [14]
- Half uniform crossover [15]
- Three parents crossover [16]
- Partially matched crossover [17]
- Multi-point crossover [18]

2.1.4 Đột biến (mutation)

Đột biến là toán tử cuối cùng của giải thuật di truyền, trong đó một hoặc nhiều gen được thay đổi sau khi tạo ra các giải pháp con. Tỷ lệ đột biến được đặt ở mức thấp trong giải thuật di truyền vì tỷ lệ đột biến cao sẽ chuyển giải thuật thành

tìm kiếm ngẫu nhiên nguyên thủy. Toán tử đột biến duy trì sự đa dạng của quần thể bằng cách cung cấp thêm một cấp độ ngẫu nhiên nữa. Trên thực tế, toán tử này ngăn các giải pháp trở nên giống nhau và giảm khả năng tối ưu cục bộ trong giải thuật di truyền. Một ví dụ khái niệm của toán tử này được minh họa trong hình 2.3. Có thể thấy trong hình này rằng những thay đổi nhỏ ở một số gen được chọn ngẫu nhiên xảy ra sau giai đoạn lai ghép.



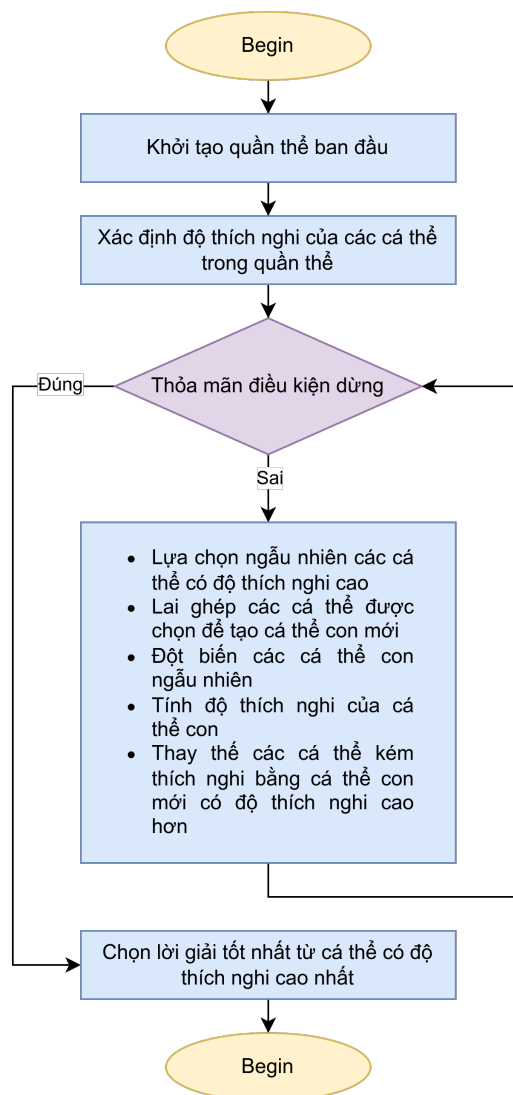
Hình 3: Toán tử đột biến thay đổi một hoặc nhiều gen của các nhiễm sắc thể con sau khi quá trình lai ghép diễn ra

Kết hợp lại với nhau, hầu hết các giải thuật tiến hóa đều sử dụng ba toán tử tiến hóa: chọn lọc, lai ghép và đột biến. Các toán tử này được áp dụng cho từng thế hệ nhằm nâng cao chất lượng gen ở thế hệ tiếp theo. Một toán tử tiến hóa phổ biến khác là chủ nghĩa tinh hoa [19], trong đó một hoặc nhiều giải pháp tốt nhất được duy trì và chuyển giao mà không sửa đổi sang thế hệ tiếp theo. Mục tiêu chính là ngăn chặn các giải pháp như vậy (tinh hoa) bị xuống cấp khi áp dụng các toán tử chéo hoặc đột biến.

Giải thuật di truyền bắt đầu với một nhóm cá thể ngẫu nhiên. Cho đến hết tiêu chí cuối cùng, thuật toán này cải thiện tổng thể bằng cách sử dụng ba toán tử nêu trên. Giải pháp tốt nhất trong quần thể cuối cùng được trả về dưới dạng xấp xỉ

tốt nhất của mức tối ưu toàn cục cho một vấn đề nhất định. Tỷ lệ lựa chọn, chéo và đột biến có thể được thay đổi hoặc đặt để cố định các số trong quá trình tối ưu hóa. Các phần tiếp theo điều tra tác động của việc thay đổi tỷ lệ như vậy đối với hiệu suất của giải thuật di truyền.

Sơ đồ tổng quan của giải thuật di truyền:



Hình 4: Sơ đồ tổng quan của giải thuật di truyền

2.2 Xây dựng mô hình

Như đã mô tả ở phần giới thiệu, xếp lịch học là bài toán phân công đa chiều, trong đó giảng viên phân công lớp để dạy môn học trong một phòng vào một thời điểm cụ thể.

Trong hệ thống đào tạo theo niên chế, việc lập thời khóa biểu được dựa trên lớp học cố định. Khi thời khóa biểu được thiết lập, các sinh viên trong một lớp đều có lịch học thống nhất và đồng đều. Ngược lại, trong hệ thống đào tạo tín chỉ, việc lập thời khóa biểu dựa trên từng học phần riêng lẻ, và sinh viên có quyền đăng ký các môn học phù hợp với lịch trình cá nhân. Điều này làm cho việc sắp xếp thời khóa biểu trở nên phức tạp hơn nhiều.

Để giảm bớt sự phức tạp trong việc lập thời khóa biểu, ta có thể tổ chức các lớp tín chỉ theo nhóm, với đơn vị cơ sở là ngành học. Số lượng nhóm thời khóa biểu được xác định dựa trên số lượng sinh viên của từng ngành trong mỗi khóa học và sĩ số trung bình cho một lớp tín chỉ. Mỗi nhóm thời khóa biểu có ít nhất một lớp cho mỗi môn học được mở theo kế hoạch đào tạo. Do đó, khái niệm về nhóm thời khóa biểu có thể được coi là tương đương với khái niệm về lớp niên chế. Việc tổ chức lớp tín chỉ theo nhóm giúp sinh viên dễ dàng đăng ký hơn và tạo ra nhiều lựa chọn hơn khi tìm giờ trống để bù giờ.

Việc sắp xếp thời khóa biểu cần đảm bảo thỏa mãn tất cả các ràng buộc cứng và thỏa mãn tối đa các ràng buộc mềm [20]:

Gọi T là tập hợp các giảng viên; mỗi lớp học của từng môn sẽ được phân công giảng viên trước đó.

Gọi C là tập hợp các lớp. Mỗi lớp có số tiết giảng cố định trong một tuần; và lớp c chỉ được giao cho một giảng viên duy nhất tại bất kỳ thời điểm nào.

Gọi S là tập hợp các môn học.

Gọi R một tập hợp các phòng học.

Gọi P là tập hợp các tiết dạy trong ngày.

Gọi D là tập hợp các ngày dạy

Đặt $A_{t,s}$ là tập hợp các khoảng thời gian mà giảng viên dạy môn s có mặt.

Và gọi các biến quyết định là:

$$X_{t,c,s,r,p,d} = \begin{cases} 1 & \text{nếu giảng viên } t \text{ dạy môn học } s \text{ cho lớp } c \text{ tại phòng } r \text{ vào tiết } p \text{ của ngày } d \\ 0 & \text{nếu ngược lại} \end{cases}$$

2.2.1 Ràng buộc cứng

Những ràng buộc cứng là yếu tố then chốt trong việc xây dựng mô hình giải thuật xếp thời khóa biểu. Bất kỳ giải pháp khả thi nào cũng phải thỏa mãn mọi ràng buộc cứng. Tập hợp các ràng buộc cứng gồm các ràng buộc như sau [21]:

HC1: Một giảng viên không được phép xếp dạy nhiều hơn một lớp tại bất kỳ tiết học nào

$$\sum_{c \in C} X_{t,c,s,r,p,d} \leq 1 \quad \forall t \in T_s, s \in S, r \in R, p \in P, d \in D \quad (1)$$

HC2: Một lớp học không thể được phân công cho nhiều hơn một giảng viên tại bất kỳ tiết học nào

$$\sum_{s \in S} \sum_{t \in T_s} X_{t,c,s,r,p,d} \leq 1 \quad \forall c \in C, r \in R, p \in P, d \in D \quad (2)$$

HC3: Một phòng học không thể sắp xếp cho nhiều hơn một lớp tại bất kỳ tiết học nào

$$\sum_{s \in S} \sum_{t \in T_s} \sum_{c \in C} X_{t,c,s,r,p,d} \leq 1 \quad \forall r \in R, p \in P, d \in D \quad (3)$$

HC4: Lớp học c phải tổ chức l_c tiết trên một tuần

$$\sum_{s \in S} \sum_{t \in T} \sum_{c \in C} X_{t,c,s,r,p,d} \leq 1 \forall r \in R, p \in P, d \in D \quad (4)$$

2.2.2 Ràng buộc mềm

Các ràng buộc mềm được sử dụng như một công cụ đánh giá chất lượng của một phương án sắp xếp thời khóa biểu. Các ràng buộc này có thể thay đổi tùy theo quy chế của các trường Đại học, ví dụ thời gian ưu tiên của giảng viên, các bài giảng liên tiếp riêng biệt cho sinh viên hoặc tối ưu việc sử dụng phòng. Sau đây là một số ràng buộc mềm dựa theo nhu cầu giảng dạy, học tập tại các trường Đại học.

SC1 - Nhu cầu tiết học của giảng viên

Phân công giảng viên vào một nhóm tiết học dựa trên nhu cầu của giảng viên. Mỗi giảng viên liệt kê các ưu tiên về tiết học để dạy hoặc nghỉ. Cho $T = \{1, 2, \dots, t\}$ là một tập hợp các giảng viên và $P = \{1, 2, \dots, p\}$ là một tập hợp các khe thời gian, trong đó $(t \leq p)$. $c_{i,j}$ được định nghĩa là sự ưu tiên của giảng viên i đối với việc được phân công vào dạy tiết j . Ma trận nhu cầu của giảng viên theo tiết học $(t \times p)$ được hình thành để chứa thông tin về sự ưu tiên của giảng viên như sau: Giá trị 1 biểu thị lựa chọn tiết học đầu tiên của giảng viên, giá trị 2 biểu thị lựa chọn tiết học thứ hai, v.v. cho đến một số cụ thể biểu thị số lượng khe thời gian tối đa p . Nếu $c_{i,j}$ chưa được gán một giá trị nguyên (tức là giảng viên i chưa có tiết j trong danh sách nhu cầu của họ), thì $c_{i,j}$ được gán giá trị phạt B có độ lớn phù hợp. Ngoài ra, ta có thể gán số trọng số ưu tiên cho mỗi giảng viên để giúp họ có khả năng lớn hơn được phân tiết học theo nhu cầu của mình. Công thức lập trình toán học có thể được minh họa như sau [22]:

$$\text{Min} \sum_{i=1}^t \sum_{j=1}^p w_i c_{i,j} x_{i,j} \quad (5)$$

$$s.t. \sum_{j=1}^t x_{i,j} = 1 \forall i \in p \quad (6)$$

Phương trình (6) đảm bảo điều kiện mỗi giảng viên chỉ được phân công vào một tiết học

$$\sum_{i=1}^t x_{i,j} \leq 1 \forall j \in p \quad (7)$$

Phương trình (7) chỉ ra rằng mỗi tiết học chỉ được sắp xếp tối đa một lần

$$x_{i,j} \in (0,1) \forall i \in p, j \in T \quad (8)$$

Trong đó:

$$X_{i,j} = \begin{cases} 1 & \text{Nếu giảng viên } i \text{ được phân công dạy tiết } j \\ 0 & \text{nếu ngược lại} \end{cases} \forall i \in P, j \in T, c_{i,j} \in B \text{ if } c_{i,j} \notin T$$

SC2: Tối ưu sử dụng phòng

Gọi $R = [1, 2, \dots, m]$ là tập hợp các phòng, $D = [1, 2, \dots, y]$ là tập hợp ngày và $P = [1, 2, \dots, n]$ là tập hợp các tiết học giảng dạy trong ngày. Sức chứa của các phòng là RC_r và gọi $SN_{r,d,t}$ là số sinh viên theo học khóa học được phân vào phòng r trong khung thời gian p của ngày d . Việc sử dụng phòng có thể được hình thành dưới dạng hàm chi phí tối thiểu hóa. Hàm chi phí được tính bằng chênh lệch giữa số lượng sinh viên và sức chứa phòng. Công thức toán học như sau [20]:

$$\text{Min} \sum_{r=1}^m \sum_{d=1}^y \sum_{p=1}^n RC_r - SN_{r,d,p} \quad (9)$$

$$s.t. \sum_{r=1}^m SN_r \leq R_i \text{ and } SN_{r,d,p} \text{ exists} \quad (10)$$

SC3: Tất cả các lớp đều phải được xếp lịch

Gọi ct_c là số tiết cần giảng dạy của lớp c . Hàm chi phí được tính bằng chênh lệch giữa tổng số tiết cần phải xếp và số tiết đã xếp:

$$\sum_{c \in C} ct_c - \sum_{c \in C} X_{t,c,s,r,p,d} \forall t \in T, s \in S, r \in R, p \in P, d \in D \quad (11)$$

SC4: Các lớp trong cùng một nhóm khóa ngành không được phép trùng lịch

Để đảm bảo sinh viên có thể đăng ký đủ các môn học theo chương trình đào tạo kế hoạch, các lớp học của các môn tương ứng trong cùng một nhóm khóa ngành không được phép trùng lịch với nhau [23]

2.2.3 Hàm thích nghi

Giá trị thích nghi của một nhiễm sắc thể phương án có thể biểu diễn như sau [24]:

$$eval(f) = \frac{1}{1 + cost(f)} \quad (12)$$

Trong đó $cost(f)$ là giá trị đánh giá việc vi phạm các ràng buộc:

$$cost(f) = \sum_{i=1}^{ct} n_i(f) \times W_i \quad (13)$$

Trong đó ct là tổng số lượng ràng buộc, $n_i(f)$ là mức độ phạt của việc vi phạm ràng buộc i , và w_i là trọng số của ràng buộc.

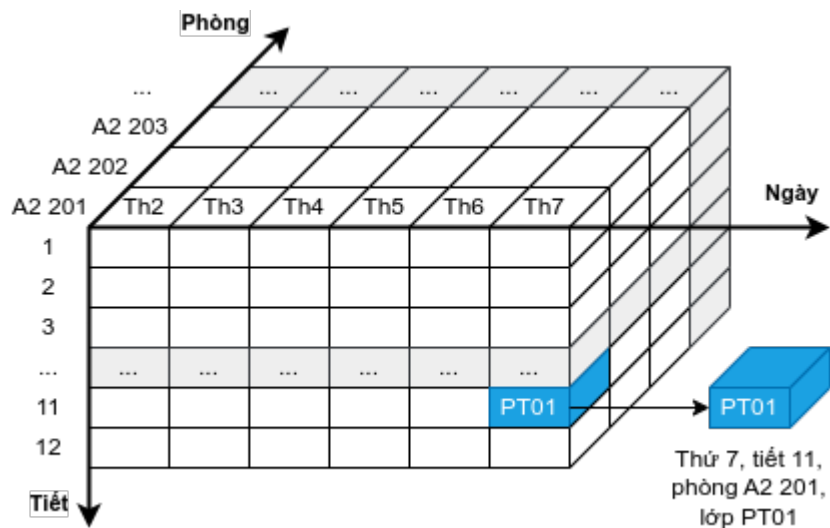
Mục tiêu của giải thuật là giảm thiểu số lượng vi phạm ràng buộc mềm trong một giải pháp được tạo ra dựa trên các ràng buộc cứng.

2.2.4 Biểu diễn nhiễm sắc thể

Ta có thể coi mỗi nhiễm sắc thể là một mảng 3 chiều [23]: Chiều thứ nhất biểu diễn các ngày trong tuần, chiều thứ hai biểu diễn các tiết học trong ngày, chiều thứ ba biểu diễn các phòng học. Thông thường, thời khóa biểu của sinh viên sẽ cố định theo tuần để đảm bảo sự ổn định, do đó ta mô hình hóa nhiễm sắc thể là

phương án sắp xếp của một tuần học chưa tính đến các ràng buộc về ngày nghỉ và nhân bản ra các tuần còn lại. Điều này không làm ảnh hưởng tới tính đúng đắn của phương án sắp xếp do khi áp dụng vào các tuần cụ thể, phương án chỉ bớt đi các lịch bị ràng buộc chứ không bổ sung thêm.

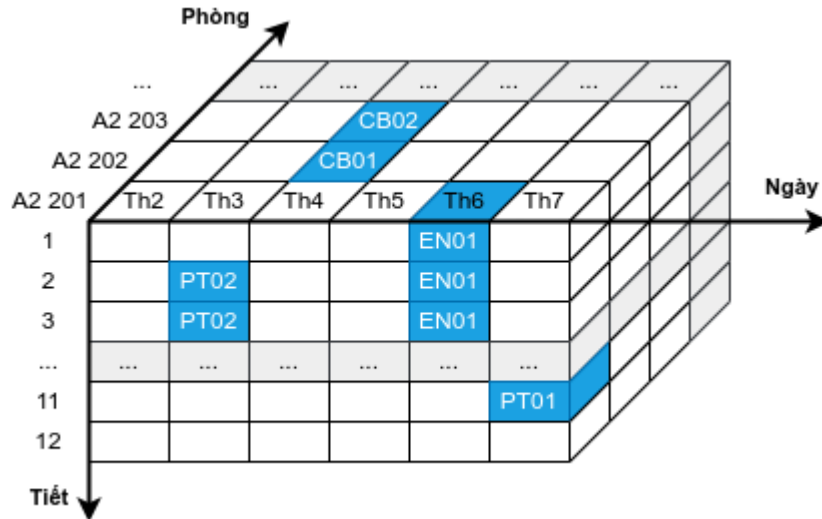
Cách biểu diễn này cũng đảm bảo nhiệm sắc thể không vi phạm ràng buộc cứng HC3, do mỗi ô trong mảng ba chiều chỉ được phân một lớp học



Hình 5: Biểu diễn nhiệm sắc thể dưới dạng mảng ba chiều

2.2.5 Khởi tạo quần thể

Việc khởi tạo quần thể được thực hiện bằng cách khởi tạo ngẫu nhiên các nhiệm sắc thể trong quần thể. Một nhiệm sắc thể được khởi tạo bằng cách điền lần lượt các lớp học vào các vị trí ngẫu nhiên còn trống trong mảng ba chiều. Nếu một lớp có số tiết trong một buổi lớn hơn một thì sẽ điền liên tiếp vào các tiết tính từ tiết bắt đầu được chọn.



Hình 6: Điền lần lượt các lớp học vào các tiết ngẫu nhiên

2.2.6 Toán tử chọn lọc

Do cấu trúc biểu diễn nhiễm sắc thể khá phức tạp nên ta chọn toán tử lai ghép đa điểm để áp dụng với các điểm tạo ngẫu nhiên [23]

Chọn hai *NST* gốc lai ghép ngẫu nhiên: N_1, N_2
 Gọi hai *NST* con được sinh ra: C_1, C_2
 Tạo mặt nạ lai ghép nhị phân M với các điểm lai ghép ngẫu nhiên
 For each *gen* in *NST*:
 Begin
 If ($M_{gen}=1$) Then
 C_1 nhận *gen* từ *NST* gốc N_1
 C_2 nhận *gen* từ *NST* gốc N_2
 If ($M_{gen}=0$) Then
 C_1 nhận *gen* từ *NST* gốc N_2
 C_2 nhận *gen* từ *NST* gốc N_1
 End

2.2.7 Toán tử đột biến

Sau khi thực hiện xong các toán tử lai ghép và thu được thế hệ nhiễm sắc thể mới, ta thực hiện toán tử đột biến với một số lượng các nhiễm sắc thể con ngẫu nhiên.

```

Gọi  $p_m$  là xác suất đột biến
For each gen in NST:
Begin
     $x = \text{Số nguyên ngẫu nhiên trong khoảng từ 1 đến 1000}$ 
    If  $(x < p_m \times 1000)$  Then
        Begin
             $rangen = \text{Là một gen ngẫu nhiên trong NST}$ 
             $gen = rangen$ 
        End
    End
End

```

2.2.8 Cải tiến

Các toán tử lai ghép và đột biến của giải thuật di truyền chủ yếu dựa vào sự ngẫu nhiên, dẫn đến khó kiểm soát sự phát triển của giải thuật và tăng khả năng vi phạm các ràng buộc. Để giảm thiểu tình trạng này, ta có thể cải tiến các toán tử lai ghép và đột biến

2.2.8.1 Cải tiến lai ghép

Với mỗi vị trí lai ghép, ta kiểm tra thêm việc lai ghép có vi phạm các ràng buộc cứng không, nếu vi phạm thì ta bỏ qua không thực hiện lai ghép. Việc loại bỏ trước các cá thể vi phạm ràng buộc cứng không làm tăng khả năng giải thuật bị tối ưu cục bộ, vì việc trao đổi thông tin gen dẫn đến vi phạm các ràng buộc cứng sẽ không giúp ích cho việc tối ưu các nhiễm sắc thể thế hệ mới. Thay vào đó ta biến đổi phép lai ghép để luôn đảm bảo tất cả các thế hệ nhiễm sắc thể đều thỏa mãn các

ràng buộc cứng. Điều này cũng giúp giảm độ phức tạp tính toán hàm thích nghi bằng cách bỏ qua các ràng buộc cứng.

```

Chọn hai NST gốc lai ghép ngẫu nhiên:  $N_1, N_2$ 
Gọi hai NST con được sinh ra:  $C_1, C_2$ 
Tạo mặt nạ lai ghép nhị phân  $M$  với các điểm lai ghép ngẫu nhiên
For each  $gen$  in  $NST$ :
Begin
    If ( $M_{gen}=1$ ) Then
         $C'_1 = C_1$  nhận  $gen$  từ NST gốc  $N_1$ 
         $C'_2 = C_2$  nhận  $gen$  từ NST gốc  $N_2$ 
        If  $C'_1$  và  $C'_2$  đều không vi phạm  $HC1, HC2, HC3, HC4$  Then
             $C_1 = C'_1$ 
             $C_2 = C'_2$ 
    If ( $M_{gen}=0$ ) Then
         $C'_1 = C_1$  nhận  $gen$  từ NST gốc  $N_2$ 
         $C'_2 = C_2$  nhận  $gen$  từ NST gốc  $N_1$ 
        If  $C'_1$  và  $C'_2$  đều không vi phạm  $HC1, HC2, HC3, HC4$  Then
             $C_1 = C'_1$ 
             $C_2 = C'_2$ 
End

```

2.2.8.2 Cải tiến đột biến

Nếu cải tiến lai ghép giúp loại bỏ từ đầu các phương án vi phạm ràng buộc cứng, thì việc cải tiến đột biến sẽ giúp việc tối ưu các ràng buộc mềm diễn ra với tốc độ nhanh hơn, bằng việc thêm các toán tử có tính định hướng cho việc thay đổi gen trên nhiễm sắc thể.

Việc tăng tốc tối ưu ràng buộc mềm giúp tốc độ hội tụ của thuật toán nhanh hơn nhưng cũng có thể làm mất mát các thông tin hữu ích trong quá trình di truyền, dẫn đến tình trạng tối ưu cục bộ. Vì vậy ta cần bổ sung các toán tử đột biến có tính biến dị mạnh.

2.3 Đánh giá

Kết quả đánh giá trên bộ dữ liệu thời khóa biểu học kỳ 1 năm học 2023-2024 của hệ Đại học Chính quy, học viện Công nghệ Bưu chính Viễn thông. Dữ liệu bao gồm:

- 326 môn học
- 14 tiết học
- 175 phòng học
- 1555 lớp học trung bình 3 tiết trên tuần
- 272 giảng viên
- Lịch học từ thứ 2 đến thứ 7

Tổng số tiết tối đa có thể xếp trong một tuần: 14700. Tổng số tiết cần xếp trong một tuần: ~ 5000 .

Giải thuật được thử nghiệm đánh giá với các bộ tham số đầu vào sau:

- P1: Kích thước quần thể ban đầu
- P2: Số vòng lặp tối đa
- P3: Phần trăm số tiết trống (dự trù cho các lịch dạy bù)
- P4: Số lớp tối thiểu học cùng thời gian

Tập giá trị của các tham số:

- P1: {20, 40}
- P2: {500, 1000}
- P3: {5, 15}

- P4: {2, 3}

Các ràng buộc được đánh trọng số ưu tiên. Các ràng buộc có trọng số cao hơn sẽ làm tăng điểm phạt nhiều hơn, do đó chúng sẽ được ưu tiên tối ưu trước. Ta bỏ qua các ràng buộc cứng ở bước tối ưu lai ghép, do đó chỉ có các ràng buộc mềm được đánh trọng số

Bảng trọng số các ràng buộc mềm

Bảng 3: Bảng trọng số ràng buộc mềm

| Ràng buộc | Trọng số |
|-----------|----------|
| SC1 | 1 |
| SC2 | 1 |
| SC3 | 2 |
| SC4 | 2 |

Bảng kết quả đánh giá bao gồm giá trị các tham số đầu vào, thời gian thực thi và ước tính hiệu quả dựa trên độ thích nghi tốt nhất của các nhiễm sắc thể ứng với mỗi bộ tham số đầu vào.

Bảng 4: Bảng đánh giá hiệu quả giải thuật

| STT | P1 | P2 | P3 | P4 | Thời gian (phút) | Độ thích nghi tốt nhất |
|-----|----|------|----|----|------------------|------------------------|
| 1 | 20 | 500 | 5 | 2 | 2 | 0.94 |
| 2 | 20 | 500 | 5 | 3 | 2 | 0.78 |
| 3 | 20 | 500 | 15 | 2 | 2.5 | 0.87 |
| 4 | 20 | 500 | 15 | 3 | 2.5 | 0.75 |
| 5 | 20 | 1000 | 5 | 2 | 4 | 0.97 |
| 6 | 20 | 1000 | 5 | 3 | 4 | 0.81 |
| 7 | 20 | 1000 | 15 | 2 | 4.2 | 0.90 |

| STT | P1 | P2 | P3 | P4 | Thời gian (phút) | Độ thích nghi tốt nhất |
|-----|----|------|----|----|------------------|------------------------|
| 8 | 20 | 1000 | 15 | 3 | 4.2 | 0.78 |
| 9 | 40 | 500 | 5 | 2 | 5.1 | 0.95 |
| 10 | 40 | 500 | 5 | 3 | 5.1 | 0.77 |
| 11 | 40 | 500 | 15 | 2 | 5.7 | 0.88 |
| 12 | 40 | 500 | 15 | 3 | 5.7 | 0.75 |
| 13 | 40 | 1000 | 5 | 2 | 10 | 0.97 |
| 14 | 40 | 1000 | 5 | 3 | 10 | 0.81 |
| 15 | 40 | 1000 | 15 | 2 | 10.3 | 0.97 |
| 16 | 40 | 1000 | 15 | 3 | 10.3 | 0.87 |

Từ bảng đánh giá, ta nhận thấy kích thước ban đầu của quần thể không ảnh hưởng đến độ tối ưu của giải thuật, do việc khởi tạo quần thể là ngẫu nhiên. Việc tăng số lớp tối thiểu học cùng thời gian ảnh hưởng trực tiếp tới tính hiệu quả của giải thuật, do càng nhiều lớp học cùng một thời điểm sẽ làm tăng khả năng các lớp trong cùng một nhóm khóa ngành bị trùng lịch, dẫn tới vi phạm ràng buộc mềm SC4.

2.4 Kết luận

Chương này trình bày về khái niệm của giải thuật di truyền, cách mô hình hóa giải thuật cho bài toán xếp thời khóa biểu tín chỉ, và đánh giá hiệu quả của giải thuật trên tập dữ liệu của Học viện Công nghệ Bưu chính Viễn thông. Chương 3 của luận văn sẽ đề cập tới việc thiết kế và xây dựng ứng dụng xếp thời khóa biểu tín chỉ bán tự động.

CHƯƠNG 3. XÂY DỰNG ỨNG DỤNG XẾP THỜI KHÓA BIỂU BÁN TỰ ĐỘNG

Chương này sẽ trình bày về việc xây dựng ứng dụng xếp thời khóa biểu bán tự động sử dụng giải thuật di truyền đã nghiên cứu tại chương 2.

3.1 Phân tích thiết kế

3.1.1 Danh mục Use-case

Bảng 5: Danh mục Use-case

| STT | Tên use-case | Tên tác nhân chính | Mô tả trường hợp sử dụng |
|-----|------------------|------------------------------|---|
| 1 | Quản lý học phần | Admin Chuyên viên đào tạo | <p>Xem danh sách học phần</p> <p>Xem chi tiết học phần</p> <p>Thêm mới học phần</p> <p>Lưu vết thông tin khởi tạo (ngày tạo, người tạo)</p> <p>Chỉnh sửa thông tin học phần</p> <p>Lưu vết thông tin chỉnh sửa (ngày sửa, người sửa)</p> <p>Xóa 1 hoặc nhiều học phần</p> <p>Tìm kiếm học phần theo trường thông tin học phần</p> <p>Lọc xem học phần theo điều kiện</p> <p>Ấn một hoặc nhiều học phần</p> <p>Bỏ Ấn một hoặc nhiều học phần</p> |

| STT | Tên use-case | Tên tác nhân chính | Mô tả trường hợp sử dụng |
|-----|----------------------------|------------------------------|---|
| 2 | Quản lý chương trình khung | Admin Chuyên viên đào tạo | <p>Xem chương trình khung theo ngành</p> <p>Xem chi tiết thông tin chương trình khung (các kỳ học, các học phần theo kỳ, khóa sinh viên đang áp dụng)</p> <p>Thêm mới thông tin chương trình khung (thông tin chung, thông tin các kỳ học, các học phần trong kỳ, danh sách khóa sinh viên áp dụng)</p> <p>Lưu vết thông tin khởi tạo (ngày tạo, người tạo)</p> <p>Chỉnh sửa thông tin chương trình khung (thông tin chung, thông tin các kỳ học, các học phần trong kỳ, danh sách khóa sinh viên áp dụng)</p> <p>Lưu vết thông tin chỉnh sửa (ngày sửa, người sửa)</p> <p>Xóa 1 hoặc nhiều chương trình khung</p> <p>Lọc xem chương trình khung theo điều kiện</p> |

| STT | Tên use-case | Tên tác nhân chính | Mô tả trường hợp sử dụng |
|-----|---|------------------------------|---|
| | | | <p>Tìm kiếm thông tin chương trình khung theo trường thông tin</p> <p>Ấn một hoặc nhiều chương trình khung</p> <p>Bỏ Ấn một hoặc nhiều chương trình khung</p> |
| 3 | Quản lý thời khóa biểu (quản lý tiết học) | Admin Chuyên viên đào tạo | <p>Xem thời khóa biểu theo kỳ năm học</p> <p>Xem thời khóa biểu theo lớp tín chỉ</p> <p>Xuất thời khóa biểu theo kỳ</p> <p>Xuất thời khóa biểu theo lớp tín chỉ</p> <p>Import thời khóa biểu theo kỳ</p> <p>Import thời khóa biểu theo lớp tín chỉ</p> <p>Thêm mới thời khóa biểu lớp tín chỉ</p> <p>Lưu vết thông tin khởi tạo (ngày tạo, người tạo)</p> <p>Chỉnh sửa thông tin thời khóa biểu lớp tín chỉ</p> |

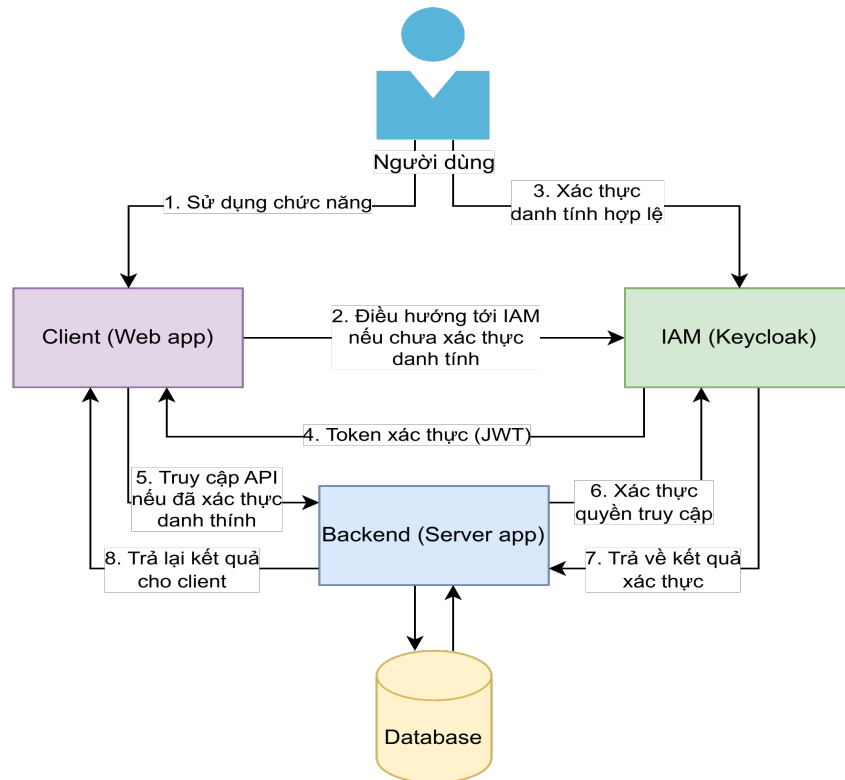
| STT | Tên use-case | Tên tác nhân chính | Mô tả trường hợp sử dụng |
|-----|--------------------------------------|------------------------------|--|
| | | | <p>Lưu vết thông tin chỉnh sửa (ngày sửa, người sửa)</p> <p>Xóa 1 hoặc nhiều thời khóa biểu kỳ học</p> <p>Xóa 1 hoặc nhiều thời khóa biểu lớp tín chỉ</p> <p>Tìm kiếm thời khóa biểu theo trường thông tin của thời khóa biểu</p> <p>Lọc xem thời khóa biểu theo điều kiện</p> |
| 4 | Quản lý thời khóa biểu cá nhân | Admin Chuyên viên đào tạo | <p>Xem thời khóa biểu theo ngày</p> <p>Xem thời khóa biểu theo tuần</p> <p>Xem thời khóa biểu theo tháng</p> <p>Thêm mới lịch cá nhân</p> <p>Lưu vết thông tin khởi tạo (ngày tạo, người tạo)</p> <p>Chỉnh sửa lịch cá nhân</p> <p>Lưu vết thông tin chỉnh sửa (ngày sửa, người sửa)</p> <p>Xóa lịch cá nhân</p> <p>Phân biệt lịch theo màu</p> <p>Tìm kiếm lịch</p> |
| 5 | Thông tin | Admin | <p>Thống kê số tiết (lý thuyết,</p> |

| STT | Tên use-case | Tên tác nhân chính | Mô tả trường hợp sử dụng |
|-----|-------------------------|---------------------|--|
| | tổng hợp thời khóa biểu | Chuyên viên đào tạo | thực hành, ...) trong kỳ Thống kê số tiết (lý thuyết, thực hành, ...) theo lớp tín chỉ Thống kê số tiết (lý thuyết, thực hành, ...) theo giảng viên Thống kê số tiết đã hoàn thành/số tiền chưa hoàn thành Thống kê số lịch cá nhân theo ngày/tuần/tháng |

3.2 Kiến trúc tổng quan

Kiến trúc tổng quan của hệ thống bao gồm các thành phần chính sau:

- Ứng dụng web (client): Là thành phần giao tiếp trực tiếp với người dùng, cung cấp cho người dùng các thao tác tương tác với hệ thống
- Ứng dụng backend (server): Là thành phần chịu trách nhiệm cung cấp các API xử lý nghiệp vụ cho client
- Hệ thống định danh xác thực người dùng (IAM): Là thành phần có chức năng xác thực danh tính và quyền truy cập của người dùng
- Cơ sở dữ liệu (database): Là thành phần lưu trữ dữ liệu dạng cấu trúc/phi cấu trúc, được hình thành từ quá trình vận hành và sử dụng ứng dụng
- Client kết nối với Backend thông qua các API với giao thức HTTP



Hình 7: Mô hình hoạt động của hệ thống

3.3 Công nghệ

3.3.1 ReactJS (Client)

ReactJS là một thư viện JavaScript mã nguồn mở được phát triển và quản lý bởi Facebook, được sử dụng để xây dựng các giao diện người dùng (UI) tương tác cho các ứng dụng web. Ra mắt vào năm 2011, ReactJS đã nhanh chóng trở thành một trong những thư viện JavaScript phổ biến nhất hiện nay nhờ vào những ưu điểm nổi bật sau:

Mô hình lập trình dựa trên thành phần (Component-based):

ReactJS sử dụng mô hình lập trình dựa trên thành phần, giúp cho việc xây dựng các UI phức tạp trở nên dễ dàng và hiệu quả hơn. Mỗi thành phần trong ReactJS đều có một chức năng riêng và có thể được tái sử dụng nhiều lần trong ứng dụng. Điều này giúp cho các nhà phát triển có thể tiết kiệm thời gian và công sức khi viết code.

Sử dụng JSX:

JSX là một cú pháp mở rộng của JavaScript, được sử dụng để viết các thành phần ReactJS. JSX giúp cho việc viết code ReactJS trở nên dễ dàng và trực quan hơn, so với việc viết code JavaScript thuần túy.

Virtual DOM:

ReactJS sử dụng thuật toán render ảo (virtual DOM) để tối ưu hóa hiệu suất của ứng dụng. Virtual DOM là một bản sao nhẹ của DOM thực (Document Object Model), giúp cho ReactJS có thể xác định nhanh chóng những phần nào của UI cần được cập nhật khi có thay đổi trong state của ứng dụng. Nhờ vậy, các ứng dụng ReactJS có thể chạy nhanh và mượt mà hơn.

Data Binding một chiều:

ReactJS sử dụng data binding một chiều, có nghĩa là dữ liệu chỉ chảy từ thành phần cha sang thành phần con. Điều này giúp cho việc quản lý state của ứng dụng trở nên dễ dàng và hiệu quả hơn.

3.3.2 NestJS (Backend)

NestJS là một framework JavaScript mã nguồn mở được xây dựng trên nền tảng Express và TypeScript, được sử dụng để phát triển các ứng dụng web server-side mạnh mẽ, có thể mở rộng và dễ bảo trì. NestJS ra mắt vào năm 2018 và nhanh chóng trở thành một trong những framework Node.js phổ biến nhất hiện nay nhờ vào những ưu điểm nổi bật sau:

Kiến trúc hướng đối tượng:

NestJS sử dụng kiến trúc hướng đối tượng (OOP) giúp cho việc tổ chức code trở nên rõ ràng, dễ hiểu và dễ bảo trì hơn. Việc sử dụng OOP cũng giúp cho các nhà phát triển có thể dễ dàng tái sử dụng code và xây dựng các ứng dụng web phức tạp.

Mô hình MVC:

NestJS sử dụng mô hình MVC (Model-View-Controller) giúp cho việc phân tách các mối quan tâm (separation of concerns) trong ứng dụng trở nên rõ ràng hơn. Mô hình MVC giúp cho các nhà phát triển có thể dễ dàng viết code dễ hiểu, dễ bảo trì và dễ kiểm thử.

Hỗ trợ Dependency Injection:

NestJS hỗ trợ Dependency Injection (DI) là một kỹ thuật giúp cho việc quản lý các phụ thuộc trong ứng dụng trở nên dễ dàng hơn. DI giúp cho các nhà phát triển có thể viết code linh hoạt, dễ kiểm thử và dễ mở rộng.

Hỗ trợ TypeScript:

NestJS được xây dựng trên nền tảng TypeScript là một ngôn ngữ mở rộng của JavaScript, cung cấp kiểu dữ liệu tĩnh và các tính năng nâng cao, giúp cho code dễ đọc và dễ bảo trì hơn.

3.3.3 Keycloak (IAM)

Keycloak là một server bảo mật OAuth 2.0 và OpenID Connect mã nguồn mở, được sử dụng để bảo mật các ứng dụng web và mobile. Keycloak được phát triển bởi Red Hat và được cộng đồng hỗ trợ tích cực. Keycloak đã trở thành một trong những giải pháp bảo mật OAuth phổ biến nhất hiện nay nhờ vào những ưu điểm nổi bật sau:

Dễ cài đặt và sử dụng:

Keycloak có thể được cài đặt và sử dụng dễ dàng trên các hệ điều hành phổ biến như Linux, Windows và macOS. Keycloak cũng cung cấp giao diện quản trị web trực quan giúp cho việc quản lý tài khoản người dùng, vai trò và quyền truy cập trở nên dễ dàng.

Hỗ trợ nhiều giao thức bảo mật:

Keycloak hỗ trợ nhiều giao thức bảo mật phổ biến như OAuth 2.0, OpenID Connect, SAML và Kerberos. Điều này giúp cho Keycloak có thể được tích hợp với nhiều ứng dụng và hệ thống khác nhau.

Có thể mở rộng và tùy chỉnh:

Keycloak là một framework có thể mở rộng và tùy chỉnh cao. Các nhà phát triển có thể dễ dàng thêm các tính năng mới vào Keycloak hoặc thay đổi hành vi mặc định của nó để phù hợp với nhu cầu của mình.

Bảo mật cao:

Keycloak được thiết kế để bảo mật cao và đáp ứng các yêu cầu bảo mật nghiêm ngặt. Keycloak sử dụng các thuật toán mã hóa mạnh mẽ và các biện pháp bảo mật tốt nhất để bảo vệ dữ liệu người dùng.

3.4 Chức năng chính

Như đã giới thiệu trong chương 1, quy trình xếp thời khóa biểu tại các trường Đại học gồm các công việc sau:

Bảng 6: Bảng công việc xếp thời khóa biểu

| STT | Tên công việc | Đối tượng thực hiện | Kết quả |
|-----|---------------------------------------|---------------------|---|
| 1 | Rà soát nhu cầu học tập của sinh viên | Phòng đào tạo | Danh sách lớp tín chỉ gồm môn học, số thứ tự nhóm, sĩ số, số tiết giảng dạy trên tuần |
| 2 | Báo giảng | Phòng đào tạo | Chuyển danh sách lớp cho các |

| | | | |
|---|---|------------------------|--|
| | | | khoa/bộ môn |
| 3 | Phân công giảng dạy | Khoa/bộ môn | Phân công giảng viên thuộc khoa/bộ vào các lớp tín chỉ, ghi chú lịch nghỉ của thỉnh giảng (nếu có) |
| 4 | Tiết lập danh mục học phần | Phòng đào tạo | Danh mục học phần gồm khoa/bộ môn quản lý, số tín chỉ, loại phòng học |
| 5 | Thiết lập danh mục phòng học | Quản lý cơ sở vật chất | Danh mục phòng học gồm trạng thái, loại phòng, sức chứa tối đa |
| 6 | Tổng hợp dữ liệu phân công giảng viên và sắp xếp thời khóa biểu | Phòng đào tạo | Thời khóa biểu lớp tín chỉ |

3.4.1 Rà soát nhu cầu học phần


Thông thường trước mỗi học kỳ, các trường giao nhiệm vụ cho cố vấn học tập tổng hợp nhu cầu đăng ký học phần của sinh viên. Kết quả nhu cầu đăng ký sẽ được sử dụng để lập danh sách lớp tín chỉ trong học kỳ. Tuy nhiên ta có thể dựa vào chương trình đào tạo của các khóa ngành và kết quả học tập học phần của sinh viên để dự tính các số liệu này, bao gồm số lượng nhu cầu học theo chương trình, số lượng nhu cầu học lại, học cải thiện.

3.4.1.1 Chương trình đào tạo

Mỗi khóa ngành thực hiện đào tạo tại các trường Đại học sẽ có một chương trình đào tạo. Chương trình đào tạo bao gồm danh sách các môn học phần theo khối kiến thức và học kỳ dự kiến của các học phần.

Khung chương trình đào tạo

Xem theo chuyên ngành

 In khung chương trình

Học kỳ 1

| TT | Tên học phần / khối học phần | Mã học phần | Loại học phần | Số tín c |
|----|--------------------------------|-------------|---------------|----------|
| 1 | Giáo dục Quốc phòng-An ninh | DHGQ09 | Bắt buộc | 8 |
| 2 | Tiếng Anh 1 | DHEN01 | Bắt buộc | 3 |
| 3 | Giáo dục thể chất (HP1) | DHGQ03 | Bắt buộc | 1 |
| 4 | Tâm lý học quản trị kinh doanh | DHTL23 | Bắt buộc | 3 |
| 5 | Tin học đại cương | DHIT33 | Bắt buộc | 3 |
| 6 | Quản trị học | DHHT10 | Bắt buộc | 3 |
| 7 | Kinh tế vi mô | DHKT02 | Bắt buộc | 3 |

Dựa theo chương trình đào tạo và học kỳ, ta có thể ước tính được số sinh viên dự kiến học các môn học theo kế hoạch của chương trình đào tạo.

3.4.1.2 Kết quả học tập học phần

Kết quả học tập học phần của sinh viên là điểm số tối đa sinh viên đạt được cho một môn học.

Thông tin điểm học phần sinh viên

Mã SV:

Họ tên:

Trạng thái SV:
Đang học

Mã học phần: DHIT20

Tên học phần: Các vấn đề hiện đại của Công nghệ phần mềm

Số tín chỉ: 2 tín chỉ

Lớp tín chỉ: DHIT20-20231-01

Điểm thành phần

| | | | | | |
|------------|----|------------|----|------------|----|
| Chuyên cần | 10 | Kiểm tra 1 | 9 | Kiểm tra 2 | -- |
| Kiểm tra 3 | -- | Kiểm tra 4 | -- | | |

Điểm kết thúc học phần

| | | | | | |
|----------------|--------------|----------------|-----|----------------|-----|
| Trạng thái thi | Đủ điều kiện | Điểm thi lần 1 | 8.3 | Điểm thẩm định | -- |
| Điểm phúc khảo | -- | Điểm thi lần 2 | -- | Điểm cuối | 8.3 |

Điểm tổng kết

| | | | | | |
|------------|-----|-----------|-----|----------|---|
| Điểm hệ 10 | 8.7 | Điểm hệ 4 | 3.7 | Điểm chữ | A |
|------------|-----|-----------|-----|----------|---|

Đóng

Hình 9: Điểm học phần của sinh viên

Theo quy chế đào tạo, nếu sinh viên chưa đạt đủ điều kiện về điểm cho một môn học thì có thể đăng ký học lại môn học đó trong học kỳ tiếp theo. Do đó kết quả học phần của sinh viên có thể được sử dụng để ước tính số sinh viên đăng ký học lại theo từng môn học.

3.4.1.3 Rà soát nhu cầu

Chỉnh sửa thông số học phần học kỳ

Học kỳ: Học kỳ 1 Năm học 2023 - 2024

Học phần: Tiếng Anh A2 (HP1)

Mã học phần: DHNT15

Số tín chỉ: 2 tín chỉ

Số nhu cầu dự kiến: 774

Sĩ số lớp tín chỉ tối thiểu

Sĩ số lớp tín chỉ tối đa

Sĩ số nhóm thực hành tối thiểu

Sĩ số nhóm thực hành tối đa

Số nhu cầu kế hoạch

Đề cương học phần

Tải lại

Tổng số: 2

| | TT | Phiên bản | Trọng số học phần | | | | | | Trạng thái |
|----------------------------------|----|------------|-------------------|-----|-----|-----|-----|------|------------|
| | | | CC | KT1 | KT2 | KT3 | KT4 | KTHP | |
| <input checked="" type="radio"/> | 1 | DHNT150003 | 10% | 20% | 10% | -- | -- | 60% | Chờ duyệt |
| <input type="radio"/> | 2 | DHNT150002 | 10% | 20% | -- | -- | -- | 70% | Chờ duyệt |

Đã chọn: 1 (Bỏ chọn tất cả)

Tổng số: 2

< 1 >

10 / trang

Lưu lại

Hủy

Hình 10: Cấu hình thông tin rà soát học phần

Rà soát học phần

+ Học kỳ 1 Năm học 2023 - 2024

579 Học phần dự kiến mở lớp **14,585** Số nhu cầu dự kiến **2893** Số lớp tín chỉ dự kiến

Nhu cầu học phần Ý kiến phòng ban Cập nhật đề cương

Chọn bộ môn quản lý Chọn ngành đào tạo Chọn khóa sinh... Học phần mở lớp Tất cả

Thêm mới Rà soát Xuất dữ liệu ☒ Lưu DS lớp tín chỉ Tải lại Tổng số: **579**

| TT | Mã HP | Tên học phần | STC | Thông số mở lớp | | | Số nhu cầu | | Số lớp dự kiến | Thao tác |
|----|--------|---------------------|-----|-----------------|---------|--------------------------|------------|----------|----------------|-------------------|
| | | | | SS lớp | SS nhóm | Đề cương | Dự kiến | Kế hoạch | | |
| 1 | DHNT15 | Tiếng Anh A2 (HP1) | 2 | 80 - 100 | 25 - 50 | DHNT150003 (10-20-10-60) | 774 | 774 | 8 | ✎ |
| 2 | DHNT16 | Tiếng Anh A2 (HP2) | 2 | 50 - 100 | 0 - 0 | DHNT160002 (10-20-10-60) | 696 | 696 | 7 | ✎ |
| 3 | DHNT18 | Tiếng Anh B1 (HP2) | 2 | 50 - 100 | 0 - 0 | DHNT180003 (10-20-10-60) | 659 | 659 | 7 | ✎ |
| 4 | DHNT17 | Tiếng Anh B1 (HP1) | 2 | 50 - 100 | 0 - 0 | DHNT170003 (10-20-10-60) | 624 | 624 | 7 | ✎ |
| 5 | DHCT11 | Logic học đại cương | 2 | 50 - 100 | 0 - 0 | DHCT110003 (10-30-60) | 417 | 417 | 5 | ✎ |

Hình 11: Bảng thống kê nhu cầu học phần

Từ các dữ liệu trên, nhu cầu của sinh viên với các học phần trong học kỳ được tổng hợp. Đề lập danh sách lớp dự kiến, mỗi học phần được cấu hình số tối thiểu và số tối đa, từ đó suy ra số lớp dự kiến. Danh sách lớp tín chỉ có thể được lập sau khi hoàn tất cấu hình học phần.

3.4.2 Phân công giảng dạy

Sau khi lập xong danh sách lớp tín chỉ, phòng đào tạo báo giảng tới các khoa bộ môn. Nhiệm vụ của khoa bộ môn là phân công giảng viên của mình dạy các lớp tín chỉ của các môn học do khoa bộ môn quản lý.

Phân công giảng dạy

+ Học kỳ 2 Năm học 2023 - 2024 **Chọn bộ môn quản lý**

Thời gian phân công giảng dạy từ 29/03/2024 đến 28/04/2024

1 Lớp tín chỉ **0** Lớp đã duyệt **1** Lớp đang xử lý **0** Lớp chưa có đăng ký

Danh sách học phần

Tải lại Tổng số: **194**

| TT | Mã | Tên học phần | STC |
|----|--------|---|-----|
| 1 | DHKT63 | Quản trị rủi ro trong kinh doanh | 3 |
| 2 | DHQG09 | Giáo dục quốc phòng - an ninh | 8 |
| 3 | DHTH14 | Nhập môn quản trị kinh doanh | 3 |
| 4 | DHTH57 | Strategy Management/Quản trị chiến lược | 3 |
| | | Human resource | |

Danh sách lớp tín chỉ DHKT63

Duyệt phân công **Nhập DS phân công** **Tải lại** Tổng số: **1**

| Mã lớp | Sĩ số | Giảng viên | Trạng | Thao tác |
|-----------------|-------|---------------|-------------|----------|
| DHKT63-20232-01 | 5 | Lê Khánh Linh | Đang | |

Tổng số: 1 **1** 10 / trang

Hình 12: Chức năng phân công giảng viên của khoa bộ môn

3.4.3 Thiết lập danh mục phòng học

Danh mục phòng học do chuyên viên cơ sở vật chất quản lý. Danh mục này cần được cập nhật đầy đủ dữ liệu, bao gồm mã phòng, trạng thái, tính chất phòng học, sức chứa. Trong đó:

- Trạng thái phòng thể hiện phòng có thể được sử dụng hay không (Hoạt động/Bảo trì)
- Tính chất phòng học thể hiện các lớp học nào có thể xếp được vào phòng học (Lý thuyết/Thực hành/Thí nghiệm/Giáo dục thể chất)
- Sức chứa thể hiện phòng có thể chứa được sĩ số tối đa ước tính của một lớp hay không ($Sĩ\ số\ tối\ đa \leq Sức\ chứa \times 85\%$)

Thêm mới phòng học, giảng đường

Tòa nhà: A2 | Tầng thứ: 2 | Số phòng: 204

* Mã phòng: TOA_2_204 | * Tên phòng: Phòng 204 - Tầng 2 - TOA_2

* Loại phòng: Lý thuyết | * Trạng thái: Hoạt động

Sức chứa học: 100 | Sức chứa thi: 50

☐ Thiết bị CNC

Thêm mới Hủy

Hình 13: Thông tin phòng học

3.4.4 Thiết lập danh mục môn học

Phòng đào tạo cần rà soát và cập nhật thông tin môn học để chuẩn bị cho việc xếp thời khóa biểu. Số tín chỉ của môn học sẽ ảnh hưởng tới số tiết cần xếp trong một tuần của môn học đó.

Chỉnh sửa học phần

1 Thông tin chung > 2 Đề cương chi tiết > 3 Chương trình áp dụng > 4 HP tương đương

* Trình độ đào tạo: Đại học

* Bộ môn quản lý: Khoa Quản trị Kinh Doanh (15)

* Mã học phần: DHKT63

* Tên học phần (tiếng Việt): Quản trị rủi ro trong kinh doanh

* Số tín chỉ: 3

Tên học phần (tiếng Anh): Nhập tên học phần

* Tính chất học phần: Học phần chuyên môn

Phương pháp tính học phí: Luôn tính theo Tiêu chuẩn

Trạng thái kích hoạt: ☒

Lưu lại Hủy

Hình 14: Các thông tin của môn học

3.4.5 Xếp thời khóa biểu bán tự động

Sau khi các dữ liệu đầu vào đã được hoàn thiện, giải thuật di truyền sẽ được sử dụng để khởi tạo thời khóa biểu. Tài nguyên phòng học của trường có thể ảnh hưởng tới khả năng xếp được toàn bộ lịch học. Trong trường hợp không thể tìm được phương án xếp tối ưu nhất, ứng dụng cần cung cấp chức năng thao tác xếp thời khóa biểu thủ công.

3.4.5.1 Cấu hình tham số giải thuật

Trước khi tiến hành xếp thời khóa biểu, cần thực hiện cấu hình các tham số cho giải thuật di truyền. Các tham số đã được đề cập tại phần 2.4 của chương 2. Sau khi hoàn thành cấu hình tham số, ta có thể bắt đầu xếp thời khóa biểu tự động.

Cấu hình xếp thời khóa biểu

Kích thước quần thể ban đầu

40

Số vòng lặp tối đa

100

* Phần trăm số tiết trống (dự trù cho các lịch dạy bù)

15

%

Số lớp tối thiểu học cùng thời gian

2

Lưu lại

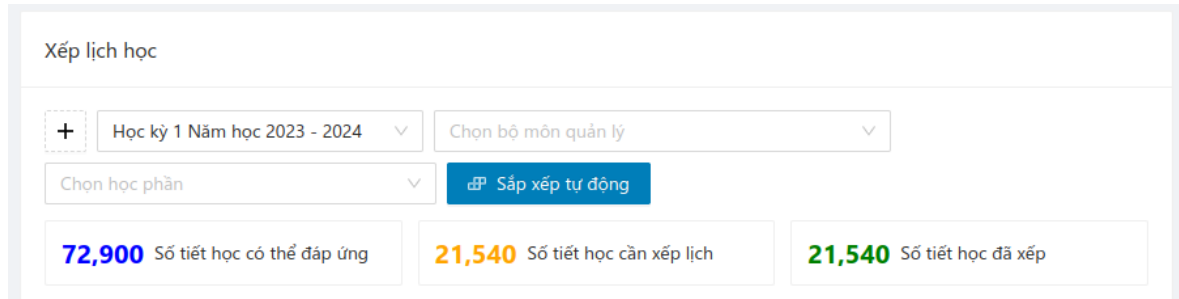
Hình 15: Cấu hình tham số giải thuật di truyền

3.4.5.2 Xếp thời khóa biểu tự động

Các chức năng trong module xếp thời khóa biểu tự động bao gồm:

Dashboard thống kê:

- Số tiết có thể đáp ứng: Cho biết ước tính tương đối về tài nguyên của trường
- Số tiết học cần xếp lịch: Cho biết tổng số tiết học cần xếp lịch của tất cả các lớp tin chỉ trong học kỳ
- Số tiết học đã xếp: Cho biết tổng số tiết học đã được sắp xếp (tự động hoặc thủ công)



Xếp lịch học

+ Học kỳ 1 Năm học 2023 - 2024 ▾ Chọn bộ môn quản lý ▾

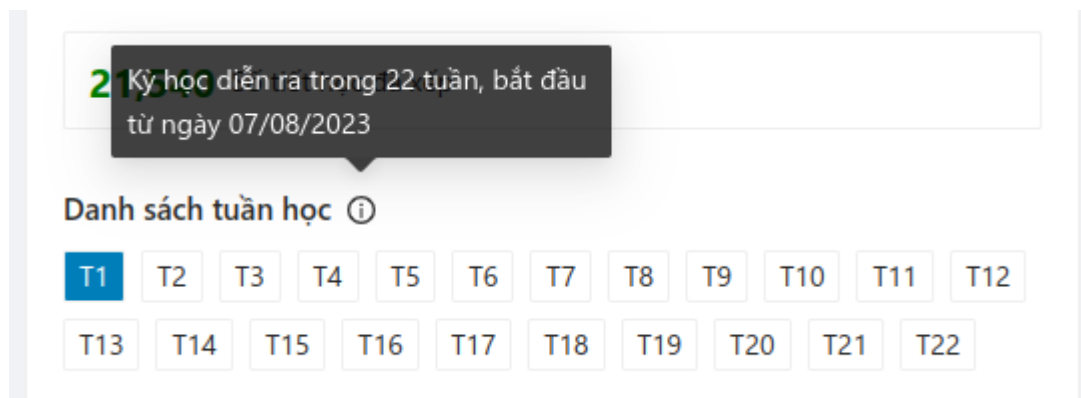
Chọn học phần ▾ [Sắp xếp tự động](#)

72,900 Số tiết học có thể đáp ứng **21,540** Số tiết học cần xếp lịch **21,540** Số tiết học đã xếp

Hình 16: Dashboard xếp thời khóa biểu

Danh sách tuần học:

Cho phép lọc dữ liệu thời khóa biểu theo tuần học của học kỳ.



21,540 Kỳ học diễn ra trong 22 tuần, bắt đầu từ ngày 07/08/2023

Danh sách tuần học ⓘ

| | | | | | | | | | | | |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| T1 | T2 | T3 | T4 | T5 | T6 | T7 | T8 | T9 | T10 | T11 | T12 |
| T13 | T14 | T15 | T16 | T17 | T18 | T19 | T20 | T21 | T22 | | |

Hình 17: Danh sách các tuần học trong kỳ

Danh sách các lớp chưa xếp thời khóa biểu:

Hiển thị các lớp chưa được xếp thời khóa biểu hoặc chưa xếp đủ tiết, trong trường hợp chưa thực hiện xếp tự động, hoặc xếp tự động không thể xếp được lịch học cho toàn bộ lớp.

Lớp tín chỉ chưa xếp thời khóa biểu (49/614) Tải lại

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| DHDL36-20231-01 0/135 (TT3: Thực tập tốt nghiệp) | DHDL36-20231-02 0/135 (TT3: Thực tập tốt nghiệp) | DHBC33-20231-01 0/120 (Thực tập tốt nghiệp) | DHBC33-20231-03 0/120 (Thực tập tốt nghiệp) | DHGQ09-20231-01 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-02 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-03 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-04 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-05 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-06 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-07 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-08 0/120 (Giáo dục Quốc phòng-An ninh) |
| DHGQ09-20231-09 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-10 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-11 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-12 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-13 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-14 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-15 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-16 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-17 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-18 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-19 0/120 (Giáo dục Quốc phòng-An ninh) | DHGQ09-20231-20 0/120 (Giáo dục Quốc phòng-An ninh) |

Hình 18: Danh sách các lớp tín chỉ chưa được xếp đủ lịch học

Xếp thời khóa biểu thủ công:

Hiện thị giao diện thời khóa biểu theo phòng học và theo tuần. Tại đây người dùng có thể thao tác thủ công trên lịch tuần để thêm, sửa hoặc xóa thời khóa biểu.

Phòng học

Đã xếp: 25/58 Tổng số: 58

| TT | Mã | Sức chứa | Đã xếp | Trạng thái |
|----|--------|----------|--------|------------|
| 1 | 506 | 100 | 60/90 | Hoạt động |
| 2 | 602 | 100 | 22/90 | Hoạt động |
| 3 | P64 | 100 | 0/90 | Hoạt động |
| 4 | 504 | 100 | 58/90 | Hoạt động |
| 5 | 604 | 100 | 58/90 | Hoạt động |
| 6 | 402-PH | 100 | 74/90 | Hoạt động |
| 7 | 403-PH | 100 | 59/90 | Hoạt động |
| 8 | TC203 | 100 | 0/90 | Hoạt động |
| 9 | TC202 | 100 | 0/90 | Hoạt động |
| 10 | 405-PH | 100 | 58/90 | Hoạt động |

Xếp lịch

Đang hiển thị: **Tuần 1 (07/08-13/08). Phòng 506**

Kéo thả để cập nhật thời gian, số tiết mỗi ca học. Kéo ra ngoài bảng để xóa bỏ Tải lại

| | Thứ Hai | Thứ Ba | Thứ Tư | Thứ Năm | Thứ Sáu | Thứ Bảy | Chủ Nhật |
|---------|---|--|--|--|--|---|----------|
| Tiết 1 | DHCT07-20231-01 (CTXH với người khuyết tật) | DHCT15-20231-03 (Chủ nghĩa xã hội khoa học) | | DHLD58-20231-05 (Pháp luật đại cương) | | DHNT24-20231-04 (Tiếng Anh du lịch 1) | |
| Tiết 2 | | | | | | | |
| Tiết 3 | | | | | | | |
| Tiết 4 | | | | | | | |
| Tiết 5 | DHCT14-20231-01 (CTXH với người bị AH bởi HIV) | DHCT15-20231-04 (Chủ nghĩa xã hội khoa học) | | DHLD58-20231-06 (Pháp luật đại cương) | | DHDL50-20231-01 (Du lịch cộng đồng) | |
| Tiết 6 | | | | | | | |
| Tiết 7 | | | | | | | |
| Tiết 8 | DHCT15-20231-01 (Chủ nghĩa xã hội khoa học) | DHLD58-20231-03 (Pháp luật đại cương) | DHLD58-20231-03 (Pháp luật đại cương) | DHNT22-20231-01 (Tiếng Anh du lịch 1) | DHNT24-20231-01 (Tiếng Anh du lịch 1) | DHDL11-20231-01 (Giao tiếp và lễ tân ngoại giao) | |
| Tiết 9 | | | | DHNT22-20231-02 (Tiếng Anh du lịch 1) | DHNT24-20231-02 (Tiếng Anh du lịch 1) | DHDL11-20231-02 (Giao tiếp và lễ tân ngoại giao) | |
| Tiết 10 | | | | | | | |
| Tiết 11 | DHCT15-20231-02 (Chủ nghĩa xã hội khoa học) | DHLD58-20231-02 (Pháp luật đại cương) | DHLD58-20231-04 (Pháp luật đại cương) | DHNT22-20231-03 (Tiếng Anh du lịch 1) | DHNT24-20231-03 (Tiếng Anh du lịch 1) | DHDL11-20231-03 (Giao tiếp và lễ tân ngoại giao) | |
| Tiết 12 | | | | | | | |
| Tiết 13 | | | | | | | |
| Tiết 14 | | | | | | | |
| Tiết 15 | | | | | | | |

Kéo vào đây hoặc ra ngoài bảng để xóa lịch học

Hình 19: Giao diện xếp thời khóa biểu thủ công

Để thêm một lớp vào thời khóa biểu, ta kéo lớp từ danh sách lớp chưa xếp đủ lịch và thả vào trong lịch tuần. Tương tự để thay đổi một lớp đã xếp lịch, ta kéo lớp từ tiết hiện tại sang một tiết học đang trống. Để xóa lịch, ta kéo lớp đã xếp từ trong lịch tuần ra bên ngoài hoặc vào ô xóa lịch. Các tùy chọn thêm/sửa/xóa lịch là:

- Chỉ áp dụng với tuần này
- Áp dụng từ tuần này về sau
- Áp dụng với tất cả các tuần

| TT | Tên lớp | Học phần | Phòng | Chuỗi lịch học hiện tại | Thay đổi |
|----|-----------------|---------------------------|-------|--|-------------------|
| 1 | DHCT15-20231-03 | Chủ nghĩa xã hội khoa học | 506 | Thứ Ba, tiết 1-3 Tuần áp dụng: 1, 2 | Thứ Tư, tiết: 1-3 |

☒ Chỉ áp dụng với tuần này (1)
☐ Áp dụng từ tuần này (1) về sau
☐ Áp dụng với toàn bộ chuỗi lịch học hiện tại

Xác nhận Hủy

Hình 20: Các thao tác xếp lịch

3.5 Kết luận

Chương này đã giới thiệu về mô hình tổng quan và các chứng năng chính của hệ thống ứng dụng xếp thời khóa biểu dựa trên các công việc xếp thời khóa biểu của các trường Đại học.

KẾT LUẬN

Đề án “**Ứng dụng trí tuệ nhân tạo hỗ trợ sắp xếp lịch học tại Học viện Công nghệ Bưu chính Viễn thông**” tập trung nghiên cứu bài toán xếp thời khoá biểu trong các trường Đại học nói chung và Học viện Công nghệ Bưu chính Viễn thông nói riêng, từ đó suy ra các vấn đề và các phương pháp giải quyết. Trọng tâm nghiên cứu của đề án là thuật toán di chuyển và các cách cải tiến. Việc xây dựng mô hình được thực hiện và đánh giá để chọn ra phương án phù hợp nhất với bài toán, từ đó là tiền đề để xây dựng ứng dụng xếp thời khoá biểu bán tự động áp dụng cho mô hình quản lý đào tạo số tại Học viện Công nghệ Bưu chính Viễn thông. Cụ thể đề án đã đạt được một số kết quả như sau:

- Tìm hiểu tổng quan bài toán xếp thời khoá biểu học chế tín chỉ
- Nắm được lý thuyết về giải thuật di truyền và cách xây dựng mô hình giải thuật di truyền cho bài toán xếp thời khoá biểu học chế tín chỉ
- Đánh giá cơ bản mô hình giải thuật dựa trên dữ liệu thực tế
- Phân tích và xây dựng ứng dụng xếp thời khoá biểu bán tự động áp dụng mô hình giải thuật di truyền

Tuy nhiên đề án tốt nghiệp vẫn còn một số điểm hạn chế như chưa bổ sung nhiều ràng buộc mềm để tăng tính linh hoạt cho ứng dụng. Giải thuật cũng chưa được thực hiện đánh giá trên các bộ dữ liệu khác nên chưa có đủ góc nhìn về hiệu quả của giải thuật. Các vấn đề này cần được nghiên cứu thêm trong tương lai.

TÀI LIỆU THAM KHẢO

- [1] S. Lukas, A. Aribowo, và M. Muchri, “Solving timetable problem by genetic algorithm and heuristic search case study: universitas pelita harapan timetable”, *Real-World Appl. Genet. Algorithm*, vol 9, số p.h 4, tr 19–21, 2012.
- [2] J. S. Tan, S. L. Goh, G. Kendall, và N. R. Sabar, “A survey of the state-of-the-art of optimisation methodologies in school timetabling problems”, *Expert Syst. Appl.*, vol 165, tr 113943, tháng 3 2021, doi: 10.1016/j.eswa.2020.113943.
- [3] D. E. Goldberg và J. H. Holland, “Genetic Algorithms and Machine Learning”, *Mach. Learn.*, vol 3, số p.h 2, tr 95–99, tháng 10 1988, doi: 10.1023/A:1022602019183.
- [4] J. H. Holland, “Genetic algorithms”, *Sci. Am.*, vol 267, số p.h 1, tr 66–73, 1992.
- [5] S. Mirjalili, “Genetic Algorithm”, trong *Evolutionary Algorithms and Neural Networks*, vol 780, trong *Studies in Computational Intelligence*, vol. 780. , Cham: Springer International Publishing, 2019, tr 43–55. doi: 10.1007/978-3-319-93025-1_4.
- [6] J. Genlin, “Survey on genetic algorithm”, *Comput. Appl. Softw.*, vol 2, số p.h 1, tr 69–73, 2004.
- [7] E. Cantú-Paz và others, “A survey of parallel genetic algorithms”, *Calc. Paralleles Reseaux Syst. Repartis*, vol 10, số p.h 2, tr 141–171, 1998.
- [8] D. E. Goldberg và K. Deb, “A comparative analysis of selection schemes used in genetic algorithms”, trong *Foundations of genetic algorithms*, vol 1, Elsevier, 1991, tr 69–93.
- [9] D. E. Goldberg, “A note on Boltzmann tournament selection for genetic algorithms and population-oriented simulated annealing”, *Complex Syst.*, vol 4, tr 445–460, 1990.
- [10] B. L. Miller, D. E. Goldberg, và others, “Genetic algorithms, tournament selection, and the effects of noise”, *Complex Syst.*, vol 9, số p.h 3, tr 193–212, 1995.
- [11] R. Kumar, “Blending roulette wheel selection & rank selection in genetic algorithms”, *Int. J. Mach. Learn. Comput.*, vol 2, số p.h 4, tr 365, 2012.
- [12] G. Syswerda, “A study of reproduction in generational and steady-state genetic algorithms”, trong *Foundations of genetic algorithms*, vol 1, Elsevier, 1991, tr 94–101.
- [13] T. Blicke và L. Thiele, “A comparison of selection schemes used in evolutionary algorithms”, *Evol. Comput.*, vol 4, số p.h 4, tr 361–394, 1996.
- [14] E. Semenkin và M. Semenkina, “Self-configuring genetic algorithm with modified uniform crossover operator”, trong *Advances in Swarm Intelligence:*

- Third International Conference, ICSI 2012, Shenzhen, China, June 17-20, 2012 Proceedings, Part I 3*, Springer, 2012, tr 414–421.
- [15] X.-B. Hu và E. Di Paolo, “An efficient genetic algorithm with uniform crossover for the multi-objective airport gate assignment problem”, trong *2007 IEEE Congress on Evolutionary Computation*, IEEE, 2007, tr 55–62.
 - [16] S. Tsutsui, M. Yamamura, và T. Higuchi, “Multi-parent recombination with simplex crossover in real coded genetic algorithms”, trong *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation-Volume 1*, 1999, tr 657–664.
 - [17] T. Baeck, D. B. Fogel, và Z. Michalewicz, B.t.v, *Evolutionary Computation I: Basic Algorithms and Operators*, 0 a.b. CRC Press, 2018. doi: 10.1201/9781482268713.
 - [18] L. J. Eshelman, R. A. Caruana, và J. D. Schaffer, “Biases in the crossover landscape”, trong *Proceedings of the third international conference on Genetic algorithms*, 1989, tr 10–19.
 - [19] C. W. Ahn và R. S. Ramakrishna, “Elitism-based compact genetic algorithms”, *IEEE Trans. Evol. Comput.*, vol 7, số p.h 4, tr 367–385, 2003.
 - [20] M. M., R. A., và A. M., “Genetic Algorithm for Solving Course Timetable Problems”, *Int. J. Comput. Appl.*, vol 124, số p.h 10, tr 1–7, tháng 8 2015, doi: 10.5120/ijca2015905408.
 - [21] D. Zhang, Y. Liu, R. M’Hallah, và S. C. H. Leung, “A simulated annealing with a new neighborhood structure based algorithm for high school timetabling problems”, *Eur. J. Oper. Res.*, vol 203, số p.h 3, tr 550–558, tháng 6 2010, doi: 10.1016/j.ejor.2009.09.014.
 - [22] M. El-Sherbiny và Y. Ibrahim, *An Artificial Immune Algorithm with Alternative Mutation Methods: Applied to the Student Project Assignment Problem*. 2012.
 - [23] Phạm Anh Tuấn, “Ứng dụng giải thuật di truyền để xếp thời khóa biểu hệ tín chỉ cho trường Đại học”, 2012.
 - [24] A. O. Modupe, O. E. Olusayo, và O. S. Olatunde, “Development of a University Lecture Timetable using Modified Genetic Algorithms Approach”, 2014. [Online]. Available at: <https://api.semanticscholar.org/CorpusID:32481576>