

ĐỒ ÁN  
TỐT NGHIỆP ĐẠI HỌC  
NGÀNH CÔNG NGHỆ THÔNG TIN

Xây dựng và thử nghiệm thuật toán quản lý tiến độ thanh toán dự án sử dụng giải thuật di truyền và cân bằng NASH

Sinh viên thực hiện :

**Bạch Văn Hải** - 20101464  
Lớp CNTT01 - K55

Giảng viên hướng dẫn :

**PGS. Huỳnh Quyết Thắng**

HÀ NỘI, 5/2015

# Mục lục

Lời mở đầu	5
Lời cảm ơn	6
Phiếu giao nhiệm vụ đồ án tốt nghiệp	7
<b>1 Cơ sở lý luận về lý thuyết trò chơi và cân bằng Nash</b>	<b>8</b>
1.1 Giới thiệu về lý thuyết trò chơi	8
1.2 Biểu diễn trò chơi	8
1.2.1 Dạng chuẩn tắc	8
1.2.2 Dạng mở rộng	9
1.3 Các loại trò chơi	10
1.3.1 Trò chơi hợp tác và trò chơi không hợp tác	10
1.3.2 Trò chơi đối xứng và bất đối xứng	10
1.3.3 Trò chơi tổng bằng không và tổng khác không	10
1.3.4 Trò chơi đồng thời và trò chơi tuần tự	10
1.3.5 Trò chơi thông tin hoàn hảo, thông tin không hoàn hảo	11
1.3.6 Các trò chơi dài vô tận	11
1.4 Cân bằng Nash	11
<b>2 Kết hợp giữa giải thuật di truyền và cân bằng Nash</b>	<b>13</b>
2.1 Các chiến lược trong game và cân bằng Nash	13
2.2 Giải thuật di truyền	14
2.2.1 Nhiễm sắc thể	14
2.2.2 Các toán tử di truyền	14
2.2.3 Các bước cơ bản của giải thuật di truyền	15
2.2.4 Hàm mục tiêu	15
2.2.5 Điều kiện dừng của thuật toán	16
2.3 Tích hợp lý thuyết trò chơi và giải thuật di truyền: Sự chọn lọc Nash	16
2.4 Ví dụ hàm thích nghi và hàm mục tiêu cho mục tiêu lập lịch	17
<b>3 Ứng dụng lý thuyết trò chơi vào quản trị tiến độ thanh toán dự án</b>	<b>19</b>
3.1 Một số khái niệm cần biết trong kinh tế	19
3.2 Đặc tả vấn đề	20
3.3 Mô hình hóa trò chơi và cân bằng Nash	21
3.3.1 Mạng dự án(Project Network)	21
3.3.2 Các ràng buộc nguồn lực(Resource Constraints)	21
3.3.3 Đa cách thức(Multimodal mode)	21
3.3.4 Cách thức mô hình hóa việc thanh toán dự án	21
3.3.5 Tối ưu hóa mục tiêu	24
3.4 Giải thuật di truyền cho giải pháp cân bằng Nash	25
3.4.1 Mô hình chung của thiết kế giải thuật di truyền	25
3.4.2 Thiết kế giải thuật di truyền cho chiến lược đầu tiên	27

3.4.3	Thiết kế giải thuật di truyền cho chiến lược thứ hai . . . . .	28
3.4.4	Giá trị lý tưởng cho nhà thầu và khách hàng . . . . .	31
3.4.5	Lựa chọn thông số . . . . .	31
<b>4</b>	<b>Xây dựng và thử nghiệm thuật toán tối ưu hóa quản lý thanh toán dự án</b>	<b>33</b>
4.1	Xây dựng và thử nghiệm mô hình . . . . .	33
4.1.1	Kịch bản mô phỏng . . . . .	33
4.1.2	Các tham số của thuật toán di truyền . . . . .	36
4.1.3	Bộ dữ liệu đầu vào . . . . .	36
4.2	Đánh giá kết quả . . . . .	38
	<b>Kết luận và định hướng phát triển</b>	<b>41</b>
	<b>Tài liệu tham khảo</b>	<b>42</b>

# Danh sách hình vẽ

3.1	Sự tương tác của nhà thầu và khách hàng . . . . .	22
3.2	Trao đổi chéo . . . . .	26
3.3	Ví dụ về toán tử lai ghép chéo hai điểm . . . . .	28
3.4	Ví dụ về toán tử lai ghép MCUOX . . . . .	30
3.5	Các tham số giải thuật sử dụng trong chiến lược ban đầu . . . . .	32
3.6	Các tham số giải thuật sử dụng trong chiến lược thứ hai . . . . .	32
4.1	Mạng hoạt động Activity network. . . . .	33
4.2	Các thông số tương ứng với các activity . . . . .	34
4.3	Sơ đồ mạng dự án 1 . . . . .	36
4.4	Bảng các thông số của các activity của mạng dự án 1 . . . . .	37
4.5	Sơ đồ mạng dự án 2 . . . . .	37
4.6	Bảng các thông số của các activity của mạng dự án 2 . . . . .	38

# Lời mở đầu

Có thể thấy rằng, việc lập một kế hoạch thanh toán dự án là một trong những vấn đề quan trọng, nhận được nhiều sự quan tâm lớn về cả lý thuyết và thực hành trong quản lý dự án. Tuy nhiên các nghiên cứu trước đó chỉ đơn phương tìm hiểu cách tối ưu hóa tiến độ thanh toán dự án cho nhà đầu tư, nhà phát triển hoặc khách hàng. Trong một cái nhìn thực tế hơn, quá trình ra quyết định trong lập kế hoạch thanh toán dự án phải đáp ứng yêu cầu chấp nhận được của cả hai bên: nhà đầu tư và khách hàng. Nhà đầu tư, nhà phát triển mong muốn việc thanh toán dự án xảy ra càng muộn càng tốt, trong khi đó, khách hàng lại nỗ lực để có được các khoản cần thanh toán càng sớm càng tốt. Vì vậy, cần thiết có một cách thức để giải quyết vấn đề này nhằm đáp ứng yêu cầu của cả hai bên. Trong bài cáo đồ án tốt nghiệp này, em xin đề xuất và nghiên cứu về vấn đề "Tối ưu tiến độ thanh toán dự án dựa trên lý thuyết trò chơi".

Về cơ bản nội dung bài báo cáo đồ án tốt nghiệp được chia ra làm 4 chương:

- **Chương 1- Cơ sở lý luận về lý thuyết trò chơi và cân bằng Nash:** Cho cách nhìn khái quát về các lý thuyết trò chơi như cách biểu diễn, các loại trò chơi và đặc biệt là khái niệm về cân bằng Nash
- **Chương 2 - Kết hợp giữa giải thuật di truyền và cân bằng Nash:** Cho biết các lý thuyết, các bước cơ bản trong giải thuật di truyền; các chiến lược trong game và cách thức để tích hợp giải thuật di truyền với cân bằng Nash.
- **Chương 3 - Ứng dụng lý thuyết trò chơi vào quản trị tiến độ thanh toán dự án:** Đây chính là nội dung chính của đồ án tốt nghiệp khi đề cập đến cách thức mô hình hóa bài toán, đưa ra các chiến lược, giải pháp cụ thể để triển khai giải thuật di truyền.
- **Chương 4 - Xây dựng và thử nghiệm mô hình thanh toán dự án:** Đưa ra kịch bản mô phỏng, lựa chọn dữ liệu đầu vào, các tham số trong giải thuật; Đánh giá các kết quả thu được và đưa ra nhận xét, kết luận tổng quát.

Hà Nội, Ngày 14 Tháng 5 Năm 2015  
Sinh viên

Bạch Văn Hải

# Lời cảm ơn

Trong suốt quá trình học tập và thực hiện đồ án tốt nghiệp này, tôi đã nhận được đã được sự hướng dẫn nhiệt tình từ các thầy cô bạn bè và những người thân của mình. Tôi xin chân thành cảm ơn:

- Các thầy cô giáo trong trường, đặc biệt là các thầy cô trong viện công nghệ thông tin và truyền thông SoICT đã giúp đỡ và truyền đạt cho tôi những kiến thức bổ ích và tạo điều kiện thuận lợi cho tôi học tập và phát triển bản thân.
- PGS.TS Huỳnh Quyết Thắng đã giúp đỡ và hướng dẫn nhiệt tình trong suốt quá trình thực hiện đồ án tốt nghiệp này.
- Anh Trịnh Bảo Ngọc đã tư vấn, hỗ trợ tôi trong quá trình thực hiện đồ án này.
- Bạn bè và người thân đã động viên tôi trong quá trình làm đồ án.

Xin chân thành cảm ơn.

# Phiếu giao nhiệm vụ đồ án tốt nghiệp

## 1. Thông tin về sinh viên

- Họ và tên sinh viên: **Bạch Văn Hải**
- Điện thoại liên lạc: **01649612854**      Email: **haibachvan@gmail.com**
- Lớp: **Công nghệ thông tin 1**      Hệ đào tạo: **Đại học chính quy**
- Đồ án tốt nghiệp được thực hiện tại: **ĐH Bách Khoa Hà Nội**
- Thời gian làm ĐATN: Từ ngày **20/02/2015** đến **31/05/2015**

## 2. Mục đích nội dung của ĐATN

Áp dụng lý thuyết trò chơi trong việc tối ưu hóa tiến độ thanh toán dự án

## 3. Các nhiệm vụ của ĐATN

- Tìm hiểu và nắm bắt được các khái niệm liên quan đến Lý thuyết trò chơi và Cân bằng Nash.
- Nắm bắt giải thuật di truyền và cách tích hợp nó với cân bằng Nash
- Mô hình hóa được bài toán tối ưu về tiến độ thanh toán dự án, đưa ra kết quả tốt nhất đáp ứng yêu cầu của cả khách hàng và nhà đầu tư.
- Xây dựng thử nghiệm một sản phẩm theo mô hình đã đưa ra.

## 4. Lời cam đoan của sinh viên

Tôi – **Bạch Văn Hải** - cam kết ĐATN là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của **PGS.TS Huỳnh Quyết Thắng**.

Các kết quả nêu trong ĐATN là trung thực, không phải là sao chép toàn văn của bất kỳ công trình nào khác.

Hà Nội, Ngày 14 Tháng 5 Năm 2015  
Sinh viên

Bạch Văn Hải

## 5. Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành của ĐATN và cho phép bảo vệ:

Hà Nội, Ngày 31 Tháng 5 Năm 2015  
Giáo viên hướng dẫn

PGS.TS Huỳnh Quyết Thắng

# Chương 1

## Cơ sở lý luận về lý thuyết trò chơi và cân bằng Nash

(Tài liệu tham khảo: [http://en.wikipedia.org/wiki/Game\\_theory](http://en.wikipedia.org/wiki/Game_theory))

### 1.1 Giới thiệu về lý thuyết trò chơi

Lý thuyết trò chơi là một ngành chuyên nghiên cứu về việc đưa ra quyết định chiến lược. Lý thuyết trò chơi thường được mô tả như một lý thuyết trong toán học, nghiên cứu tình huống trong đó người chơi sẽ hành động theo những cách khác nhau để tối ưu hóa lợi ích của mình. Điều quan trọng nhất của lý thuyết này là phương pháp tiếp cận để đưa ra các quyết định để giải quyết một vấn đề nào đó. Hay nói cách khác là nó sẽ xác định xác suất thành công khi cho trước một không gian chiến lược.

Lý thuyết trò chơi được áp dụng sử dụng vào rất nhiều các ngành, lĩnh vực như Chính trị học, Đạo đức học, Kinh tế... và đặc biệt là Khoa học máy tính ứng dụng trong Trí tuệ nhân tạo và Điều khiển học. Lý thuyết trò chơi đã dần đóng vai trò ngày càng quan trọng trong logic và khoa học máy tính. Một số lý thuyết logic có cơ sở trong ngữ nghĩa trò chơi, mô phỏng các tính toán tương tác với nhau.

### 1.2 Biểu diễn trò chơi

Các trò chơi được nghiên cứu xem xét trong lý thuyết trò chơi là các đối tượng toán học được xác định rõ. Để xác định đầy đủ, một trò chơi phải xác định các yếu tố sau: các người chơi tham gia trò chơi; những thông tin và hành động có sẵn cho mỗi người chơi tại mỗi thời điểm quyết định (tập các chiến lược) và cơ chế thưởng phạt tương ứng với mỗi tổ hợp các chiến lược. Các yếu tố này thường được sử dụng cùng với một khái niệm giải pháp lựa chọn để suy ra một tập hợp các chiến lược cân bằng cho mỗi người chơi. Những chiến lược cân bằng xác định một trạng thái cân bằng của các trò chơi - một trạng thái ổn định, trong đó một trong hai kết quả xảy ra hoặc một loạt các kết quả xảy ra với xác suất đã biết.

#### 1.2.1 Dạng chuẩn tắc

Một trò chơi dạng chuẩn tắc:

	Người chơi 2 chọn cột trái	Người chơi 2 chọn cột phải
Người chơi 1 chọn hàng trên	4, 3	-1, -1
Người chơi 1 chọn hàng dưới	0, 0	3, 4



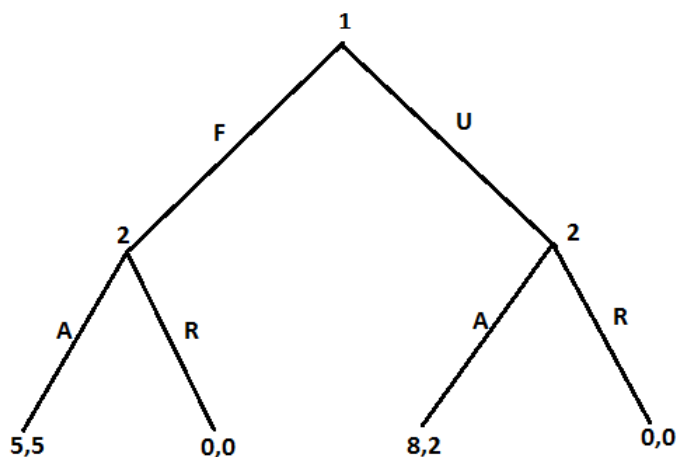
Trò chơi chuẩn tắc(hay trò chơi chiến lược) thường được biểu diễn bằng một ma trận cho biết thông tin về các người chơi, chiến lược và cơ chế thưởng phạt. Nói chung, nó có thể biểu diễn bằng bất kì hàm nào đó mà liên kết giữa cơ chế thưởng phạt với tổ hợp các chiến lược, hành động tương ứng. Lấy một ví dụ: có hai người chơi, một người chọn hàng, một người chọn cột. Mỗi người chơi có hai chiến lược, mỗi chiến lược được biểu diễn một ô được xác định bởi số hiệu hàng và số hiệu cột của nó. Mức độ thưởng phạt được ghi trong ô đó. Giá trị thứ nhất là mức thưởng phạt cho đầu người chơi theo hàng; giá trị thứ hai là mức thưởng phạt tương ứng với người chơi theo cột. Giả sử người chơi 1 chọn hàng trên và người chơi 2 chọn cột trái. Khi đó, người chơi 1 nhận được 4 điểm và người chơi 2 nhận được 3 điểm.

Khi một trò chơi được biểu diễn bằng dạng chuẩn tắc, nó được coi rằng mỗi người chơi hành động một cách đồng thời hoặc ít nhất không biết hành động của người kia. Nếu mỗi người chơi có thông tin về sự lựa chọn của người chơi khác thì trò chơi thường được biểu diễn theo kiểu mở rộng.

Mỗi dạng trò chơi mở rộng có một dạng trò chơi chuẩn tắc tương ứng. Tuy nhiên, việc chuyển đổi từ hình thức chuẩn tắc sang dạng mở rộng có thể dẫn đến việc biểu diễn và tính toán trong thực tế trở nên khó khăn hơn.

### 1.2.2 Dạng mở rộng

Các trò chơi dạng mở rộng có thể được sử dụng để hợp thức hóa các trò chơi với một trình tự thời gian của các lượt đi. Các trò chơi sẽ được biểu diễn bằng cây. Mỗi đỉnh (hoặc nút) biểu diễn một điểm mà người chơi có thể lựa chọn. Người chơi được xác định bằng một số ghi cạnh đỉnh. Các đoạn thẳng đi ra từ đỉnh đó biểu diễn các hành động có thể cho người chơi đó. Mức thưởng phạt được ghi rõ tại đáy cây. Các dạng mở rộng có thể được xem như là một sự tổng quát nhiều người chơi của một cây quyết định. Trong trò chơi trong hình gồm có hai người chơi. Người chơi 1 đi trước



và chọn  $F$  hoặc  $U$ . Người chơi 2 nhìn thấy nước đi của người chơi 1 và chọn  $A$  hoặc  $R$ . Sau khi hai người chơi đã đưa ra sự lựa chọn của mình thì trò chơi coi như được hoàn thành và mỗi người chơi sẽ nhận được cơ chế thưởng phạt tương ứng. Giả sử người chơi 1 chọn  $U$  và sau đó người chơi 2 chọn  $A$ . Khi đó, người chơi 1 được 8 điểm và người chơi 2 được 2 điểm.

Các trò chơi mở rộng còn có thể mô tả các trò chơi đi đồng thời và các trò chơi với những thông tin không hoàn hảo. Để biểu diễn nó, sử dụng hoặc là một đường chấm chấm nối hai đỉnh khác nhau hoặc một đường tròn vẽ quanh hai đỉnh khác nhau để biểu diễn rằng chúng đều thuộc cùng một tập hợp thông tin.

## 1.3 Các loại trò chơi

### 1.3.1 Trò chơi hợp tác và trò chơi không hợp tác

Trò chơi hợp tác là trò chơi mà giữa các người chơi có hình thành các cam kết ràng buộc. Còn trong trò chơi không hợp tác thì điều này không xảy ra. Thông thường thì sự trao đổi, giao tiếp giữa các người chơi được cho phép trong trò chơi hợp tác. Trong hai loại trò chơi, trò chơi không hợp tác có thể mô hình hóa tình huống đến các chi tiết nhỏ nhất và cho các kết quả chính xác. Trò chơi hợp tác tập trung vào các trò chơi có độ phức tạp lớn. Một nỗ lực đáng kể đã được thực hiện nhằm liên kết giữa hai cách tiếp cận. Chương trình Nash đã xây dựng những giải pháp hợp tác là các điểm cân bằng không hợp tác. Trò chơi lai là trò chơi chứa các thành phần hợp tác và không hợp tác.

### 1.3.2 Trò chơi đối xứng và bất đối xứng

Một trò chơi đối xứng là một trò chơi mà cơ chế thưởng phạt ứng với một chiến lược cụ thể nào đó chỉ phụ thuộc vào chiến thuật được sử dụng, chứ không phụ thuộc vào người nào đang chơi. Nếu như tính danh của những người chơi có thể thay đổi mà không làm thay đổi phần lợi đối với chiến thuật chơi thì một trò chơi là đối xứng. Các bài toán song đề tù nhân, săn hươu... đều là các trò chơi đối xứng.

	E	F
E	1, 2	0, 0
F	0, 0	1, 2

Đa số những trò chơi bất đối xứng được nghiên cứu là những trò chơi mà tập hợp chiến thuật khác nhau được sử dụng bởi hai người chơi. Ví dụ trò chơi tối hậu thư và trò chơi kẻ độc tài... là các trò chơi không đối xứng vì mỗi người chơi có các chiến lược khác nhau. Tuy nhiên, trong một vài trường hợp trò chơi vẫn có chiến lược giống nhau cho cả hai người chơi nhưng vẫn là trò chơi bất đối xứng.

### 1.3.3 Trò chơi tổng bằng không và tổng khác không

Trò chơi tổng bằng không là một trường hợp đặc biệt của trò chơi tổng khác không, trong đó sự lựa chọn của người chơi không làm thay đổi các nguồn lực có sẵn. Trong trò chơi tổng bằng không, với mọi tổ hợp của các chiến lược chơi, tổng điểm của tất cả các người chơi trong ván chơi luôn bằng 0. Nói một cách không chính thức, đầu thủ này hưởng lợi từ thiệt hại của đối thủ khác. Các loại cờ cổ điển như cờ vây, cờ vua và cờ tướng cũng là các trò chơi tổng bằng 0

	A	B
A	-1, 1	3, -3
B	0, 0	-2, 2

Nhiều trò chơi mà các nhà lý thuyết trò chơi nghiên cứu, trong đó có song đề tù nhân là các trò chơi tổng khác không, do có một số kết cục có tổng kết quả lớn hơn hoặc nhỏ hơn 0. Nói một cách không chính thức, trong các trò chơi tổng khác 0 thì một lợi ích của người chơi này không nhất thiết phải tương ứng với thiệt hại của người chơi kia. Có thể biến đổi một trò chơi bất kỳ thành một trò chơi tổng bằng 0 bằng cách bổ sung một người chơi bù nhìn sao cho thiệt hại của người chơi bù nhìn này bù lại tổng thu hoạch của các người chơi khác.

### 1.3.4 Trò chơi đồng thời và trò chơi tuần tự

Trò chơi đồng thời là trò chơi mà cả hai người chơi di chuyển đồng thời hoặc nếu nếu không di chuyển cùng lúc thì người chơi đi sau không thể xác định được những hành động của người chơi đã đi trước. Trò chơi tuần tự là trò chơi mà người chơi mà người chơi sau có một số thông tin về hành

động trước đó. Thông tin này không cần phải là thông tin đầy đủ về mọi hành động của người chơi trước đó. Ví dụ, người chơi 1 có thể biết rằng người chơi 2 đã thực hiện những hành động cụ thể nào trước đó nhưng người chơi đó lại không biết được những hành động có sẵn khác của người chơi 2

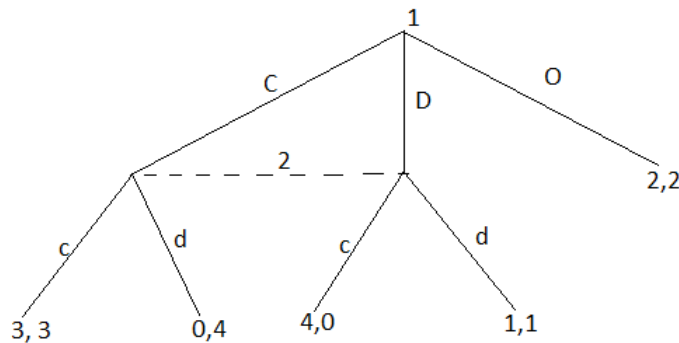
So sánh trò chơi đồng thời và trò chơi tuần tự:

	Trò chơi tuần tự	Trò chơi đồng thời
Biểu diễn bằng	Cây quyết định	Ma trận thưởng phạt
Trục thời gian	Có	Không
Còn được gọi là	Trò chơi mở rộng	Trò chơi chiến lược

Biểu diễn dạng chuẩn tắc được dùng để biểu diễn các trò chơi đồng thời và biểu diễn dạng mở rộng được dùng cho các trò chơi tuần tự.

### 1.3.5 Trò chơi thông tin hoàn hảo, thông tin không hoàn hảo

Các trò chơi thông tin hoàn hảo lập thành một tập con quan trọng của các trò chơi tuần tự. Một trò chơi được gọi là có thông tin hoàn hảo nếu mọi người chơi biết tất cả các nước đi của người chơi khác đã thực hiện. Do vậy chỉ có các trò chơi tuần tự mới có thể là các trò chơi thông tin hoàn hảo. Hầu hết các trò chơi nghiên cứu trong lý thuyết trò chơi là các trò chơi có thông tin không hoàn hảo, tuy một số trò chơi hay như cờ vây, cờ vua lại là trò chơi thông tin hoàn hảo.



Tính chất thông tin hoàn hảo thường bị nhầm lẫn với khái niệm về thông tin đầy đủ. Tính chất thông tin đầy đủ đòi hỏi rằng mỗi người chơi biết về các chiến lược và thành quả thu được của các người chơi khác, nhưng không nhất thiết về các hành động của họ.

### 1.3.6 Các trò chơi dài vô tận

Bởi các lí do hiển nhiên, các trò chơi được nghiên cứu bởi các nhà kinh tế gia và những người chơi trong thế giới thực nhìn chung là kết thúc trò chơi trong hữu hạn các bước đi. Các nhà toán học lý thuyết không bị cản trở bởi điều đó và lý thuyết về tập hợp đặc biệt nghiên cứu về các trò chơi kết thúc sau vô hạn các bước đi bởi người thắng là không biết được cho đến sau khi các bước đó hoàn thành.

Sự chú ý thường không phải là quá nhiều về cách nào tốt nhất để chơi trò chơi mà đơn giản là chỉ phụ thuộc vào người chơi hay người kia có hay không một chiến thuật chiến thắng.

## 1.4 Cân bằng Nash

Cân bằng Nash xác định một chiến lược tối ưu cho các trò chơi khi chưa có điều kiện tối ưu nào được xác định trước đó. Định nghĩa cơ bản của cân bằng Nash: Nếu tồn tại một tập hợp các chiến

lược cho một trò chơi với đặc tính là không một đối thủ nào có thể hưởng lợi bằng cách thay đổi chiến lược hiện tại của mình khi các đối thủ khác không thay đổi, tập hợp các chiến lược đó và phần thu nhận tương ứng tạo nên cân bằng Nash. Nói cách khác, cân bằng Nash đạt được nếu như thay đổi một cách đơn phương của bất cứ ai trong số các đối thủ cũng sẽ làm cho chính người đó thu lợi ít hơn mức có được với chiến lược hiện tại. Khái niệm này áp dụng cho những trò chơi gồm từ hai đối thủ trở lên và Nash đã chỉ ra rằng tất cả các khái niệm khác nhau về giải pháp (solution) trong các trò chơi được đưa ra trước đó đều có cân bằng Nash.

Một ví dụ đơn giản: trong một trò chơi gồm hai đối thủ cùng chọn song song một số bất kỳ từ 0 đến 10. Người nào chọn số lớn hơn sẽ thua và phải trả tiền cho người kia. Trò chơi này chỉ có 1 cân bằng Nash duy nhất: cả hai đối thủ đều chọn số 0. Bất kỳ sự lựa chọn nào khác (mà không biết sự lựa chọn của đối thủ kia) cũng có thể làm đối thủ thua cuộc. Khi thay đổi luật chơi: mỗi đấu thủ sẽ được hưởng số tiền bằng con số mà cả hai cùng chọn, nếu không chọn trùng nhau thì không ai có tiền, ta sẽ có 11 cân bằng Nash.

Một trò chơi có thể có nhiều hoặc không có cân bằng Nash. Nash cũng chứng minh rằng nếu cho phép các chiến lược hỗn hợp (mixed strategies) tức là các đối thủ chọn ngẫu nhiên các chiến lược dựa vào khả năng đã được ấn định trước, thì bất cứ một trò chơi với  $n$  đối thủ nào trong đó mỗi đối thủ có thể chọn trong giới hạn cho trước nhiều chiến lược sẽ có ít nhất 1 cân bằng Nash của các chiến lược hỗn hợp.

Cân bằng Nash giúp làm rõ sự phân biệt giữa các trò chơi hợp tác và không hợp tác. Trong các trò chơi không hợp tác không tồn tại cơ chế thỏa thuận, vì thế chỉ có thỏa thuận cân bằng mới được duy trì. Một hướng lý thuyết trò chơi mới được mở đường bằng cân bằng Nash xóa bỏ sự phân biệt này bằng cách xóa bỏ các cơ chế áp đặt có liên quan trong mô hình trò chơi, từ đó các trò chơi được mô hình hóa với tính chất không hợp tác.

## Chương 2

# Kết hợp giữa giải thuật di truyền và cân bằng Nash

*Tài liệu tham khảo:*

[1] Tài liệu học viện bưu chính viễn thông. "*Giải thuật di truyền*"

[6] Massimo Orazio Spata and Salvatore Rinaudo. "*Merging nash equilibrium solution with Genetic algorithm: the game genetic algorithm*"

### 2.1 Các chiến lược trong game và cân bằng Nash

Một chiến lược trò chơi dạng  $G$  với hai người chơi là  $(X, Y, E, h)$  trong đó:

- $X, Y, E$  là các tập hợp.
- $h$  là một hàm được định nghĩa trong  $X \times Y$
- $X$  đặc tả lại các lựa chọn có thể của người chơi 1,  $Y$  đặc tả lại các lựa chọn có thể của người chơi 2.
- $E$  đặc tả những chiến lược có thể được trong trò chơi.
- $h$  là một hàm đầu ra cung cấp các kết quả đạt được dựa trên sự lựa chọn của người chơi. Vì vậy nếu người chơi 1 chọn  $x$  và người 2 chọn  $y$  thì có  $h(x, y)$  kết quả.

Trong game  $G$ , chúng ta cần biết đến sự ưu tiên của người chơi cho các phần tử khác nhau của  $E$ . Để nhanh và đơn giản cho việc mô tả lại sự ưu tiên đó, ta sử dụng hàm tiện ích (utility function)  $u(x)$ . Lấy ví dụ, giả định người chơi 1 được đưa ra một hàm  $u$  được xác định trong  $E$  và có giá trị trong  $R$ .

Có một trò chơi dạng  $(X, Y, E, h)$  và hai hàm tiện ích (cho cả người chơi)  $(w, v)$  đây là dạng biểu thức của game:

$$(X, Y, E, h, w, v)$$

Cân bằng Nash cho  $G = (X, Y, f_1, f_2)$  là  $(x^*, y^*)$  thuộc  $X \times Y$ :

- $f_1(x^*, y^*) \geq f_1(x, y^*)$  mọi  $x$  thuộc  $X$
- $f_2(x^*, y^*) \geq f_2(x^*, y)$  mọi  $y$  thuộc  $Y$

## 2.2 Giải thuật di truyền

Các quy tắc di truyền trong chọn lọc tự nhiên cung cấp cho hướng tiến hóa sinh học của sinh vật để thích ứng và đặc điểm của môi trường. các nhân tố tốt nhất được lựa chọn thông qua các đặc điểm thích nghi môi trường, từ đó chúng ta có khả năng tốt hơn trong việc tồn tại và tái sản xuất. Lựa chọn này phát triển từ thế hệ này sang thế hệ khác cho phép tăng sự thích nghi trung bình quần thể.

Mỗi giải pháp có thể có một vấn đề được biểu diễn như là một cá nhân(gọi là agent), có sức mạnh là chất lượng của các giải pháp. Bắt đầu từ một quần thể của cá thể hoặc giải pháp, cơ chế tiến hóa được áp dụng định kỳ để tìm ra giải pháp tốt hơn cho vấn đề.

Chúng ta tập trung sự chú ý về việc thiết kế các thuật toán này cung cấp mã giả. Cách hoạt động cơ bản để bắt đầu các thuật toán được tạo ra quần thể ban đầu. Quần thể bao gồm một tập hợp các thực thể, mỗi trong số đó tương ứng với một giải pháp cho vấn đề. Quần thể đại diện cho toàn bộ thuật toán tìm kiếm không gian, có thể được tạo ra bằng nhiều cách. Nói chung, chúng ta có thể lấy con đường thế hệ ngẫu nhiên, mà là cung cấp cho các thuật toán bằng cách tạo ra các kích thước tối đa của quần thể, do đó tương ứng với số lượng các thực thể mà quần thể cần có. Trong phương pháp tiếp cận này, làm việc chủ yếu trên bit các đối tượng: điều này có nghĩa rằng thuật toán tạo ra các chuỗi nhị phân ngẫu nhiên để xử lý. Một khi các thuật toán đã hoàn thành công việc của mình, các thực thể được lựa chọn như là Giải pháp tốt nhất là chuyển đổi thành một mô hình đối tượng để giải thích các giải pháp được tìm thấy. Một phương pháp thứ hai cho thế hệ mới trong quần thể có thể được thiết lập từ một tập dữ liệu. Trong phương pháp này, các thuật toán có thể cung cấp một nguồn dữ liệu tương ứng với các đối tượng mô hình quy định trên, và nó sẽ được để xác minh tính chính xác của các số liệu và chuyển đổi cùng thành chuỗi nhị phân. Nó có thể định nghĩa khác nhau thuật toán cho các loại khác nhau của sự chọn lọc di truyền cho phép các thuật toán GA. Các giải pháp đề xuất là việc chọn lọc theo cân bằng Nash của Lý thuyết trò chơi.

### 2.2.1 Nhiễm sắc thể

Giải thuật di truyền GA sử dụng một số thuật ngữ của ngành di truyền học như: Nhiễm sắc thể, Quần thể, Gen... Nhiễm sắc thể(NST) được tạo thành từ các Gen(được biểu diễn của một chuỗi tuyến tính). Mỗi gen mang một số đặc trưng và có vị trí nhất định trong NST. Mỗi NST sẽ biểu diễn một lời giải của bài toán.

### 2.2.2 Các toán tử di truyền

#### Toán tử sinh sản

Toán tử sinh sản bao gồm hai quá trình: quá trình sinh sản(phép tái sinh), quá trình chọn lọc(phép chọn).

*Phép tái sinh* là quá trình các NST được sao chép trên cơ sở độ thích nghi. Độ thích nghi là một hàm được gán giá trị thực, tương ứng với mỗi NST trong quần thể. Quá trình này được mô tả:

- Xác định độ thích nghi của từng NST trong quần thể ở thế hệ thứ  $t$ , lập bảng cộng dồn các giá trị thích nghi(theo thứ tự gán cho từng NST). Giả sử, quần thể  $P$  có  $n$  cá thể. Gọi độ thích nghi của NST  $i$  tương ứng là  $f_i$  tổng cộng dồn thứ  $i$  là  $f_{ti}$  được xác định bởi:

$$f_{ti} = \sum_{j=1}^t f_j$$

- Gọi  $F_n$  là tổng độ thích nghi của toàn quần thể. Chọn một số ngẫu nhiên  $f$  trong khoảng 0 tới  $F_n$ . Chọn cá thể thứ  $k$  đầu tiên thỏa mãn  $f \geq f_{tk}$  đưa vào quần thể mới.

*Phép chọn (Selection)* là quá trình loại bỏ NST kém thích nghi trong quần thể. Quá trình này được mô tả như sau:

- Sắp xếp quần thể theo thứ tự mức độ thích nghi giảm dần.
- Loại bỏ các NST ở cuối dãy. Giữ lại  $n$  cá thể tốt nhất.

## Toán tử ghép chéo

Ghép chéo là quá trình tạo NST mới trên cơ sở các NST cha-mẹ bằng cách ghép một đoạn trên NST cha-mẹ với nhau. Toán tử ghép chéo được gán với một xác suất là  $pc$ . Quá trình được mô tả như sau:

- Chọn ngẫu nhiên một cặp NST(cha-mẹ) trong quần thể. Giả sử, NST cha-mẹ có cùng độ dài  $m$ .
- Tạo một số ngẫu nhiên trong khoảng từ 1 tới  $m - 1$  (gọi là điểm ghép chéo). Điểm ghép chéo chia NST cha-mẹ thành hai chuỗi con có độ dài là  $m_1$  và  $m_2$ . Hai chuỗi con mới được tạo thành là  $m_{11} + m_{22}$  và  $m_{21} + m_{12}$
- Đưa hai NST mới vào quần thể.

## Toán tử đột biến

Đột biến là hiện tượng NST con mang một số đặc tính không có trong mã di truyền của cha mẹ.

- Chọn ngẫu nhiên một NST trong quần thể.
- Tạo một số ngẫu nhiên  $k$  trong khoảng từ 1 tới  $m$ ,  $1 \leq k \leq m$
- Thay đổi bit thứ  $k$ . Đưa NST này vào quần thể để tham gia quá trình tiến hóa ở thế hệ tiếp theo.

### 2.2.3 Các bước cơ bản của giải thuật di truyền

Một giải thuật di truyền đơn giản bao gồm 6 bước:

- **Bước 1:** Khởi tạo một quần thể ban đầu gồm các chuỗi NST.
- **Bước 2:** Xác định mục tiêu cho từng NST tương ứng.
- **Bước 3:** Tạo các NST mới dựa trên thuật toán di truyền
- **Bước 4:** Loại bớt các NST có độ thích nghi thấp.
- **Bước 5:** Xác định hàm mục tiêu cho các NST mới và đưa vào quần thể.
- **Bước 6:** Kiểm tra thỏa mãn điều kiện dừng. Nếu điều kiện đúng, lấy ra NST tốt nhất, giải thuật dừng lại; ngược lại quay về bước 3.

### 2.2.4 Hàm mục tiêu

Sau mỗi thế hệ được hình thành, cần tính lại độ thích nghi cho từng cá thể để chuẩn bị cho một thế hệ mới. Do số lượng các cá thể tăng lên, độ thích nghi giữa các cá thể không có sự chênh lệch đáng kể. Do đó, các cá thể có độ thích nghi cao chưa hẳn chiếm ưu thế trong thế hệ tiếp theo. Vì vậy, cần ấn định tỷ lệ đối với hàm thích nghi nhằm tăng khả năng cho các NST đạt độ thích nghi cao. Có 3 cơ chế định tỷ lệ: Định tỷ lệ tuyến tính, Phép cắt Sigma, Định tỷ lệ cho luật dạng lũy thừa.

#### Định tỷ lệ tuyến tính

Độ thích nghi được xác định theo công thức:

$$f'_i = a * f_i + b$$

Cần chọn các tham số  $a, b$  sao cho độ thích nghi trung bình được ánh xạ vào chính nó. Tăng độ thích nghi tốt nhất bằng cách nhân nó với độ thích nghi trung bình. Cơ chế này tạo ra các giá trị âm cần xử lý riêng. Ngoài ra các tham số  $a, b$  thường gắn với quần thể và không phụ thuộc vào bài toán.

## Phép cắt Sigma

Phương pháp này được thiết kế vừa để cải tiến phương pháp định tỷ lệ tuyến tính vừa để xử lý các giá trị âm, vừa kết hợp thông tin mà bài toán phụ thuộc. Ở đây, độ thích nghi mới được tính theo công thức:

$$f'_i = f_i + (\bar{f} - c * \sigma)$$

trong đó  $c$  là một số nguyên nhỏ (thường lấy giá trị từ 1 tới 5);  $\sigma$  là độ lệch chuẩn của quần thể. Với giá trị âm thì  $f'$  được thiết lập bằng 0.

## Định tỷ lệ cho luật dạng lũy thừa

Trong phương pháp này, độ thích nghi lúc khởi tạo có năng lực đặc biệt:

$$f'_i = f_i^k$$

với  $k$  gần bằng 1. Tham số  $k$  định tỷ lệ hàm  $f$ . Tuy nhiên, một số nhà nghiên cứu cho rằng nên chọn  $k$  độc lập với bài toán. Bằng thực nghiệm cho thấy nên chọn  $k = 1.005$ .

### 2.2.5 Điều kiện dừng của thuật toán

Có hai loại điều kiện dừng cơ bản. Các điều kiện này dùng các đặc trưng tìm kiếm để quyết định ngừng quá trình tìm kiếm.

Dựa trên cấu trúc NST do sự hội tụ của quần thể bằng cách kiểm soát số alen được hội tụ, ở đây alen được coi như hội tụ nếu một số phần trăm quần thể đã được định trước có cùng (hoặc tương đương đối với các biểu diễn không nhị phân) giá trị trong alen này. Nếu số alen hội tụ vượt quá số phần trăm nào đó của tổng số alen, việc tìm kiếm sẽ kết thúc.

Dựa trên ý nghĩa đặc biệt của một NST: đo tiến bộ của giải thuật trong một số thế hệ cho trước. Nếu tiến bộ này nhỏ hơn một hằng số xác định, kết thúc tìm kiếm.

## 2.3 Tích hợp lý thuyết trò chơi và giải thuật di truyền: Sự chọn lọc Nash

Ban đầu cá thể trong quần thể  $P$  được lựa chọn theo tiêu chí của cân bằng Nash thông qua các hàm thích nghi. Ví dụ giá trị hàm thích nghi của tất cả các người chơi sẽ được đặt trong một ma trận  $n \times n$ , nơi nó được tính bằng cân bằng Nash. Sau đó, NST tốt nhất sẽ được sao chép vào quần thể mới. Phương pháp này có thể phát triển một cách nhanh chóng bởi vì hiệu suất của thuật toán di truyền lọc các giải pháp tốt nhất. Dưới đây là mã giả của thuật toán:



```

procedure NashSelection(P)
begin
    for all  $n$  player agents of  $P(t)$  do
         $fk = \text{fitness value of player agent } k$ 
    find Nash Equilibrium between the  $f_1 \dots f_n$  value
    of player agents
    all agents point of Equilibrium are selected
end

```

Hãy xem xét một ví dụ, với  $n = 2$  người chơi. Chiến lược  $X$  đặt ra cho người chơi 1 và chiến lược  $Y$  đặt ra cho người chơi 2.  $X$  tương ứng với các tập hợp con của các biến xử lý bởi người chơi 1 và tối ưu hóa cùng  $f_1$ .  $Y$  tương ứng với các tập hợp con của các biến xử lý bởi người chơi 2 và tối ưu cùng  $f_2$ . Trong trường hợp này, cân bằng Nash là giải pháp cho vấn đề tối ưu hóa mục tiêu kép. Do đó, theo quy tắc Nash Theory thì người chơi 1 chọn các tiêu chí đầu tiên  $f_1$  bằng cách sửa đổi  $X, Y$  khi được cố định bởi người chơi 2. Tương tự, người chơi 2 chọn các tiêu chí  $f_2$  bằng cách sửa đổi  $Y$  trong khi  $X$  được cố định bởi người chơi 1.

Các bước tiếp theo bao gồm trong việc tạo ra một quần thể  $P$  mới cho các hai người chơi. Tối ưu hóa nhiệm vụ của người chơi 1 được thực hiện thông qua hàm thích nghi  $f_1$  trong quần thể  $P_1$ , trong khi tối ưu hóa nhiệm vụ của người chơi 2 thực hiện thông qua hàm thích nghi  $f_2$  trong quần thể  $P_2$ .

Cân bằng Nash đạt được khi không người chơi nào có thể cải thiện được thêm các tiêu chí của nó nữa.

## 2.4 Ví dụ hàm thích nghi và hàm mục tiêu cho mục tiêu lập lịch

Giả sử rằng có  $m$  thiết bị  $M_i (i = 1, 2, \dots, m)$  nên sắp xếp  $n$  công việc  $J_j (j = 1, 2, \dots, n)$  với  $m < n$ , mỗi công việc  $J$  sẽ tạo ra một tác nhân. Đối với tất cả các công việc, một công việc được phân bổ khoảng thời gian để một hay nhiều thiết bị thực hiện, thường đặt tên là lịch trình. Một lịch trình nếu cho phép thì với mỗi thời điểm  $t$  một công việc được lên kế hoạch được lên kế hoạch một cách chính xác. Một lịch trình tối ưu sẽ làm giảm thiểu lỗi và làm tối đa hóa một tiêu chí nhất định: các hàm mục tiêu.

Chi phí cho mỗi công việc  $J_i$  là  $f_i$  thường được tính toán dựa trên việc xem xét khối lượng công việc  $w_j$  và thời gian hoàn thành công việc  $C_i$  (khoảng thời gian cần thiết để hoàn thành công việc  $J_j$ ). Hàm mục tiêu được xác định từ các hàm chi phí. Nhìn chung thì hàm mục tiêu sẽ được thể hiện dưới dạng  $\sum_j f_j$  hoặc  $\max_j f_j$

Các tiêu chí lịch trình công việc với các chức năng mục tiêu:

- Tổng thời gian hoàn thành  $\sum_j w_j C_j$
- Thời gian tối đa hoàn thành công việc hoặc makespan  $C_{max} = \max_j C_j$
- Độ trễ tối đa:  $L_{max} = \max_j L_j$

Mục tiêu của thuật toán điều phối lập lịch trình bày là để gán từng công việc  $J$  đến đúng với một máy  $M$ . Nói cách khác, mục tiêu của nó là để có được một chuỗi các công việc trên máy, giảm thiểu makespan tức là tổng thời gian hoàn thành tất cả các công việc.

Một số ràng buộc cần được xem xét:

- Quyền ưu tiên công việc không được cho phép.
- Mỗi máy  $m$  có thể xử lý chỉ một công việc  $J$  tại một thời điểm.
- Tất cả công việc có thể có đều có sẵn tại thời điểm  $t = 0$
- Thời gian vận chuyển được bỏ qua.

## Chương 3

# Ứng dụng lý thuyết trò chơi vào quản trị tiến độ thanh toán dự án

### 3.1 Một số khái niệm cần biết trong kinh tế

Chúng ta cần phải nắm rõ một số khái niệm trong kinh tế được sử dụng trong việc mô hình hóa bài toán:

- **Net Present Value(NPV):** là giá trị hiện tại thuần của dòng tiền của chuỗi thời gian các dòng tiền cả vào và ra, được định nghĩa là tổng các giá trị hiện tại(PV) của các dòng tiền cụ thể của cùng một thực thể. Nó là một công cụ trung tâm trong việc phân tích dòng tiền chiết khấu và là phương pháp tiêu chuẩn cho việc sử dụng giá trị thời gian của tiền để thẩm định dự án dài hạn.

$$NPV = \text{Giá hiện tại của dòng tiền vào} - \text{Giá trị hiện tại của dòng tiền ra}$$

- **Internal Rate Of Return(IRR):** là tỉ lệ hoàn vốn nội bộ làm cho giá trị hiện tại ròng của tất cả dòng tiền(cả dương và âm) từ một đầu tư cụ thể bằng không. Hay hiểu một cách khác thì IRR chính là nghiệm của phương trình  $NPV = 0$ . IRR càng cao thì khả năng thực thi dự án càng cao. IRR được sử dụng để đo lường, sắp xếp các dự án có triển vọng theo thứ tự.
- **Profit(Lợi nhuận):** là phần tài sản mà nhà thầu sau khi trừ đi hết các chi phí liên quan, bao gồm chi phí cơ hội. Nó là phần chênh lệch giữa tổng doanh thu và tổng chi phí.
- **Monthly Loan:** Lãi suất cho vay hàng tháng (tính theo đơn vị %)
- **Discount Rate(Tỉ lệ chiết khấu):** Trong đánh giá dự án, đánh giá quyết định đầu tư, tỉ lệ chiết khấu được dùng để tính tỉ suất hoàn vốn nội bộ(IRR), giá trị hiện tại thuần IRR. Nó đóng vai trò giúp qui giá trị các luồng tiền trong tương lai về thời điểm hiện tại, sau khi tính đến các nhân tố như lãi suất, lạm phát... Việc chọn được một tỉ lệ chiết khấu phù hợp là vô cùng quan trọng trong việc đánh giá dự án.
- **Prepay Rate:** là ngân sách hay là chi phí dự kiến ban đầu sẽ phải mất cho dự án.

## 3.2 Đặc tả vấn đề

Một đánh giá của quản lý dự án và lập lịch thanh toán dự án trong các tài liệu đã được công bố cho thấy một nỗ lực đáng kể trong việc tối ưu hóa chi phí và các nguồn lực sử dụng với mục tiêu giảm thiểu makespan. Việc tối ưu hóa tiến độ và chi phí thanh toán dự án ngày càng được quan tâm mặc dù các nỗ lực ban đầu đã xuất hiện từ khá sớm. Các khía cạnh tài chính trong quản lý dự án được bao gồm việc tối đa hóa các giá trị hiện tại ròng của dòng tiền NPV với mục tiêu gồm cả các tiêu cực (việc giải ngân) và tích cực (biên lai) trong suốt dự án. Một đánh giá của mô hình lưới dự án với các dòng tiền chiết khấu.

Các vấn đề được đề cập trong đề tài tốt nghiệp này được coi là phần mở rộng của các vấn đề điều phối tiến độ thanh toán. Các vấn đề điều phối thanh toán dự án xem xét về số tiền và thời gian của các khoản thanh toán như là các biến quyết định có thể ảnh hưởng tới lợi nhuận tài chính của cả nhà thầu, nhà phát triển và khách hàng. Số lượng và thời gian thanh toán là vấn đề quan trọng trong các cuộc đàm phán giữa các khách hàng và nhà thầu, nhà phát triển. Ở đây, các khoản thanh toán được xảy ra tại các nút sự kiện của mạng dự án.

Cả hai nhà thầu, phát triển và khách hàng đều muốn tối đa hóa lợi nhuận hoạt động tài chính và giá trị NPV của họ. Ngân sách bao gồm một tỉ suất lợi nhuận được thỏa thuận bởi hai bên. Các nhà thầu đáp ứng các chi phí cho các công việc, hoạt động bằng việc sử dụng các khoản thanh toán của khách hàng đối với ngân sách. Khi tiền tích lũy của các nhà thầu mang giá trị âm, giả sử nhà thầu được vay vốn với một lãi suất vay vốn lớn hơn lãi suất chiết khấu thường được sử dụng để tính toán NPV. Tiến độ thanh toán tối ưu nhất với nhà thầu là có được tổng số thanh toán một lần khi bắt đầu dự án. Sau khi nhận được tổng số tiền trả ban đầu, các nhà thầu sẽ cố gắng giảm thiểu chi phí của mình bằng cách lập lịch trình công việc theo cách thức sao cho dòng chảy tiền mặt cao hơn sẽ được sắp xếp càng muộn càng tốt. Để tính toán NPV của khách hàng, ta giả định rằng ngân sách ban đầu có sẵn và NPV của tất cả các khoản giải ngân bởi khách hàng phải được trừ vào khoản tiền này. Tiến độ thanh toán tối ưu nhất của khách hàng là việc thanh toán một lần duy nhất khi dự án được hoàn thành. Sau đó, khách hàng sẽ không phải lo lắng về lịch trình công việc mà chỉ quan tâm đến thời hạn dự án. Tiến độ thanh toán dự án tối ưu cho cả nhà thầu và khách hàng được gọi là giải pháp lý tưởng cho cả hai.

Các vấn đề thanh toán từ quan điểm của nhà thầu đã được xây dựng bởi Dayanand và Padman(1993). Hai ông đã đề xuất một chương trình và thử nghiệm một số heuristics. Dayanand và Padman trình bày một quy trình gồm hai giai đoạn, trong đó giai đoạn đầu tiên bao gồm một mô phỏng thuật toán luyện kim và trong giai đoạn thứ hai, hoạt động này được cơ cấu lại để nâng cao NPV của dự án. Hiệu suất của phương án này là tốt hơn so với các thử nghiệm với heuristics. Sau đó, nhìn vào các vấn đề lịch trình thanh toán từ quan điểm khách hàng, hai ông đã đưa ra mô hình lập trình tuyến tính đã được giới thiệu. Các phân tích cho thấy khách hàng có được những lợi ích lớn nếu các dự án kết thúc sớm. Làm các khoản thanh toán tại khoảng thời gian bằng nhau đã được chứng minh thương làm tăng chi phí của khách hàng. Năm 1997, Dayanand và Padman giới thiệu một số mô hình để phân tích xác định lịch trình thanh toán với mục tiêu tối đa hóa NPV của nhà thầu. Trong các mô hình, một thời hạn được áp đặt và số lượng các khoản thanh toán được cố định. Tổng số tiền mà khách hàng phải trả được xác định sẽ không bị thay đổi trong toàn bộ tiến độ của dự án.

Trong các nghiên cứu truyền thống trước đây về lịch trình, tiến độ thanh toán dự án với khía cạnh tài chính thì chỉ có giá trị hiện tại thuần của dòng tiền NPV của nhà thầu mới được xem xét và tối ưu hóa. Nhưng trong thực tế, cả khách hàng và nhà thầu đều có liên quan và cần phải đáp ứng các yêu cầu về thời gian, số lượng các khoản thanh toán mà khách hàng sẽ phải trả cho nhà thầu, phát triển thực hiện dự án. Nhà thầu, phát triển cần phải xác định một lịch trình hoạt động tương ứng với dự án đang thực hiện. Cả hai lịch trình thanh toán và lịch trình hoạt động có ảnh hưởng

đáng kể đến NPV của khách hàng và các nhà thầu, phát triển. Cả khách hàng và nhà thầu đều có ý định muốn tối đa hóa NPV của riêng mình. Nhưng gia tăng NPV của người này lại thường gây ra sự sụt giảm NPV của người khác.

### 3.3 Mô hình hóa trò chơi và cân bằng Nash

#### 3.3.1 Mạng dự án(Project Network)

Mạng dự án là một phương pháp mà cốt lõi của nó là dùng lý thuyết trò chơi có hướng để xác định đường đi trong mạng, từ thời điểm khởi đầu dự án đến thời điểm kết thúc dự án, qua một số công việc và các mối quan hệ giữa các công việc này. Một mạng dự án(project network) xác định rõ các quyền ưu tiên(thứ bậc) của hoạt động tham gia trong dự án.

Một mạng dự án là một dạng đồ thị có hướng không có chu trình(**DAG-Direct Cylic Graph**). Có hai kiểu mạng dự án thường gặp là **AoN** (activity-on-node: công việc trên nút) và **AoA**(activity-on-Arc: công việc trên vòng cung)

Một AoA network  $G = (E, A)$  là một đồ thị mà tập các nút  $E = \{e_1, e_2, \dots, e_m\}$  tương ứng với tập các sự kiện trong dự án, còn tập các cung  $A = \{a_1, a_2, \dots, a_n\}$  tương ứng với tập các hoạt động. Trong đó  $m$  là số các sự kiện và  $n$  là số các nhiệm vụ trong dự án.

Dựa trên **AoA network** các ràng buộc ưu tiên của dự án được xác định:

- Sự kiện  $e_i$  sẽ được diễn ra ngay sau khi tất cả các hoạt động tiền nhiệm trực tiếp của  $e_i$  đã hoàn thành.
- Chỉ khi xuất hiện của  $e_i$  thì các hoạt động kế tiếp trực tiếp của  $e_i$  mới bắt đầu.

#### 3.3.2 Các ràng buộc nguồn lực(Resource Constraints)

Chúng ta sử dụng kiểu số thực cho các tài nguyên, nguồn lực sử dụng trong dự án. Đối với mỗi khoảng thời gian trong quá trình xử lý dự án, số lượng các nguồn lực có sẵn được giới hạn bởi  $r_k$  (với  $k \in (i=1,2,\dots, R)$ )

#### 3.3.3 Đa cách thức(Mutil mode)

Mỗi hoạt động  $a_i \{1,2,\dots,n\}$  có thể tương ứng với bất cứ cách xử lý nào trong tập hợp hữu hạn các cách thức xử lý  $M_i = \{m_{i1}, m_{i2}, \dots, m_{|M_i|}\}$ , trong đó  $m_{ij}$  là cách xử lý thứ  $j$  của  $a_i$  và  $|M_i|$  là tổng số cách có thể của  $a_i$ .

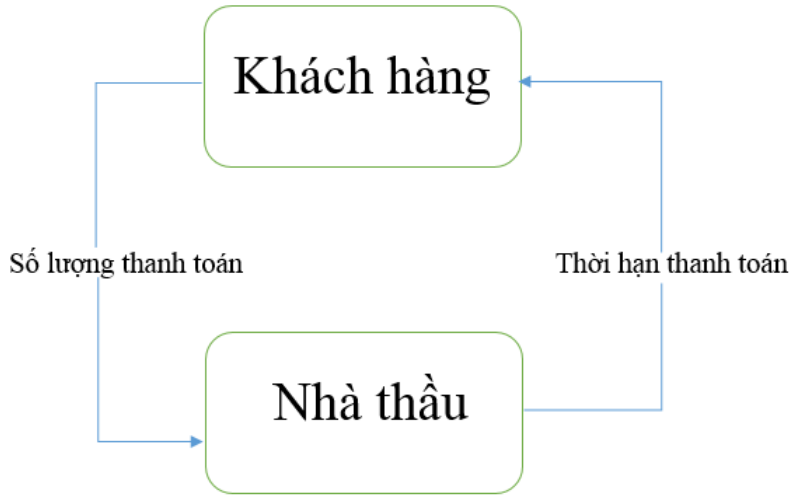
Cách khác để biểu diễn **thời gian/chi phí/nguồn lực** thông qua  $a_i$ , với việc thực hiện  $a_i$  theo các chế độ khác nhau tương ứng với số lượng thời gian, chi phí, nguồn lực. Cho một cách thức  $m_{ij}$ , chúng ta biểu thị thời gian của nó  $m_{ij}.d$ , chi phí  $m_{ij}.c$  và tiêu thụ nguồn lực thứ  $k$  là  $m_{ij}.r_k$ . Để lập kế hoạch dự án với nhiều chế độ chúng ta cần ánh xạ từng hoạt động  $a_i$  tới một hoạt động thực thi từ  $M_i$ .

#### 3.3.4 Cách thức mô hình hóa việc thanh toán dự án

Các vấn đề trong điều phối tiến độ thanh toán dự án được đại diện bởi các nút trạng thái trong mạng dự án AoA. Giả định rằng các điều khoản thanh toán trong hợp đồng dự án được thực hiện trong các giai đoạn khác nhau và việc thanh toán dự án được thực hiện tại một nút của một mạng lưới hoạt động nơi mà sẽ được hoàn thành từ một hay nhiều hoạt động, công việc khác nhau. Dự án sẽ được bắt đầu từ nút trạng thái đầu tiên và sẽ kết thúc tại nút trạng thái cuối cùng. Mỗi công việc tham gia trong dự án cần sự tham gia của một hay nhiều các nguồn lực, nguồn tài nguyên khác nhau với các mức sử dụng khác nhau. Mỗi cách hoàn thành công việc, hoạt động sẽ tương ứng với một chế độ liên đến công việc, hoạt động này. Vì vậy trong việc đưa ra các quyết định tiến độ thanh toán dự án thì giao thức lý tưởng nhất cho các nhà phát triển, chủ thầu là việc chỉ phải thanh toán

số tiền một lần cho tất cả dự án khi dự án kết thúc. Trong trường hợp này thì giá trị hiện tại ròng của dòng tiền là tối thiểu. Tuy nhiên giao thức có lợi nhất cho các khách hàng là các khách hàng có được tất cả các khoản thanh toán khi bắt đầu dự án. Trong trường hợp này, giá trị hiện tại thuần của dòng tiền thu được của khách hàng là tối đa.

Rõ ràng trong hầu hết các trường hợp, bất kỳ kế hoạch lợi nhuận nào cũng khó đáp ứng yêu cầu chấp nhận của cả hai bên và việc thỏa hiệp cần phải được xem xét. Một tình huống thực tế hơn là trong quá trình thỏa hiệp này, nhà thầu hay nhà phát triển phải đưa ra một lịch trình các khoản cần thanh toán ban đầu cho khách hàng, sau đó khách hàng xem xét nó và đưa ra các lịch trình hoạt động tương ứng với lịch trình thanh toán đó.



Hình 3.1: Sự tương tác của nhà thầu và khách hàng

Vì vậy vấn đề của việc tối ưu hóa tiến độ thanh toán dự án có thể được mô hình hóa thành một dạng trò chơi với thông tin đầy đủ được đại diện bởi các hình thức chiến lược như sau:

$$G = \{S_{client}, S_{client} \rightarrow S_{cont}; u_{client}(S_{client}, S_{cont}), u_{cont}(S_{client}, S_{cont})\}$$

Trong đó: (Owner: chủ thầu, nhà phát triển ; Client: Khách hàng, người chơi)

- $S_o$  biểu thị không gian chiến lược hay các quyết định được đưa ra của nhà phát triển.
- $u_o(S_o, S_c)$  là lợi ích của nhà phát triển (pay-off function: hàm lợi ích ròng).
- $S_c$  là hàm biểu thị không gian chiến lược hoặc các quyết định được đưa ra của khách hàng, người chơi.
- $u_c(S_o, S_c)$  là hàm lợi ích của người chơi, khách hàng.

Để xây dựng một tiến độ thanh toán dự án, Client phải chỉ rõ một danh sách thanh toán:

$$\text{payment list: } (pay_1, pay_2, \dots, pay_m) \quad (1)$$

trong đó  $pay_i$  là phần trăm ngân sách mà Client dự định để trả cho Owner tại sự kiện  $i$ . Danh sách thanh toán phải thỏa mãn 2 ràng buộc:

- $\sum_{i=1}^m pay_n = 1 \quad (2)$
- $pay_1 \geq prepayRate \quad (3)$

trong đó, biểu thức (2) nghĩa là tổng tất cả phần trăm ngân sách phải bằng 1, do đó tổng ngân sách phải bằng giá trị U của dự tính theo hợp đồng. Biểu thức (3) nghĩa là ngân sách lúc bắt đầu dự án (sự kiện  $e_1$ ) không được nhỏ hơn giá trị ngân sách dự tính trước  $prepayRate$ .

### Với chủ thầu, nhà phát triển

Với chủ thầu, nhà phát triển thì việc thanh toán càng được thực hiện càng muộn càng tốt để giảm thiểu giá trị hiện tại thuần dòng tiền NPV (Giá trị hiện tại thuần dòng tiền (NPV) là giá trị hiện tại của tổng các dòng tiền trong tương lai. NPV có công thức quan trọng trong việc sử dụng để tính toán sinh lời của dự án thầu và định giá chứng khoán).

$$u_{client}(S_{client}, S_{cont}) = NPV_{client} - NPV_{client}^{ideal}$$

trong đó:

- $NPV_o$  biểu thị các giá trị hiện tại thuần của dòng tiền thực trả cho nhà phát triển, chủ thầu.
- $NPV_{oi}$  là giá trị tối thiểu của giá trị hiện tại thuần của dòng tiền.

Nếu thời gian của  $a_i$  là  $a_i.st$  và thời gian khi  $e_i$  xảy ra là  $e_i.t$ , NPV của Owner là  $NPV_o$  được xác định theo công thức:

$$NPV_{cont} = \sum_{i=1}^m pay_i \cdot U \cdot e^{-\alpha \cdot e_i.t} - \sum_{i=1}^m m_{ij}.c \cdot e^{-\alpha \cdot a_i.st}$$

trong đó:

- U là ngân sách chi phí dự tính trong hợp đồng.
- r là lợi tức - lợi nhuận thu được.
- $\alpha$  là tỉ lệ chiết khấu (discount rate) - dùng để tính tỉ suất hoàn vốn nội bộ (IRR) và giá trị hiện tại thuần NPV.
- $m_{ij}$  là chế độ xử lý  $a_i$  trong tiến độ dự án,  $m_{ij}.c$  là chi phí của  $m_{ij}$

Các nhà thầu, phát triển dự kiến các quyết định của khách hàng, người chơi để lựa chọn quyết định  $S_c$  theo  $S_c(S_o)$  nên ở giai đoạn đầu tiên là:

$$\min_{S_{client}} u_{cont}(S_{client}, S_{cont}(S_{client})) = NPV_{client} - NPV_{client}^{ideal}$$

### Với khách hàng, người chơi

Để có được lợi nhuận nhiều nhất từ các chủ thầu thì những hoạt động tốn kém nhất của dự án được bố trí càng muộn càng tốt.

Hàm lợi ích ròng của khách hàng:

$$u_{cont}(S_{client}, S_{cont}) = NPV_{cont}^{ideal} - NPV_{cont}$$

trong đó:

- $NPV_{ci}$  biểu thị giá trị hiện tại ròng của dòng tiền thực thu của khách hàng, người chơi.
- $NPV_c$  là giá trị tối đa của giá trị hiện tại thuần của dòng tiền mặt của khách hàng hay người chơi.

NPV của Client được xác định theo công thức:

$$NPV_{client} = - \sum_{i=1}^m pay_i \cdot U \cdot e^{-\alpha \cdot e_i \cdot t} + r \cdot e^{-\alpha \cdot e_m \cdot t}$$

trong đó:

- $U$  là ngân sách chi phí dự tính trong hợp đồng.
- $r$  là lợi tức - lợi nhuận thu được.
- $\alpha$  là tỉ lệ chiết khấu (discount rate) - dùng để tính tỉ suất hoàn vốn nội bộ (IRR) và giá trị hiện tại thuần NPV.

Việc giải quyết yêu cầu đưa ra được chia làm hai giai đoạn khác nhau. Hai giai đoạn này được sử dụng để tìm ra cân bằng Nash hoàn hảo cho mô hình bài toán với thông tin đầy đủ. Giai đoạn thứ hai sẽ được xem xét trước, cụ thể là, để tối ưu hóa các quyết định của khách hàng hay người chơi trong trường hợp một lịch các khoản cần thanh toán  $S_o$  đã được cung cấp bởi nhà thầu hay nhà phát triển. Vấn đề đối với khách hàng, người chơi bây giờ là phải dựa trên lịch trình  $S_o$  để đưa ra một lịch trình hoạt động tương ứng  $S_c$  tối ưu để tối đa hóa giá trị hiện tại ròng của dòng tiền, hạn chế tối đa các giá trị của hàm lợi ích  $uc$ :

$$\min_{S_{cont}} u_{cont}(S_{client}, S_{cont}(S_{cont})) = NPV_{cont}^{ideal} - NPV_{cont}$$

trong đó  $S_c(S_o)$  có thể được làm việc với một phương pháp thích hợp như thuật toán di truyền GA.

Thay thế chiến lược của nhà thầu, phát triển  $S_o^*$  ở trạng thái cân bằng Nash vào  $S_c(S_o)$  cho chiến lược đáp ứng của khách hàng, người chơi  $S_c^*$ . Vì vậy, các giải pháp của các trò chơi hoàn hảo cân bằng Nash ( $S_o^*$ )

### 3.3.5 Tối ưu hóa mục tiêu

Mục đích của việc tối ưu hóa tiến độ thanh toán dự án là tìm được tiến độ thanh toán thỏa mãn các ràng buộc trong payment list và điều phối việc ánh xạ mỗi hoạt động  $a_i$  để có một cách thức thực thi thỏa mãn  $m_{ij}$

#### Tối ưu hóa tiến độ điều phối hoạt động

Trong quá trình thanh toán dự án, khách hàng xác định tiến độ thanh toán được đưa ra trong payment list và nhà thầu, nhà phát triển sẽ xác định lịch trình hoạt động cụ thể của danh sách công việc activity list và các danh sách chế độ tương ứng mode list. Khi cả khách hàng và nhà thầu muốn tối ưu hóa NPV của mình và đạt được lợi ích tối đa, đó là lý do mà họ hợp tác, đàm phán với nhau để đưa quyết định cuối cùng cho lịch trình thanh toán và lịch trình công việc. Tuy nhiên trong thực tế thì lịch trình thanh toán dự án thường được xác định bởi các khách hàng, sau đó mới đến các nhà thầu, nhà phát triển xác định lịch trình công việc. Nếu lịch trình công việc không được tối ưu hóa tốt thì sẽ dẫn tới việc nhà thầu, nhà thầu sẽ thay đổi và lịch trình công việc theo hướng nhằm nâng cao NPV để có lợi cho mình. Vì vậy trong đàm phán, chỉ có những giải pháp tối ưu hóa lịch trình công việc ở mức chấp nhận được mới được lựa chọn. Với một lịch trình thanh toán nhất định, vấn đề tối ưu hóa NPV của các nhà phát triển, nhà thầu chỉ là việc điều phối giải quyết các ràng buộc tài nguyên nguồn lực với tiền chiết khấu ròng. Để thuận tiện cho việc đưa ra quyết định, cần một giải pháp dựa trên sự khác biệt giữa giá trị giới hạn NPV của nhà phát triển, nhà thầu và NPV thực tế của lịch trình công việc. Giả sử với giải pháp  $x$  với một lịch trình thanh toán  $PS$  và một lịch trình công việc là  $AS$ , mức độ tối ưu hóa của nó trong lịch trình hoạt động được đánh giá như sau:



- **Bước 1:** Đánh giá các ràng buộc trên  $NPV_o^{upper/PS}$  của NPV nhà thầu, nhà thầu khi lịch trình thanh toán PS được cố định. Các ràng buộc trên thu được từ việc:
  - Bỏ qua tất cả các hạn chế về tài nguyên, nhân lực.
  - Giả sử tất cả các sự kiện, biến cố xảy ra sớm nhất có thể để các khoản thanh toán xảy ra càng sớm càng tốt.
  - Chi phí xảy ra càng muộn càng tốt.
  - Giả sử mọi công việc được thực hiện theo cách thức mà chi phí tổn kém là thấp nhất.
- **Bước 2:** Tính toán sự khác biệt giữa các ràng buộc  $NPV_o^{upper/PS}$  và NPV thực tế của nhà đầu tư, nhà thầu  $NPV_o$

$$OD(x) = NPV_o^{upper/PS} - NPV_o$$

Nếu giá trị mức tối ưu hóa  $OD(x)$  là nhỏ thì có nghĩa là lịch trình công việc trong giải pháp  $x$  cũng được tối ưu hóa. Nếu không, lịch trình công việc trong  $x$  không tối ưu hóa, do đó giải pháp  $x$  sẽ không được lựa chọn để đưa ra quyết định.

### 3.4 Giải thuật di truyền cho giải pháp cân bằng Nash

Giải thuật di truyền là một công cụ hiệu quả cho các vấn đề tối ưu hóa tổ hợp. Nó được áp dụng để xây dựng mô hình hóa giải quyết các mô hình trò chơi với thông tin đầy đủ, tối ưu hóa tiến độ thanh toán dự án.

#### 3.4.1 Mô hình chung của thiết kế giải thuật di truyền

##### Các ràng buộc của bài toán

Mô hình giải quyết bài toán phải thỏa mãn một số yêu cầu ràng buộc sau:

- Một trạng thái của dự án được hoàn thành khi tất cả các hoạt động ràng buộc với nó phải được hoàn thành.
- Việc chuyển đổi trạng thái dự án được thực hiện khi các trạng thái dự án ràng buộc trước đó đã được hoàn thành.

##### Lược đồ mã hóa

Trong việc tối ưu việc thanh toán dự án, cả tiến độ thanh toán và tiến độ dự án cũng cần phải được tối ưu. Để biểu diễn các biến quyết định được tối ưu thì ta thiết kế một giải pháp để xem xét vấn đề này:

- Payment list:  $(share_1, share_2, \dots, share_m)$
- Activity list:  $act_1, act_2, \dots, act_n$
- Mode list:  $(mode_1, mode_2, \dots, mode_n)$

Nếu ta tiến hành chia ngân sách dành cho việc thanh toán dự án thành 20 phần thì mỗi phần sẽ tương ứng với 5% tổng ngân sách dành cho việc thanh toán dự án. Nếu ta gán  $share_i$  với event  $e_i$  nó có nghĩa là 5% của tổng ngân sách thanh toán sẽ xuất hiện ở  $e_i$ . Theo cách này việc đưa ra  $(share_1, share_2, \dots, share_m)$  có thể được biến đổi tới dạng chuẩn của payment list:

$$pay_i = \begin{cases} \sum_{j=1}^{i-1} \Gamma_j(i).5\%.(1 - prepayRate) + prepayRate, & j = 1 \\ \sum_{j=1}^{20} \Gamma_j(i).5\%.(1 - prepayRate) \end{cases}$$

trong đó:

- 

$$\Gamma_j(i) = \begin{cases} 1, & \text{if } share_i = e_i \\ 0, & \text{otherwise} \end{cases}$$

- *prepayRate* là ngân sách dự tính trước.

## Sự chọn lọc

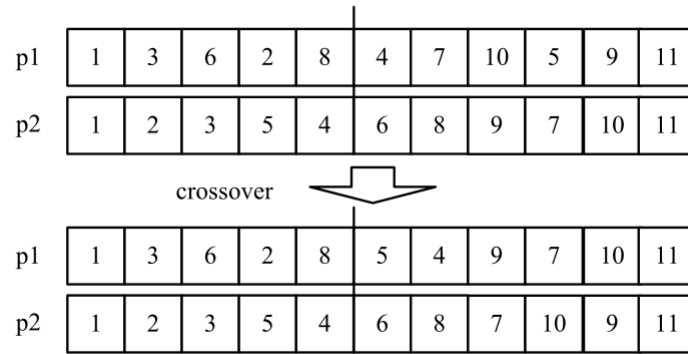
Các cá thể trong quần thể sẽ được sắp xếp dựa trên việc trội hơn và dựa trên khoảng cách tập trung tương ứng, sự chọn lọc sẽ sử dụng một toán tử tập trung. Tiến hành loại bỏ các phương án trội trong quần thể và thay thế bằng các phương án mới không bị trội. Cá thể được lựa chọn bằng lựa chọn giải nhị phân với toán tử so sánh tập trung.

## Trao đổi, lai ghép chéo

Trong giải thuật được đề xuất, các cá thể được chọn thực hiện trao đổi chéo với một xác suất  $px$ . Các toán tử trao đổi chéo có được thực hiện trong list payment, mode list hoặc mode list một cách ngẫu nhiên. Các trao đổi chéo một điểm sẽ được sử dụng với payment list và mode list.

Đối với các trao đổi chéo trong activity list, để đảm bảo rằng danh sách các hoạt động mới được tạo ra có thể luôn luôn tuân theo chế độ ưu tiên được xác định bởi các mạng AoA, một điểm làm cơ sở khi áp dụng trao đổi chéo. Ý tưởng cơ bản của trao đổi chéo này:

- Cho các vị trí trước khi lựa chọn ngẫu nhiên điểm trao đổi chéo, các hoạt động vẫn không thay đổi.
- Cho vị trí sau khi trao đổi chéo xảy ra, các hoạt động được sắp xếp lại dựa trên thứ tự xuất hiện trong cặp cha mẹ khác. Lấy một ví dụ



Hình 3.2: Trao đổi chéo

## Đột biến

Trong giải thuật, gen được lựa chọn để thực hiện đột biến với xác suất  $pm$ :

- Nếu  $share_i$  thuộc về payment list được lựa chọn để đột biến,  $share_i$  được gán ngẫu nhiên tới một khả năng có thể xảy ra  $e_{ran}$ , với  $ran$  là một số ngẫu nhiên phân phối đều trong tập  $\{1, 2, \dots, m\}$
- Nếu một gen  $act_i$  thuộc một activity list được chọn để đột biến, chúng ta sẽ hoán đổi giá trị của  $act_i$  và  $act_{i+1}$  nếu quyền ưu tiên ràng buộc bị vi phạm. Mặt khác, NST còn lại sẽ không bị thay đổi.

- Nếu một gen  $mod_i$  thuộc về mode list được chọn để đột biến thì giá trị của  $mod_i$  sẽ được đặt lại một cách ngẫu nhiên  $m_{i,ran}$  với  $ran$  là một số ngẫu nhiên phân phối đều trong tập  $1, 2, \dots, \|M_i\|$

### 3.4.2 Thiết kế giải thuật di truyền cho chiến lược đầu tiên

Hàm mục tiêu trong chiến lược đầu tiên này sẽ được tính theo công thức:

$$Z = \left\| \frac{NPV_{cont}^{ideal} - NPV_{cont}}{NPV_{cont}^{ideal}} - \frac{NPV_{client}^{ideal} - NPV_{client}}{NPV_{client}^{ideal}} \right\|$$

trong đó *ideal* chỉ giải pháp lý tưởng trong mỗi trường hợp tương ứng với  $NPV_c$  và  $NPV_o$ .

Khi tính toán giá trị của hàm mục tiêu của các NST, ta làm tròn giá trị đến chữ số có nghĩa thứ hai. Với một giá trị hàm mục tiêu nhất định có thể tồn tại một số giải pháp tương ứng để  $NPV_c$  và  $NPV_o$  khác nhau, giải pháp đó được gọi là giải pháp chi phối. Rõ ràng các giải pháp chi phối này sẽ được đưa ra bởi các nhà quyết định. Giả định rằng  $NPV_c$  được tính bằng cách sử dụng cùng một tỉ lệ chiết khấu  $r$  với  $NPV_o$ .

#### Codes (Mã)

Tiến độ thanh toán được cung cấp bởi nhà phát triển, chủ thầu được sử dụng như là các nhiệm sác thể(NST) trong giải thuật di truyền. Sự biểu diễn của NST liên quan đến tỉ lệ phần trăm của tiến độ thanh toán dự án sẽ được trả tại các nút trạng thái dự án. NST được kí hiệu bằng một loạt các số là số nút trong mạng lưới hoạt động. Vì vậy mỗi số nguyên đại diện cho một gen của NST. Số gen có liên quan trực tiếp đến kích thước của các đơn vị cổ phần của ngân sách phân bổ cho một gen. Nếu số NST có 20 gen thì mỗi gen sẽ được phân bổ 5% ngân sách tương ứng với NST đó. Vì vậy số tiền mà nhà thầu hay nhà phát triển phải trả cho một nút được thể hiện bởi tần số xuất hiện của các gen đại diện cho nút đó trong NST.

Một NST mẫu cho một mạng lưới với 6 nút trạng thái và các đơn vị cổ phần của ngân sách cho mỗi gen là 10%, ví dụ là (3/5/1/6/5/6/1/6/3/5). Nhìn vào NST mẫu, ta có thể suy ra được rằng có hai nút 1 và ba nút 6. Do đó, nút 1 và nút 6 lần lượt nhận được 20% và 30% tổng số ngân sách thanh toán. Nếu một phân phối tốt hơn được yêu cầu thì có lẽ số lượng gen trong NST cần phải được sử dụng nhiều hơn. Bằng cách này, một phân bố ngân sách chính xác hơn cho các nút trạng thái dự án có thể thu được nhưng với một chi phí gia tăng nỗ lực tính toán.

#### Opearators(Thao tác)

Tần số xuất hiện của các gen đại diện cho các nút quan trọng trong việc thể hiện tiến độ thanh toán dự án của chủ thầu.

*Toán tử lai ghép, trao đổi chéo:* Với chiến lược đầu tiên này,ta sử dụng toán tử lai ghép chéo hai điểm. Từ những thông tin quan trọng về số lần xuất hiện của nút sự kiện, nút trạng thái dự án, lai ghép chéo hai điểm có thể áp dụng được mà tránh được những phương án không khả thi gây ra. Vì vậy hai điểm được chọn lai ghép phải được đề xuất trong thuật toán. Để áp dụng được các toán tử lai ghép chéo, hai điểm được lựa chọn để phân chia NST cha mẹ thành 3 phần. Những NST con được hình thành bằng cách lấy những phần tử đầu tiên, phần tử cuối cùng của NST mẹ và phần giữa của NST cha. Lấy một ví dụ về toán tử lai ghép chéo hai điểm được đưa ra như hình phía dưới. Các gen chuyển sang NST con sẽ được in đậm. Kết quả lai ghép chéo trên một con duy nhất.

MOTHER																			
1	3	5	3	4	1	1	2	3	3	5	5	5	4	2	1	5	3	4	3
FATHER																			
2	2	5	4	1	1	3	5	4	2	1	2	3	5	4	2	3	3	1	2
CHILD																			
1	3	5	3	1	1	3	5	4	2	1	2	3	5	4	1	5	3	4	3

Hình 3.3: Ví dụ về toán tử lai ghép chéo hai điểm

*Toán tử đột biến:* Hai loại toán tử đột biến được xem xét. Một là toán tử đột biến toàn cục. Một NST được chọn để gây đột biến với xác suất đột biến toàn cục và mỗi điểm của cá thể trên NST được xem xét để sử dụng các xác suất đột biến toàn cục, các điểm được đột biến được gán với một sự kiện ngẫu nhiên. Trên một NST có thể chọn nhiều hơn một nơi để gây đột biến. Toán tử đột biến thứ hai là toán tử đột biến địa phương. Một NST được chọn để gây đột biến với xác suất đột biến địa phương và một nơi trên NST được chọn để biến đổi, các nơi được chọn gây đột biến được gán với một sự kiện ngẫu nhiên. Do đó, chỉ có một gen là bị đột biến trên một NST được lựa chọn. Nhìn chung thì các NST đột biến đại diện cho lịch trình thanh toán của chủ thầu, nhà phát triển bao gồm việc lựa chọn các NST đột biến, vị trí gen đột biến và các giá trị đột biến gen. Các vị trí đột biến và giá trị đột biến được xác định một cách ngẫu nhiên. Vì vậy các giá trị thích nghi của NST được tính.

*Giá trị hàm thích nghi* Để xác định giá trị thích nghi của một NST  $X$  của chiến lược đầu tiên này, ta lấy giá trị hàm mục tiêu cho NST  $X$  trừ đi giá trị lớn nhất của hàm mục tiêu trong quần thể. Để tính toán được giá trị thích nghi, ta cần phải xác định được hai giá trị  $NPV_c$  và  $NPV_o$ . Mỗi NST  $X$  sinh ra trong chiến lược đầu tiên này sẽ được gửi tới chiến lược thứ hai để xác định giá trị  $NPV_o$  lớn nhất, xác định vị trí và thời gian tương ứng với các khoản thanh toán. Các thông tin này được sử dụng cùng với cơ cấu thanh toán để tính  $NPV_c$ .

### Adaptability(Khả năng thích nghi)

Sau khi chủ thầu cung cấp một lịch trình thanh toán cho khách hàng, khách hàng cố gắng tối đa hóa  $NPV_c$  bằng cách điều chỉnh mạng lưới hoạt động của mình để có lịch trình tương ứng. Lịch trình của khách hàng được giao cho các chủ thầu tính toán  $NPV_o$  của nó theo hai lịch trình. Các hàm lợi ích được chỉ định trong khả năng thích ứng hay thích nghi của các NST. Nó được coi là tỉ lệ chiết khấu ( $r$ ) cho cả chủ thầu và khách hàng là như nhau cho việc tính toán  $NPV_o$  và  $NPV_o i$ . Ngoài ra số lượng các khoản thanh toán của chủ thầu không đủ, vấn đề tài chính của khách hàng từ một ngân sách với một lãi suất  $i$  cao hơn  $r$ .

### 3.4.3 Thiết kế giải thuật di truyền cho chiến lược thứ hai

Mục tiêu trong chiến lược thứ hai là tối đa hóa  $NPV_o$  trong một phân phối thanh toán cụ thể. Sự phân phối thanh toán được đặt lên một cá thể của chiến lược đầu tiên, cá thể mà giá trị thích đã được xác định. Kết quả thu được từ chiến lược thứ hai là một lịch trình công việc, hoạt động

được xác định bằng việc tối đa hóa  $NPV_o$ .  $NPV_o$  được tính toán bằng cách sử dụng tỉ lệ chiết khấu  $r$ . Trong khoảng thời gian khi các khoản thanh toán tích lũy được thực hiện bởi khách hàng không đáp ứng được lũy kế chi phí phát sinh bởi các nhà thầu, các nhà thầu vay mượn tiền với tỉ lệ vay nợ lớn hơn tỉ lệ chiết khấu  $r$ . Khi nhận được lịch trình công việc, hoạt động thì  $NPV_c$  có thể được tính toán một cách dễ dàng kể từ lần xuất hiện của các nút trạng thái và do đó các khoản thanh toán đã sẵn sàng.

## Codes (Mã)

NST của việc điều phối các hoạt động của khách hàng sẽ có thể đại diện cho các hoạt động bao gồm các bước và các chế độ hoạt động của các hoạt động, công việc. Mỗi gen phải có số thứ tự và chế độ hoạt động tương ứng. Số thứ tự của các hoạt động này được đại diện bởi các vị trí các gen trong NST. Mỗi gen trong NST được mã hóa bởi một số nguyên như  $m - n$  trong đó  $m$  biểu thị chuỗi hành động và  $n$  là số các chế độ hoạt động.

Rõ ràng là bất kì một hoạt động nào được đặt trong một gen đều không được vi phạm các mối quan hệ ưu tiên. Hơn nữa các toán tử khác gây ra một sự thay đổi trong nội dung gen của một cá thể phải đảm bảo tính khả thi. Một cá thể mẫu với 5 hoạt động, công việc được viết như sau:  $(1 - 1, 3 - 2, 5 - 1, 2 - 1, 4 - 3)$ . Trong đó, chữ số đầu tiên đại diện cho số hoạt động và chữ số thứ hai thể hiện các chế độ liên quan đến hoạt động đó.

## Opearators(Thao tác)

*Toán tử lai ghép chéo:* Để đảm bảo tính khả thi của một giải pháp, một toán tử giao phối, đa thành phần trật tự đều dựa trên trao đổi chéo được thông qua(MCUOX).

START

- Với vị trí đầu tiên của cha mẹ và con.

LOOP

- Chọn ngẫu nhiên một cặp cha mẹ.
- Tìm công việc, hoạt động đầu tiên trong cặp cha mẹ được chọn mà chưa đưa gán cho con nào. Công việc, hoạt động đó sẽ trở thành công việc, hoạt động kế tiếp trên con.
- Nếu việc gán các hoạt động được thực hiện giống nhau trên cả cha mẹ, thì việc gán các hoạt động trên con cũng hoàn toàn tương tự. Còn nếu không, chọn một cách thức gán của cặp cha mẹ một cách ngẫu nhiên và ghi nó vào con.
- Nếu tất cả các gen của NST con được hoàn thành thì STOP.
- Còn nếu không tiếp tục xem xét tới gen tiếp theo trong con và lặp lại LOOP.

**MOTHER**

[1-1, 3-2, 2-1, 5-1, 4-3, 6-2]

**FATHER**

[1-1, 4-2, 2-2, 3-1, 6-2, 5-2]

Iteration	Result of parent selection		Child coding
	Activity	Mode	
0			[-, -, -, -, -, -]
1	Mother	No need	[1-1, -, -, -, -, -]
2	Father	Mother	[1-1, 4-3, -, -, -, -]
3	Mother	Father	[1-1, 4-3, 3-1, -, -, -]
4	Mother	Father	[1-1, 4-3, 3-1, 2-2, -, -]
5	Father	No need	[1-1, 4-3, 3-1, 2-2, 6-2, -]
6	Mother	Mother	[1-1, 4-3, 3-1, 2-2, 6-2, 5-1]

**CHILD**

[1-1, 4-3, 3-1, 2-2, 6-2, 5-1]

Hình 3.4: Ví dụ về toán tử lai ghép MCUOX

*Các toán tử đột biến* Với chiến lược thứ hai này, có hai kiểu toán tử đột biến được sử dụng. Toán tử đột biến đầu tiên là đột biến tái định vị. Đột biến tái định vị chọn ngẫu nhiên ra hai vị trí trong NST và chuyển nội dung vị trí trước xuống vị trí sau. Đột biến tái định vị có thể gây ra các hành vi vi phạm quyền ưu tiên, do vậy một chức năng sửa chữa đơn gian được bổ sung đột biến tái định vị.

- Giả sử các gen ở trong vị trí  $pos1$  thay đổi vị trí vào một  $pos2$
- Các thuật toán thực hiện một chức năng sửa chữa cho hai trường hợp  $pos1 < pos2$  và  $pos1 > pos2$ .
  - Khi  $pos1 < pos2$  thì các hoạt động trong  $pos2$  đã nhảy qua một số vị trí kế tiếp. Nếu đây là trường hợp các gen chuyển giao cho vị trí locus  $pos2$  được thay đổi vị trí trong NST ở các điểm ngay trước khi sự kế thừa đầu tiên giữa  $pos1$  và  $pos2 - 1$ .
  - Khi  $pos1 > pos2$ , hoạt động đặt trên locus  $pos2$  có thể vi phạm một số mối quan hệ ưu tiên. Vì vậy hoạt động này được đặt ngay sau hoạt động tiền nhiệm cuối cùng giữa  $pos2 + 1$  và  $pos1$

Toán tử đột biến thứ hai là đột biến đảo bit, sử dụng để thay đổi các lựa chọn cho một hoạt động. Đối với các toán tử hoạt động, phương thức hoạt động, các NST được lựa chọn một cách ngẫu nhiên và các chế độ hoạt động cũng được phân chia ngẫu nhiên. Để đáp ứng tiến độ thanh toán được cung cấp bởi các chủ thầu, khách hàng.

*Vấn đề điều phối, thời hạn* Để có được giá trị NPV một cá thể trong quần thể, trước tiên cá thể đó phải được lên kế hoạch điều phối. Điều phối được sử dụng là một điều phối chuyển tiếp (Forward Scheduler) với việc điều phối hoạt động xảy ra càng sớm càng tốt nhưng vẫn phải đảm bảo không vi phạm các hạn chế tài nguyên, nguồn lực và sự sắp đặt các hoạt động trên các nhiệm sắc thể (NST).

Giả sử có hai hoạt động, công việc xảy ra liên tiếp là *act1* và *act2* trên NST theo đúng thứ tự và giả sử rằng cả hai đều có đầy đủ các điều kiện, tức là những hoạt động xảy ra trước nó đã được điều phối. Nếu các nguồn lực, tài nguyên có sẵn không đủ cho *act1* thì *act2* cũng sẽ không được điều phối ngay cả khi nguồn lực, tài nguyên đủ cho việc lập lịch nó. Do đó, việc điều phối lập lịch sẽ giữ nguyên vị trí, trật tự thứ tự của các hoạt động có trên NST, có nghĩa là bất kỳ hoạt động nào trong locus sẽ không có thời gian bắt đầu nhỏ hơn một hoạt động trong một locus nằm trước nó trên NST. Hơn nữa, để cải thiện tiến độ bằng cách không dịch chuyển thứ tự các hoạt động, công việc trên NST với thời gian bắt đầu các hoạt động bao hàm các sự can thiệp với quá trình tiến hóa tự nhiên của thuật toán di truyền. Lịch trình, điều phối lập lịch được sửa đổi theo cách này sẽ tương ứng với một NST khác trong quá trình tạo ra thuật toán mới.

Một điều phối ngược(Backward Scheduler) cũng đã được sử dụng cùng với lập lịch, điều phối chuyển tiếp. Điều phối ngược sẽ trì hoãn một số hoạt động, công việc mà không vi phạm trật tự của các hoạt động, công việc trên NST và/hoặc thanh toán bản lẻ khi hoạt động kết thúc. Nhưng điều phối lập lịch ngược lại không dẫn đến một sự gia tăng đáng kể nào về NPV để giải thích cho việc gia tăng thời gian tính toán do việc sử dụng nó. Vì hạn chế này mà điều phối ngược chỉ được áp dụng cho những NST cuối cùng để cải thiện giải pháp cuối cùng, bằng cách trì hoãn các hoạt động nhất định mà không gây ra bất kỳ sự gia tăng trong thời gian dự án.

*Giá trị thích nghi:* Một khi NST được điều phối, thời hạn kết thúc hoạt động, công việc và. Để tính giá trị hàm mục tiêu trong chiến lược thứ hai này , một tỉ lệ chiết khấu, tỉ lệ cho vay, thời gian và số lượng tiền phải trả cần phải được xác định. Giá trị thích nghi của NST trong quần thể được tính bằng giá trị tối thiểu của NPV của quần thể trừ đi giá trị NPV của NST.

### Adaptability(Khả năng thích nghi)

Sau khi nhận được thông tin về tiến độ thanh toán được đưa ra bởi chủ thầu, các khách hàng điều chỉnh mạng lưới hoạt động và các chế độ hoạt động tối đa hóa  $NPV_c$ . Các hàm lợi ích cho khách hàng được chỉ định là khả năng thích ứng hoặc thích nghi của các NST tương ứng.

### 3.4.4 Giá trị lý tưởng cho nhà thầu và khách hàng

Tiến độ thanh toán mong muốn nhất đối với nhà thầu là việc trả tiền chỉ được thực hiện vào một lần duy nhất khi bắt đầu dự án. Còn đối với khách hàng thì tiến độ thanh toán mong muốn nhất là một khoản tiền thanh toán thực hiện chỉ một lần khi dự án được hoàn thành. Cả hai điểm lý tưởng đang bị ảnh hưởng bởi các vấn đề về điều phối, thời hạn.

Để nắm bắt được các giải pháp lý tưởng nhất cho các nhà thầu và khách hàng, các thuật toán di truyền đã được sử dụng. Trong đó, cơ cấu thanh toán chỉ gồm có việc trả tiền vào đầu dự án và vào cuối dự án.

### 3.4.5 Lựa chọn thông số

Các tham số sử dụng trong giải thuật phải được lựa chọn một cách chính xác để nâng cao tính hiệu quả của giải thuật di truyền GA.

Các thông số của thuật toán di truyền được xác định theo phương pháp meta-GA với các thông số về xác suất giao phối, xác suất đột biến vị trí... Ý tưởng meta-GA sử dụng giải thuật GA trong việc tìm kiếm ra các tham số tốt nhất sử dụng trong giải thuật di truyền. Mã hóa của các NST của các meta-GA được dựa trên các thông số được thiết lập. Đối với chiến lược đầu tiên các thông số của NST như sau: *số lần lặp lại, số lượng các thế hệ, quy mô quần thể, xác suất trao đổi chéo, xác suất đột biến đảo bit, xác suất đột biến tái định vị, tỷ lệ elitist* . Đối với chiến lược thứ hai, các tham

số sẽ thành: số lần lặp lại, số lượng thể hệ, quy mô quần thể, xác suất trao đổi chéo, xác suất đột biến, tỷ lệ elitist

Mục tiêu và hàm thích nghi là như nhau cho meta-GAs. Các vấn đề được lựa chọn cho giải pháp này là một trong số 53 hoạt động, 30 nút và 3 nguồn tài nguyên, nguồn lực. Quy mô quần thể là 50 và meta-GAs tiếp tục là 60 thể hệ. Quần thể ban đầu được tạo một cách ngẫu nhiên. Các NST với giá trị thích nghi lớn nhất của thể hệ trước đó được chuyển giao cho các thể hệ tiếp theo. Để tạo ra các NST mới, trao đổi chéo hai điểm và đột biến được áp dụng liên tục với xác suất lai ghép chéo là 0,60 và xác suất của đột biến đảo bit là 0.125. Việc lựa chọn cặp NST cha mẹ cho sự lai ghép, trao đổi chéo được thực hiện một cách ngẫu nhiên từ quần thể hiện tại với sự thay thế và với xác suất lựa chọn là tỷ lệ thuận với độ thích nghi của các NST. Các giá trị tham số sẽ được sử dụng:

Parameter	Values	Value Selected
Number of replications	2, 3, 4, 5, 6, 7, 8	2
Number of generations	25, 30, 35,..., 95, 100	60
Population size	25, 30, 35,..., 95, 100	65
Crossover probability	0.2, 0.25, 0.3,..., 0.75, 0.8	0.70
Global mutation prob.	0.05, 0.1, 0.15, 0.2, 0.25	0.25
Elitist ratio	0.01, 0.02, 0.03,..., 0.14, 0.15	0.07

Hình 3.5: Các tham số giải thuật sử dụng trong chiến lược ban đầu

Parameter	Values	Value Selected
Number of replications	4, 5, 6, 7, 8, 9, 10, 11, 12	6
Number of generations	25, 30, 35,..., 95, 100	60
Population size	25, 30, 35,..., 95, 100	45
Crossover probability	0.2, 0.25, 0.3,..., 0.75, 0.8	0.65
Bit mutation probability	0.05, 0.1, 0.15, 0.2	0.15
Rep. mutation probability	0.05, 0.1, 0.15, 0.2	0.15
Elitist ratio	0.01, 0.02, 0.03,..., 0.14, 0.15	0.06

Hình 3.6: Các tham số giải thuật sử dụng trong chiến lược thứ hai



## Chương 4

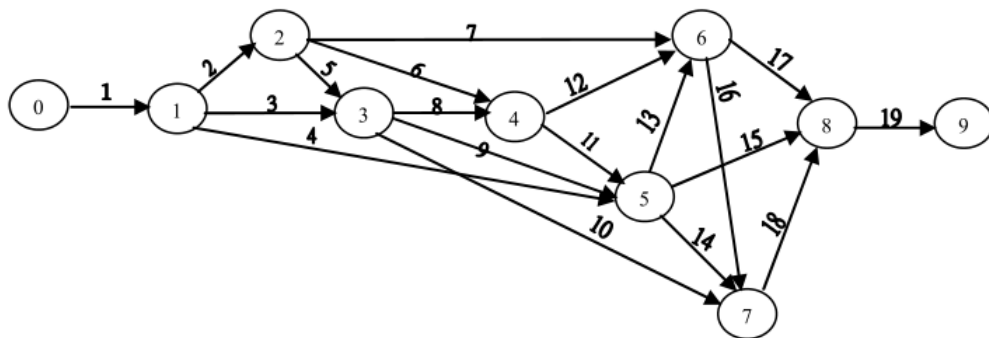
# Xây dựng và thử nghiệm thuật toán tối ưu hóa quản lý thanh toán dự án

### 4.1 Xây dựng và thử nghiệm mô hình

#### 4.1.1 Kịch bản mô phỏng

##### Đầu vào (Input)

Cho đầu vào là một mạng lưới hoạt động(Activity network) cho biết số trạng thái, số các activity sẽ có trong dự án và đặc biệt sẽ cho biết các ràng buộc yêu cầu để chuyển đổi từ trạng thái dự án này sang trạng thái dự án khác. Ví dụ: Mạng lưới hoạt động(activity network) sẽ được một tả thành



Hình 4.1: Mạng hoạt động Activity network.

một bảng trong file excel với:

- **Node:** Tên các trạng thái của dự án.
- **Next node:** Các trạng thái mà trạng thái ban đầu có ràng buộc đến.
- **Activity:** Các hoạt động tương ứng với các trạng thái project ràng buộc.

Node	NextNode	Activity
0	1	1
1	2, 3, 5	2, 3, 4
2	3, 4, 6	5, 6, 7
3	4, 5, 7	8, 9, 10
4	5, 6	11, 12
5	6, 7, 8	13, 14, 15
6	7, 8	16, 17
7	8, 9, 10	18, 19, 20
8	9, 10	21, 22
9	10	23

Trong dự án có một số hoạt động có hai đường để hoàn thành. Các thông số của các activity network sẽ được thể hiện thông qua.

- **Activity:** Các hoạt động tham gia vào trong dự án.
- **Time/d:** Thời gian cần để thực hiện hành động đó trong một đơn vị thời gian
- **Resource 1, Resource 2:** Số lượng nguồn lực, tài nguyên 1, 2 cần sử dụng để hoàn thành activity trong thời gian Time/d.

Ví dụ:

Activity	Operation mode 1			Operation mode 2		
	Time/d	Resource 1	Resource 2	Time/d	Resource 1	Resource 2
1	10	6	2			
2	30	6	1			
3	50	3	0			
4	50	3	0			
5	10	3	1			
6	60	6	2	20	1	1
7	60	6	2	100	2	1
8	50	3	1	100	4	2
9	40	6	0	70	1	1
10	90	3	2			
11	80	2	2	100	2	0
12	30	3	0	100	2	1
13	30	3	0			
14	50	3	2	60	1	0
15	150	3	2	60	6	2
16	50	3	2			
17	40	0	1	120	3	2
18	110	4	1			
19	90	6	2			

Hình 4.2: Các thông số tương ứng với các activity

Một số tham số tham gia trong quá trình demo:

- **profit:** lợi nhuận = 30% chi phí
- **discount rate:** tỉ lệ chiết khấu - dùng để tính tỉ suất hoàn vốn nội bộ (IRR), giá trị hiện tại thuần của dòng tiền (NPV). Tỉ lệ chiết suất có vai trò giúp qui giá trị các luồng tiền trong tương lai về thời điểm hiện tại sau khi đã tính đến các nhân tố như lãi suất, lạm phát.
- **monthly loan:** Lãi xuất cho vay hàng tháng

### Đầu ra mong muốn

Đầu ra mà chúng ta mong muốn chính là việc tối ưu được điều phối các hoạt động tham gia, tối ưu được tiến độ thanh toán dự án từ đó tối ưu được chi phí bỏ ra và tối ưu được lợi nhuận. Ví dụ:

Node	Time/d	Payment/%
1	10	10
2	40	10
3	100	30
4	250	15
5	350	20
6	520	0
7	690	5
8	900	5
9	990	5

*Bảng: Tối ưu tiến độ thanh toán dự án*

Activity	Starting time/d
1	0
2	10
3	40
4	40
5	90
6	190
7	250
8	90
9	310
10	100
11	350
12	430
13	460
14	460
15	520
16	490
17	490
18	790
19	900

*Bảng: Tối ưu hóa việc điều phối các hoạt động trong dự án*

### 4.1.2 Các tham số của thuật toán di truyền

Các tham số được lựa chọn để sử dụng trong giải thuật di truyền GA sẽ được xác định theo phương pháp meta-GA: (áp dụng riêng cho từng giai đoạn và chiến lược áp dụng đối với nhà phát triển, đầu tư hay khách hàng, người chơi)

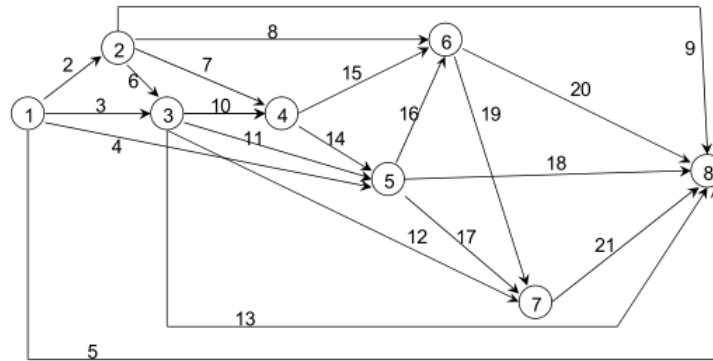
Stage	First	Second
Number of replications	2	6
Number of generations	80	80
Population size	60	50
Mating probability	0.70	0.67
Operation mode mutation probability	0.20	0.20
Positioning mutation probability		0.15
Elitist ratio	0.06	0.05

### 4.1.3 Bộ dữ liệu đầu vào

Để mô phỏng và thử nghiệm thuật toán được xây dựng để tối hóa tiến độ thanh toán dự án phần mềm, ta sử dụng hai bộ dữ liệu như sau:

#### Bộ dữ liệu 1

Sơ đồ mạng dự án:



Hình 4.3: Sơ đồ mạng dự án 1

Các giá trị giới hạn của Resource1 và Resource2 lần lượt là 5 và 2 đơn vị. Ngân sách sẽ là tổng giá trị chi phí lớn nhất của các hoạt động trong dự án. Tỷ lệ chiết khấu của khách hàng và nhà thầu đều bằng 0.5%; lãi suất cho vay là 0.9%

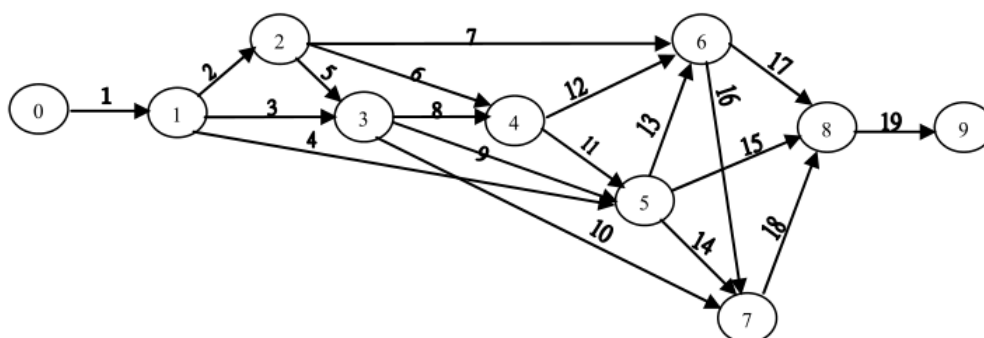
Các thông số của các activity về time/d, resource1, resource2:

Activity	Mode 1			Mode 2		
	Duration	Resource 1 Usage	Resource 2 Usage	Duration	Resource 1 Usage	Resource 2 Usage
1	0	0	0			
2	3	5	2			
3	5	3	1			
4	5	3	0			
5	5	3	0			
6	1	3	1	2	1	1
7	6	5	2	10	2	1
8	6	5	2	10	4	2
9	5	3	1	7	1	1
10	4	5	2			
11	9	3	0	10	2	0
12	7	5	2	10	2	1
13	8	2	2			
14	3	3	0	6	1	0
15	3	3	0	6	1	0
16	5	3	2			
17	11	1	0			
18	5	3	2	7	2	1
19	4	0	1			
20	3	1	1			
21	15	0	1			
22	0	0	0			

Hình 4.4: Bảng các thông số của các activity của mạng dự án 1

## Bộ dữ liệu 2

Sơ đồ mạng dự án:



Hình 4.5: Sơ đồ mạng dự án 2

Các thông số của các activity về time/d, resource1, resource2:

Activity	Operation mode 1			Operation mode 2		
	Time/d	Resource 1	Resource 2	Time/d	Resource 1	Resource 2
1	10	6	2			
2	30	6	1			
3	50	3	0			
4	50	3	0			
5	10	3	1			
6	60	6	2	20	1	1
7	60	6	2	100	2	1
8	50	3	1	100	4	2
9	40	6	0	70	1	1
10	90	3	2			
11	80	2	2	100	2	0
12	30	3	0	100	2	1
13	30	3	0			
14	50	3	2	60	1	0
15	150	3	2	60	6	2
16	50	3	2			
17	40	0	1	120	3	2
18	110	4	1			
19	90	6	2			

Hình 4.6: Bảng các thông số của các activity của mạng dự án 2

Ngoài ra còn một số tham số như giới hạn tài nguyên Resource1 và Resource2 lần lượt là 6 và 2 đơn vị. Đơn vị chi phí cho Resource1 là 1 triệu đôla và của Resource2 là 0.5 triệu đôla; lợi nhuận bằng 30% chi phí. Tỷ lệ chiết khấu của cả khách hàng và nhà thầu đều là 0.85% hàng tháng. Lãi suất cho vay hàng tháng của khách hàng là 0.9%.

## 4.2 Đánh giá kết quả

### Thiết bị tham gia vào quá trình demo sản phẩm

Máy tính Dell Inspiron với cấu hình:

- Bộ vi xử lý: *Intel(R) Core(TM)i3-2330M CPU @ 2.20GHz 2.20GHz*
- Ram: *4.00 GB (3.16GB có thể sử dụng)*
- Hệ điều hành: *Windows 8.1 Pro 32bit*

### Với bộ dữ liệu thứ nhất

Sau khi tiến hành chạy demo thuật toán với bộ dữ liệu thứ nhất Sau khi tiến hành chạy demo thuật toán với bộ dữ liệu thứ hai ta thu được hai bảng tối ưu hóa tiến độ thanh toán và bảng tối ưu hóa các hoạt động tham gia trong dự án.

Node	Time/d	Payment/%
1	0	5
2	3	5
3	9	0
4	29	0
5	35	10
6	50	15
7	56	15
8	82	50

Bảng: Tối ưu tiến độ thanh toán dự án

Activity	Starting time/d
2	0
3	3
4	9
5	69
6	6
7	23
8	39
9	67
10	19
11	9
12	29
13	74
14	29
15	33
16	45
17	45
18	54
19	50
20	59
21	59

Bảng: Tối ưu hóa việc điều phối các hoạt động trong dự án

Dựa vào hai bảng kết quả nói trên, ta đưa ra một vài nhận xét và tiến hành một số thử nghiệm:

- Thời gian mất để hoàn thành tất cả các công việc của dự án là 82. Điều phối ngược được áp dụng cho giải pháp cuối cùng dẫn đến sự trì hoãn của một số hoạt động như 5, 9 và 15.
- Nhận thấy hoạt động 12 được điều phối khá sớm. Vì vậy ta tiến hành thử nghiệm xem hoạt động 12 sẽ bị trì hoãn như thế nào khi chi phí thực hiện nó tăng lên. Khi chi phí thực hiện nó tăng thêm gấp 10 lần thì khi nó hoạt động 12 sẽ bị trì hoãn từ thời gian thực hiện 29 xuống 60. Ta thay đổi tương tự với hoạt động 4. Khi đó hoạt động 4 sẽ bị trì hoãn từ thời gian thực hiện 9 xuống thời gian thực hiện 36. Và khi đó thời gian thực hiện dự án sẽ tăng lên thành 90 và 84 tương ứng với sự thay đổi của hoạt động 12 và hoạt động 4.
- Khi ta tiến hành giảm nguồn lực, nguồn tài nguyên cần sử dụng ở chế độ 2 của hoạt động 7 hoặc chế độ 2 của hoạt động 8

Activity(Mode2)	Time/d	Resource1	Resource2
7	7	2	1
8	10	1	1

Khi đó các chế độ thứ 2 đã được lựa chọn mà không gây bất kỳ sự trì hoãn đáng kể nào trong quá trình thực hiện dự án. Và khoảng thời gian hoàn thành dự án sẽ lần lượt là 84 và 82

## Với bộ dữ liệu thứ hai

Sau khi tiến hành chạy demo thuật toán với bộ dữ liệu thứ hai ta thu được hai bảng tối ưu hóa tiến độ thanh toán và bảng tối ưu hóa các hoạt động tham gia trong dự án.

Node	Time/d	Payment/%
1	10	10
2	40	10
3	100	30
4	250	15
5	350	20
6	520	0
7	690	5
8	900	5
9	990	5

*Bảng: Tối ưu tiến độ thanh toán dự án*

Activity	Starting time/d
1	0
2	10
3	40
4	40
5	90
6	190
7	250
8	90
9	310
10	100
11	350
12	430
13	460
14	460
15	520
16	490
17	490
18	790
19	900

*Bảng: Tối ưu hóa việc điều phối các hoạt động trong dự án*



# Kết luận và định hướng phát triển

## Kết luận

Một mô hình trò chơi với thông tin đầy đủ được xây dựng nhằm giải quyết vấn đề tối ưu tiến độ thanh toán dự án với mức độ quan tâm đến cả nhà phát triển, đầu tư đến khách hàng, người chơi. Mục tiêu của mô hình là giảm thiểu độ lệch của giá trị hiện tại thuần NPV của dòng tiền trong thanh toán dự án từ giá trị có thể tối đa. Giải thuật di truyền được sử dụng để giải quyết cân bằng Nash và qua đó thu được một số kết quả.

- Sử dụng lý thuyết trò chơi để giải quyết vấn đề của tối ưu tiến độ thanh toán dự án là một cách thức đầy hứa hẹn để nghiên cứu tối ưu hóa quản lý dự án.
- Giải pháp win-win (hai bên cùng có lợi) cho cả nhà phát triển, đầu tư và khách hàng, người chơi
- Đó là phải giải quyết vấn đề tối ưu hóa quản lý dự án với các mô hình trò chơi, lý thuyết trò chơi theo cơ chế quản lý dự án mới.
- Là khả thi để giải quyết một vấn đề phức tạp bằng giải thuật di truyền.

## Định hướng phát triển

Định hướng phát triển đối với đề tài:

- Tiến hành cải thiện thuật toán di truyền về cách thức lai ghép, đột biến, cách lựa chọn các tham số phù hợp...
- Xây dựng mô hình bài toán giải quyết đa mục tiêu về thời gian, chi phí và nguồn lực.
- Triển khai cách thức giải quyết bài toán vào thực tế trong quá trình thanh toán dự án phần mềm thông qua việc xây dựng hoàn chỉnh một sản phẩm công nghệ thông tin.

# Tài liệu tham khảo

- [1] Học viện bưu chính viễn thông. *Giải thuật di truyền*
- [2] Yanf KK, Talbot FB, Patterson JH. *"Scheduling of activities to maximize the net present value of projects"*.
- [3] Sung CS, Lim SK. *A project activity scheduling problem with net present value measurement*.
- [4] DENG Ze-min , GAO Chun-ping and LI Zhong-xue. *"Optimization of project payment schedules with Nash equilibrium model"*
- [5] Wei-neng Chen and Jun Zhang. *"A Preference-Based Bi-Objective Approach to the Payment Scheduling Negotiation Problem with the Extended  $r$ -Dominance and NSGAII"*.
- [6] Massimo Orazio Spata and Salvatore Rinaudo. *"Merging nash equilibrium solution with Genetic algorithm: the game genetic algorithm"*.
- [7] Grefenstette JJ. *"Optimization of control parameters for genetic algorithms"*.
- [8] G, Ulusoy, S.Cebelli *"An equitable approach to the payment scheduling problem in project management"*.
- [9] Sivrikaya-Serfoglu *"A new uniform order-based crossover operator for genetic algorithm applications to multi-component combinatorial optimization problems"*.
- [10] Sebt, Fazel Zarandi, Alipouri *"Genetic algorithms to solve resource-constrained project scheduling problems with variable activity durations"*.
- [11] Meredith JR, Mantel SJ Jr. *"Project management: a managerial approach."*
- [12] Elmaghraby SE, Herroelen WS. *"The scheduling of activities to maximize the net present value of projects"*.