

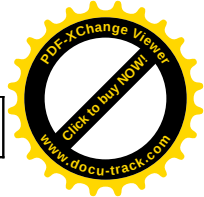
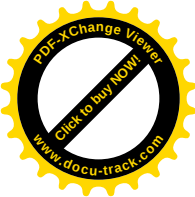
MỤC LỤC

	<i>Trang</i>
LỜI NÓI ĐẦU	6
NHẬP MÔN	8

PHẦN I

GIẢI QUYẾT VẤN ĐỀ BẰNG TÌM KIẾM

<i>Chương 1. CÁC CHIẾN THUẬT TÌM KIẾM MÙ</i>	16
1.1. Biểu diễn vấn đề trong không gian trạng thái.....	16
1.2. Các chiến lược tìm kiếm.....	19
1.3. Các chiến lược tìm kiếm mù.....	22
1.3.1. Tìm kiếm theo bề rộng.....	22
1.3.2. Tìm kiếm theo độ sâu.....	24
1.3.3. Các trạng thái lặp.....	25
1.3.4. Tìm kiếm sâu lặp.....	26
1.4. Quy vấn đề về các vấn đề con.	
Tìm kiếm trên đồ thị và/hoặc.....	27
1.4.1. Quy vấn đề về các vấn đề con.....	27
1.4.2. Đồ thị và/hoặc.....	30
1.4.3. Tìm kiếm trên đồ thị và/hoặc.....	34
<i>Chương 2. CÁC CHIẾN LƯỢC TÌM KIẾM KINH NGHIỆM</i>	36
2.1. Hàm đánh giá và tìm kiếm thiếu kinh nghiệm.....	36
2.2. Tìm kiếm tốt nhất - đầu tiên.....	37
2.3. Tìm kiếm leo đồi.....	40
2.4. Tìm kiếm BEAM.....	41

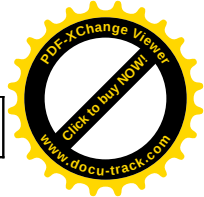
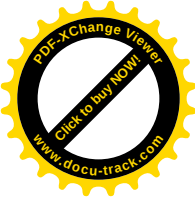


Chương 3. CÁC CHIẾN LƯỢC TÌM KIẾM TỐI ƯU	42
3.1. Tìm đường đi ngắn nhất	42
3.1.1. Thuật toán A*	44
3.1.2. Thuật toán tìm kiếm nhánh – và – cận	46
3.2. Tìm đối tượng tốt nhất	48
3.2.1. Tìm kiếm leo đồi	49
3.2.2. Tìm kiếm gradient	50
3.2.3. Tìm kiếm mô phỏng luyện kim	50
3.3. Tìm kiếm mô phỏng sự tiến hoá. Thuật toán di truyền	52
Chương 4. TÌM KIẾM CÓ ĐỐI THỦ	58
4.1. Cây trò chơi và tìm kiếm trên cây trò chơi	58
4.2. Chiến lược Minimax	60
4.3. Phương pháp cắt cụt alpha – beta	64

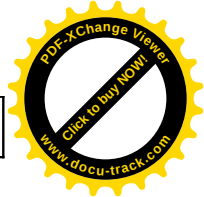
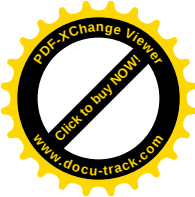
PHẦN II

TRI THỨC VÀ LẬP LUẬN

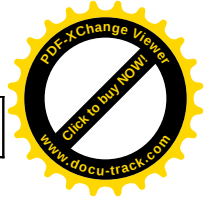
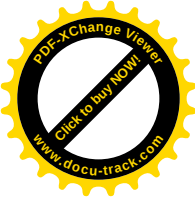
Chương 5. LOGIC MỆNH ĐỀ	69
5.1. Biểu diễn tri thức	69
5.2. Cú pháp và ngữ nghĩa của logic mệnh đề	71
5.2.1. Cú pháp	71
5.2.2. Ngữ nghĩa	72
5.3. Dạng chuẩn tắc	74
5.3.1. Sự tương đương của các công thức	74
5.3.2. Dạng chuẩn tắc	75
5.3.3. Các câu Horn	76
5.4. Luật suy diễn	77
5.5. Luật phân giải, chứng minh bác bỏ bằng luật phân giải	80



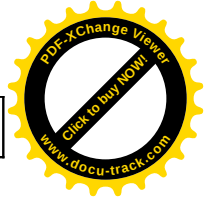
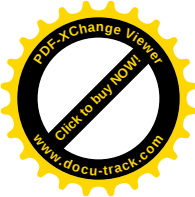
Chương 6. LOGIC VỊ TỪ CẤP MỘT	84
6.1. Cú pháp và ngữ nghĩa của logic vị từ cấp một	85
6.1.1. Cú pháp	85
6.1.2. Ngữ nghĩa	87
6.2. Chuẩn hoá các công thức	90
6.3. Các luật suy diễn	92
6.4. Thuật toán hợp nhất	95
6.5. Chứng minh bằng luật phân giải	98
6.6. Các chiến lược phân giải	103
6.6.1. Chiến lược phân giải theo bề rộng	105
6.6.2. Chiến lược phân giải sử dụng tập hỗ trợ	106
6.6.3. Chiến lược tuyến tính	107
6.7. Sử dụng logic vị từ cấp một để biểu diễn tri thức	107
6.7.1. Vị từ bằng	108
6.7.2. Danh sách và các phép toán trên danh sách	108
6.8. Xây dựng cơ sở tri thức	113
6.9. Cài đặt cơ sở tri thức	115
6.9.1. Cài đặt các hạng thức và các câu phân tử	116
6.9.2. Cài đặt cơ sở tri thức	119
Chương 7. BIỂU DIỄN TRI THỨC BỞI CÁC LUẬT VÀ LẬP LUẬN	122
7.1. Biểu diễn tri thức bởi các luật nếu – thì	122
7.2. Lập luận tiền và lập luận lùi trong các hệ dựa trên luật	124
7.2.1. Lập luận tiền	125
7.2.2. Lập luận lùi	128
7.2.3. Lập luận lùi như tìm kiếm trên độ thị và/hoặc	130
7.3. Thủ tục lập luận tiền	132
7.3.1. Thủ tục For_chain	133
7.3.2. Thủ tục rete	136



7.3.3. Hệ hành động dựa trên luật.....	143
7.4. Thủ tục lập luận lùi.....	147
7.5. Biểu diễn tri thức không chắc chắn.....	151
7.6. Hệ lập trình logic.....	153
7.7. Hệ chuyên gia.....	157
Chương 8. LOGIC KHÔNG ĐƠN ĐIỀU.....	159
8.1. Lập luận có thể xem xét lại và logic không đơn điều.....	159
8.2. Đặc điểm của logic không đơn điều.....	161
8.3. Logic mặc định.....	163
8.4. Giả thiết thế giới đóng.....	167
8.5. Bổ sung vị từ.....	169
8.6. Hạn chế phạm vi.....	171
Chương 9. LƯỚI NGỮ NGHĨA VÀ HỆ KHUNG.....	174
9.1. Ngôn ngữ mô tả khái niệm.....	174
9.2. Lưới ngữ nghĩa.....	176
9.3. Khung.....	181
Chương 10. TRI THỨC KHÔNG CHẮC CHẮN.....	186
10.1. Không chắc chắn và biểu diễn.....	187
10.2. Một số khái niệm cơ bản của lý thuyết xác suất.....	189
10.3. Mạng xác suất.....	197
10.3.1. Định nghĩa mạng xác suất.....	198
10.3.2. Vấn đề lập luận trong mạng xác suất.....	200
10.3.3. Khả năng biểu diễn của mạng xác suất.....	201
10.3.4. Sự độc lập của các biến trong mạng xác suất.....	204
10.4. Suy diễn trong mạng có cấu trúc cây.....	205
10.5. Mạng kết nối đơn.....	212
10.6. Suy diễn trong mạng đa kết nối.....	220
10.6.1. Suy diễn trong mạng đa kết nối.....	220



10.6.2. Biến đổi mạng đa kết nối thành mạng kết nối đơn.....	221
10.6.3. Phương pháp mô phỏng ngẫu nhiên.....	223
10.7. Lý thuyết quyết định.....	228
Chương 11. LOGIC MỜ VÀ LẬP LUẬN XẤP XỈ.....	234
11.1. Tập mờ.....	235
11.1.1. Khái niệm tập mờ.....	235
11.1.2. Một số khái niệm cơ bản liên quan đến tập mờ.....	239
11.1.3. Tính mờ và tính ngẫu nhiên.....	242
11.1.4. Xác định các hàm thuộc.....	243
11.2. Các phép toán trên tập mờ.....	248
11.2.1. Các phép toán chuẩn trên tập mờ.....	248
11.2.2. Các phép toán khác trên tập mờ.....	250
11.3. Quan hệ mờ và nguyên lý mở rộng.....	255
11.3.1. Quan hệ mờ.....	255
11.3.2. Hợp thành của các quan hệ mờ.....	256
11.3.3. Nguyên lý mở rộng.....	258
11.4. Logic mờ.....	259
11.4.1. Biến ngôn ngữ và mệnh đề mờ.....	259
11.4.2. Các mệnh đề hợp thành.....	262
11.4.3. Kéo theo mờ - Luật if-then mờ.....	264
11.4.4. Luật Modulus – Ponens tổng quát.....	267
11.5. Hệ mờ.....	270
11.5.1. Kiến trúc của hệ mờ.....	271
11.5.2. Cơ sở luật mờ.....	272
11.5.3. Bộ suy diễn mờ.....	273
11.5.4. Mờ hoá.....	275
11.5.5. Khử mờ.....	277
11.5.6. Hệ mờ là hệ tính xấp xỉ vạn năng.....	278
TÀI LIỆU THAM KHẢO.....	279



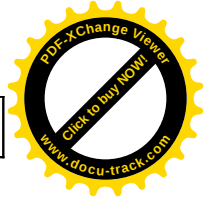
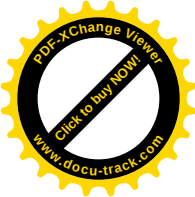
LỜI NÓI ĐẦU

Trí tuệ nhân tạo (TTNT) là một lĩnh vực của khoa học máy tính, nghiên cứu sự thiết kế của các tác nhân thông minh (“*Computational intelligence is the study of the design of intelligent agents*”). Các ứng dụng của TTNT rất đa dạng và phong phú, hiện nay đã có nhiều hệ thông minh ra đời: các hệ chuyên gia, các hệ điều khiển tự động, các robot, các hệ dịch tự động các ngôn ngữ tự nhiên, các hệ nhận dạng, các chương trình chơi cờ,... Kỹ thuật của TTNT đã được sử dụng trong việc xây dựng các hệ mềm, nhằm tạo ra các hệ mềm mang yếu tố thông minh, linh hoạt và tiện dụng. Ở nước ta, trong những năm gần đây, TTNT đã được đưa vào chương trình giảng dạy cho sinh viên các năm cuối ngành Tin học và Công nghệ thông tin. Cuốn sách này được hình thành trên cơ sở giáo trình TTNT mà chúng tôi đã giảng dạy cho sinh viên và các lớp cao học ngành Tin học và ngành Công nghệ thông tin trong các năm học từ 1997 tới nay, tại khoa Công nghệ thông tin, Đại học Khoa học tự nhiên, nay là khoa Công nghệ, Đại học Quốc gia Hà Nội.

Cuốn sách này được viết như một cuốn nhập môn về TTNT, đối tượng phục vụ chủ yếu của nó là sinh viên các ngành Tin học và Công nghệ thông tin. Nội dung cuốn sách gồm hai phần:

- **Phần 1:** Giải quyết vấn đề bằng tìm kiếm. Trong phần này, chúng tôi trình bày các phương pháp biểu diễn các vấn đề và các kỹ thuật tìm kiếm. các kỹ thuật tìm kiếm, đặc biệt là tìm kiếm kinh nghiệm (heuristic search), được sử dụng thường xuyên trong nhiều lĩnh vực nghiên cứu của TTNT.
- **Phần 2:** Biểu diễn tri thức và lập luận. Phần này đề cập đến các ngôn ngữ biểu diễn tri thức, đặc biệt là các logic và các phương pháp luận trong mỗi ngôn ngữ biểu diễn tri thức. Các kỹ thuật biểu diễn tri thức và lập luận đóng vai trò quan trọng trong việc thiết kế các hệ thông minh.

Tuy nhiên với mong muốn cuốn sách này có thể dùng làm tài liệu tham khảo cho một phạm vi rộng rãi các độc giả, chúng tôi cố gắng trình bày một cách hệ thống các khái niệm và các kỹ thuật cơ bản của TTNT, nhằm giúp cho độc giả có cơ sở để đi vào nghiên cứu các lĩnh vực chuyên sâu của TTNT, chẳng hạn như lập kế hoạch (planning), học máy (machine learning),



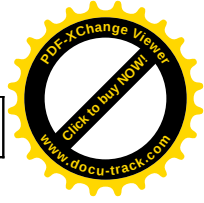
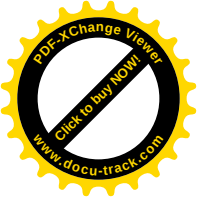
nhìn máy (computer vision), hiểu ngôn ngữ tự nhiên (natural language understanding).

Hai ngôn ngữ thao tác ký hiệu được sử dụng nhiều trong lập trình TTNT là Lisp và Prolog. Trong các sách viết về TTNT các năm gần đây, một số tác giả, chẳng hạn trong [5] và [7], đã sử dụng common Lisp để mô tả thuật toán. Trong 20, các tác giả lại sử dụng Prolog để biểu diễn thuật toán. Ở nước ta, các ngôn ngữ Lisp và Prolog được ít người biết đến, vì vậy chúng tôi biểu diễn các thuật toán trong sách này theo cách truyền thống. Tức là chúng tôi sử dụng các cấu trúc điều khiển (tuần tự, điều kiện, lặp) mà mọi người quen biết. Đặc biệt, chúng tôi sử dụng cấu trúc loop do <dãy câu> để biểu diễn rằng, <dãy câu> được thực hiện lặp. Toán tử exit để thoát khỏi vòng lặp, còn toán tử stop để dừng sự thực hiện thuật toán, các bạn có thể lựa chọn một trong các ngôn ngữ sau để sử dụng: Common Lisp, Scheme, Prolog, Smalltalk, C** hoặc ML (xem [28]).

Chúng tôi xin chân thành cảm ơn giáo sư Nguyễn Văn Hiệu, chủ nhiệm khoa Công nghệ, Đại học Quốc gia Hà Nội đã tạo điều kiện thuận lợi cho chúng tôi viết cuốn sách này.

Cuốn sách chắc chắn không tránh khỏi những thiếu sót. Chúng tôi mong nhận được sự góp ý của độc giả. Thư góp ý xin gửi về Bộ môn Khoa Học Máy Tính, Khoa Công Nghệ, Đại Học Quốc Gia Hà Nội.

Tác giả



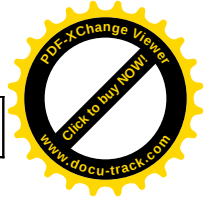
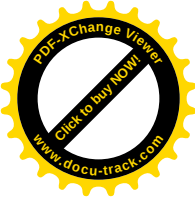
NHẬP MÔN

TRÍ TUỆ NHÂN TẠO LÀ GÌ?

Thuật ngữ **trí tuệ nhân tạo** (artificial intelligence) được Jonh McCarthy đưa ra trong hội thảo ở Dartmouth vào mùa hè 1956. Trong hội thảo này có mặt các tên tuổi nổi tiếng như Marvin Minsky, Claude Shannon, Nathaniel Rochester, Arthur Samuel, Allen Newell và Herbert Simon. Trước hội thảo này, từ năm 1952 Arthur Samuel đã viết chương trình chơi cờ. Samuel đã bác bỏ tư tưởng cho rằng máy tính chỉ có thể làm được cái mà người ta bảo nó làm, vì chương trình của Samuel có thể học để chơi tốt hơn người viết ra nó. Đến hội thảo này, Allen Newell và Herbert Simon cũng đã viết chương trình lập luận với tên gọi “the logic theorist”. Chương trình của các ông có khả năng chứng minh hầu hết các định lý trong chương 2 cuốn “Principia Mathematics” của Russell và Whitehead. Trong hội thảo ở Dartmouth, các nhà nghiên cứu đã thảo luận và vạch ra các phương hướng nghiên cứu của lĩnh vực Trí tuệ nhân tạo (TTNT). Vì vậy, hội thảo ở Dartmouth, mùa hè năm 1956 được xem là thời điểm ra đời thực sự của lĩnh vực nghiên cứu TTNT.

Trong các sách viết về TTNT các năm gần đây, các tác giả đưa ra nhiều định nghĩa về TTNT. Chúng tôi dẫn ra đây một số định nghĩa:

- “Sự nghiên cứu các năng lực trí tuệ thông qua việc sử dụng các mô hình tính toán” (“The study of mental faculties through the use of computational models” – Charniak and McDormott, 1985)
- “Nghệ thuật tạo ra các máy thực hiện các chức năng đòi hỏi sự thông minh khi được thực hiện bởi con người” (“The art of creating machines that perform functions that require intelligence when performed by people” – Kurzweil, 1990).
- “Lĩnh vực nghiên cứu tìm cách giải quyết và mô phỏng các hành vi thông minh trong thuật ngữ các quá trình tính toán” (“A field of study that seeks to explain and emulate intelligent behavior in terms of computational processes” – Schalkoff, 1990).
- “Sự nghiên cứu các tính toán để có thể nhận thức, lập luận và hành động” (“The study of computations that make it possible to perceive, reason, and act” – Winston, 1992).



- “Một nhánh của khoa học máy tính liên quan tới sự tự động hoá các hành vi thông minh” (“The branch of computer science that is concerned with the automation of intelligent behavior” – Luger and Stubblefield, 1993)

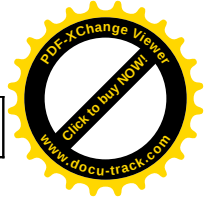
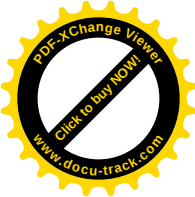
Sau đây là một số định nghĩa gần đây nhất:

- “TTNT là sự thiết kế và nghiên cứu các chương trình máy tính ứng xử một cách thông minh. Các chương trình này được xây dựng để thực hiện các hành vi mà khi ở người hoặc động vật chúng ta xem là thông minh” (“Artificial Intelligence is the design and study of computer programs that behave intelligently. These programs are constructed to perform as would a human or an animal whose behavior we consider intelligent” – Dean, Allen and Aloimonos, 1995).
- “ TTNT là sự nghiên cứu các tác nhân tồn tại trong môi trường, nhận thức và hành động” (“Artificial Intelligence is the design of agents that exists in an environment and act” – Russell and Norvig, 1995).
- “ TTNT là sự nghiên cứu ác thiết kế các tác nhân thông minh” (“Computational Intelligence is the study of the design of Intelligent agents” – Pulle, Mackworth and Goebel, 1998).

Hiện nay nhiều nhà nghiên cứu quan niệm rằng, TTNT là lĩnh vực nghiên cứu sự nghiên cứu các tác nhân thông minh (intelligent agent). Tác nhân thông minh là bất cứ ác gì tồn tại trong môi trường và hành động một cách thông minh (Một câu hỏi được đặt ra: hành động như thế nào thì được xem là thông minh?).

Mục tiêu khoa học của TTNT là hiểu được bản chất của các hành vi thông minh. mục tiêu thực tiễn, công nghệ của TTNT là xây dựng nân các hệ thông minh. Phương pháp luận nghiên cứu ở đây cũng tương tự như khi chúng ta nghiên cứu để hiểu được các nguyên lý bay, rồi thiết kế nên các máy biết bay (máy bay). Máy bay không phải là sự mô phỏng con chim, song nó có khả năng baybay tốt hơn chim.

Một số ngành khoa học khác, chẳng hạn như Triết học, Tâm lý học cũng quan tâm nghiên cứu các năng lực được xem của con người. Song khác với Triết học và Tâm lý học, TTNT là một ngành của khoa học máy tính – nghiên cứu sử lý thông tin bằng máy tính, do đó TTNT đặt ra mục tiêu nghiên cứu: làm thế nào thể hiện được các hành vi thông minh bằng thuật toán, rồi nghiên cứu các phương pháp cài đặt các chương trình có thể thực hiện được các hành vi thông minh. Như vậy chúng ta cần xây dựng các mô hình tính toán (Computational modeles phục vụ cho việc giải thích, mô tả

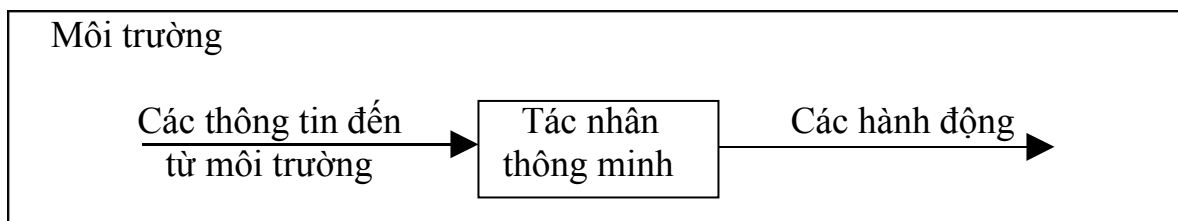


các hành vi thông minh bằng thuật toán, tiếp theo chúng ta cần chỉ ra tính hiệu quả, tính khả thi của thuật toán thực hiện một nhiệm vụ, và đưa ra các phương pháp cài đặt.

TÁC NHÂN THÔNG MINH

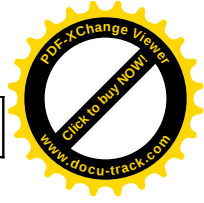
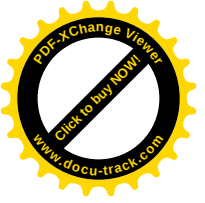
Tác nhân (agent) là bất cứ cái gì hành động trong môi trường. các tác nhân có thể là con người, con sâu, con con chó, robot infobot, ... Mục tiêu của TTNT là nhận cứu và thiết kế các tác nhân thông minh: các tác nhân tồn tại trong môi trường và hành động một cách thông minh.

Tồn tại trong môi trường, nên tác nhân thông minh cần có khả năng nhận thức được môi trường. Chẳng hạn, con người có thể nhận thức được môi trường nhờ tai, mắt và các giác quan khác. Để nhận thức được môi trường các robot sẽ được trang bị các **bộ cảm nhận** (sensors), đó là các thiết bị vật lý, chẳng hạn camera, các máy đo đạc,... Các tác nhân thông minh cần có các bộ tác động (effectors) để đưa ra các hành động đáp ứng môi trường. Với con người, đó là chân, tay và các bộ phận khác của thân thể. Với ác robot, đó có thể là các cánh tay robot,...



Chúng ta có thể xem tác nhân như một hộp đen, đầu vào là các thông tin nhận thức từ môi trường, đầu ra là các hành động thích ứng với môi trường, như trong hình trên. Bây giờ chúng ta xét xem cần phải cài đặt vào bên trong hộp đen cái gì để hành động đầu ra là hợp lý, thích ứng với các thông tin đầu vào.

Tác nhân cần có bộ nhớ để lưu giữ các tri thức chung về lĩnh vực, về môi trường mà nó được thiết kế để hoạt động trong lĩnh vực đó. Chẳng hạn, đối với robot lái taxi, tri thức về môi trường mà robot cần có là các tri thức về mạng lưới giao thông trong thành phố, về luật lệ giao thông,... Đối với các hệ chuyên gia trợ giúp chẩn đoán bệnh, các tri thức cần lưu là các tri thức của các bác sĩ về bệnh lý, về các phương án điều trị,... Bộ nhớ của tác nhân còn để ghi lại các tri thức mới mà tác nhân mới rút ra được trong quá trình hoạt động trong môi trường. Trong nhiều trường hợp, nó cũng cần lưu lại vết của các trạng thái của môi trường, bởi vì hành động thích hợp mà nó



cần đưa ra không chỉ phụ thuộc vào trạng thái hiện tại của môi trường mà còn phụ thuộc vào trạng thái của môi trường trong quá khứ.

Chúng ta cần trang bị cho tác nhân một chương trình: chương trình tác nhân (agent program). Chương trình này là sự mô tả thuật toán kết hợp với các thông tin về trạng thái của môi trường với ác tri thức đã được lưu để cho ra hành động thích ứng.

Bây giờ, chúng ta trả lời câu hỏi: “các tác nhân như thế nào thì được xem là thông minh?” bằng định nghĩa về thông minh của Turing: thử nghiệm Turing (Turing test).

THỬ NGHIỆM TURING

Alan Turing (1950) đã xác định các hành vi thông minh như là các hành vi trong các nhiệm vụ nhận thức đạt tới mức độ có thể đánh lừa được con người.

Sau đây là dạng tổng quát của thử nghiệm Turing. Một người hỏi là con người ngồi ở một phòng. Đối tác của người hỏi là một máy tính được đặt ở một phòng khác, hai bên trao đổi thông tin với nhau thông qua các phương tiện truyền tin hiện đại. Nếu máy tính có thể làm cho người hỏi tưởng lầm là có người khác đang nói chuyện với máy mình thì máy tính được xem là thông minh.

Bây giờ chúng ta xét xem, để thực hiện được ác hành vi được xem là thông minh, các tác nhân thông minh (TNTM) cần có khả năng gì?

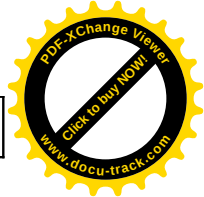
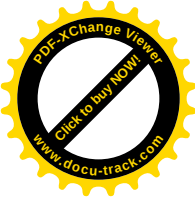
- TNTM cần có khả năng ghi nhớ và lập luận. nó sử dụng các tri thức đã lưu trữ, và bằng lập luận để rút ra những kết luận đáp ứng những câu hỏi mà người đặt ra.

Biểu diễn tri thức và lập luận là lĩnh vực nghiên cứu trung tâm của TTNT.

- Người hỏi có thể hỏi bằng ngôn ngữ tự nhiên, chẳng hạn tiếng Anh. Vì vậy, TNTM cần có khả năng biểu diễn ngôn ngữ tự nhiên.

Hiểu ngôn ngữ tự nhiên (natural language understanding) là một lĩnh vực nghiên cứu quan trọng của TTNT.

- Người hỏi có thể đưa ra một số hoàn cảnh và các hành động cần tiến hành tương ứng với mỗi hoàn cảnh đó, rồi đưa ra một hoàn cảnh mới và hỏi tác nhân cần phải làm gì trong hoàn cảnh đó. Để trả lời được câu hỏi này,



TNTM cần có khả năng học để có thể đưa ra hành động thích ứng với hoàn cảnh mới.

Học máy (machine learning) là một lĩnh vực nghiên cứu của TTNT đang phát triển mạnh mẽ và có nhiều ứng dụng trong các lĩnh vực khác nhau, chẳng hạn trong khám phá tri thức và khai thác dữ liệu.

- Người hỏi có thể đưa ra một số hình ảnh về các đối tượng và kiểm tra khả năng nhận biết của các đối tượng đó của TNTM.

Nhìn máy (computer vision) là lĩnh vực nghiên cứu để máy tính có thể hiểu được cấu trúc và các tính chất của các đối tượng trong không gian ba chiều từ các hình ảnh hai chiều.

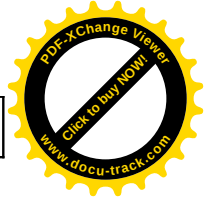
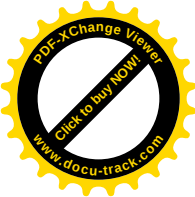
- Khi được cho các nhiệm vụ cần thực hiện, TNTM cần có khả năng đưa ra một dãy các hành động mà nó cần thực hiện để đạt được mục đích đó. Quá trình này được gọi là lập kế hoạch.

Lập kế hoạch (planning) là một lĩnh vực nghiên cứu quan trọng của TTNT.

BIỂU DIỄN VÀ LẬP LUẬN

Khi gặp một vấn đề cần giải quyết, một nhiệm vụ cần thực hiện, điều đầu tiên chúng ta phải làm là cần phải biểu diễn vấn đề, cần phải xác định cái gì tạo thành lời giải (nghiệm) của vấn đề, và sau đó mới đi tìm thuật toán tính nghiệm. Cần lưu ý rằng, biểu diễn vấn đề đóng vai trò quan trọng trong việc giải quyết một vấn đề. Trong nhiều trường hợp, khi chúng ta chọn được cách biểu diễn thích hợp cho một vấn đề thì vấn đề hầu như đã được giải quyết. Chúng ta có thể biểu diễn nhiều vấn đề bằng cách xác định trạng thái đích cần đạt tới và các hành động làm biến đổi các trạng thái của thế giới. Khi đó việc tìm nghiệm của vấn đề được quy về việc tìm kiếm một dãy các hành động dẫn tới trạng thái đích. Chú ý rằng, tìm kiếm đóng vai trò quan trọng trong việc giải quyết vấn đề và trong nhiều lĩnh vực nghiên cứu của TTNT, chẳng hạn như trong học máy và lập kế hoạch. Vì vậy, phần I của sách này dành cho việc trình bày phương pháp biểu diễn vấn đề trong không gian trạng thái và nghiên cứu các chiến lược tìm kiếm.

Thực tế cho thấy rằng, Để thực hiện được các nhiệm vụ khó khăn và phức tạp, con người cần sử dụng một khối lượng lớn tri thức về lĩnh vực liên quan. Một câu hỏi được đặt ra: tri thức là gì? Một cách không hình thức, chúng ta hiểu tri thức là thông tin liên quan tới một đối tượng, một hoàn cảnh, một lĩnh vực. Để máy tính có thể lưu trữ được tri thức, sử dụng được tri thức, chúng ta cần tìm các phương pháp biểu diễn tri thức.



Lập luận tự động được hiểu là quá trình tính toán trên các biểu diễn tri thức, mà khi cho đầu vào là các biểu diễn tri thức, chúng ta nhận được các đầu ra là các biểu diễn tri thức mới. Mục tiêu trung tâm của TTNT là nghiên cứu thiết kế các hệ thống minh, nó lưu trữ tri thức về lĩnh vực và có khả năng đưa ra hành động thích ứng bằng lập luận dựa trên các tri thức đã lưu trữ và các thông tin thu nhận từ môi trường. Do đó, vấn đề trung tâm mà chúng ta đề cập đến trong sách này là vấn đề biểu diễn tri thức và lập luận. Chúng ta sẽ nghiên cứu các mô hình biểu diễn tri thức khác nhau và các phương pháp lập luận trong mỗi mô hình đó.

CÁC ÁP DỤNG

Các áp dụng của TTNT là rất đa dạng: các hệ điều khiển tự động các quá trình sản xuất công nghiệp; các robot làm việc trong các môi trường đặc biệt, chẳng hạn các robot thám hiểm các hành tinh; các robot làm việc trong các điều kiện nguy hiểm đến tính mạng con người; các hệ chuyên gia trong các lĩnh vực; các hệ dịch tự động; các hệ nhận dạng; các infobot làm việc trong môi trường thuần túy thông tin; các chương trình chơi cờ,...

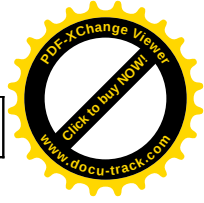
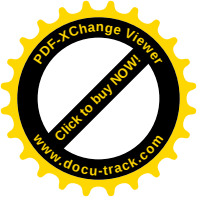
Sau đây chúng ta sẽ trình bày vài cách áp dụng. Qua các ví dụ này, chúng ta muốn chỉ ra những vấn đề cần giải quyết khi chúng ta định xây dựng một hệ thông minh.

- Robot đưa thư

Giả sử chúng ta muốn thiết kế một robot để phân phát thư, các gói nhỏ và phục vụ cà phê trong toà nhà làm việc của một công ty. Đương nhiên là robot phải được trang bị các bộ cảm nhận có thể “nhìn” và “nghe”, chẳng hạn video camera, các thiết bị thu thanh,... Nó cần có các bộ phận thực hiện hành động, chẳng hạn bánh xe chuyển động, “cánh tay” để nhặt lên, đặt xuống các đồ vật,...

Các thông tin đến từ môi trường ở đây là các hình ảnh và âm thanh, ... mà robot thu được nhờ các bộ cảm nhận. Khi nhận được mệnh lệnh, chẳng hạn “mang cà phê cho giám đốc”, nó phải đưa ra một dãy hành động đáp ứng môi trường nhằm đạt được mục đích: có cà phê cho giám đốc.

Các tri thức mà robot cần biết trước và được lưu trong cơ sở tri thức của robot là cấu trúc của toà nhà: Gồm có mấy tầng, các phòng trong mỗi tầng và hành lang được bố trí ra sao, thang máy ở đâu; các đối tượng mà robot có thể gặp trong toà nhà, ai làm việc ở đâu, cà phê ở phòng nào, ... Cơ sở tri thức của robot còn có thể chứa các tri thức cho biết các hoàn cảnh mà robot



thường gặp và các hành động mà robot cần thực hiện cách thích hợp sao cho thuận tiện cho việc lưu trữ, tìm kiếm và suy diễn.

Từ các thông tin hình ảnh, robot cần có khả năng hiểu được các đối tượng mà nó gặp để có thể suy ra các hành động thích hợp cần thực hiện.

Robot cần có khả năng hiểu được các mệnh lệnh bằng ngôn ngữ tự nhiên, từ đó suy ra các mục đích mà nó cần đạt tới.

Khi có nhiều mục đích cần đạt tới, robot cần có khả năng lập kế hoạch để đạt được các mục đích đó.

Robot cần có khả năng tìm đường đi (tối ưu nhất) giữa các vị trí trong toà nhà, từ các thông tin về sự bố trí và chức năng của các phòng và các bộ phận trong toà nhà.

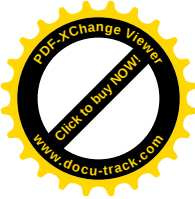
Chúng ta cũng cần trang bị cho robot có khả năng học để có thể rút kinh nghiệm, khái quát hoá từ thực tiễn trong quá trình phục vụ, và do đó, khi gặp các hoàn cảnh mới nó có thể đưa ra các hành động thích hợp.

- ***Hệ chuyên gia trong y học***

Các hệ chuyên gia y học có mục đích trợ giúp các bác sĩ trong chẩn đoán bệnh và điều trị. Môi trường của các hệ chuyên gia này là người bệnh. Các thông tin đến từ môi trường là các triệu chứng của người bệnh, các kết quả xét nghiệm từ người bệnh.

Cơ sở tri thức của các hệ chuyên gia này là các tri thức của các bác sĩ về bệnh lý, các tri thức này thường được biểu diễn dưới dạng các luật ***if-then***. Chẳng hạn, Feigenbaur và Shortiffe đã phát triển hệ MYCIN để chẩn đoán bệnh nhiễm trùng máu. Cơ sở luật của hệ MYCIN chứa khoảng 450 luật, mỗi luật gắn với một mức độ chắc chắn (điều này phù hợp với tính chất của các kết luận y học). Hệ MYCIN có khả năng làm việc tốt như các bác sĩ giỏi trong lĩnh vực.

Các hệ chuyên gia này có khả năng suy diễn để từ các triệu chứng, các kết quả xét nghiệm người bệnh và sử dụng các luật trong cơ sở luật để suy ra các kết luận về bệnh. Nó cũng có khả năng gợi ý cần kiểm tra cái gì, cần xét nghiệm cái gì ở người bệnh. Cũng như các bác sĩ, các hệ chuyên gia này có khả năng giải thích các kết luận mà nó đã đưa ra, nó có thể cho chúng ta biết tại sao nó đi đến kết luận như thế.



PHẦN I

GIẢI QUYẾT VẤN ĐỀ BẰNG TÌM KIẾM

Vấn đề tìm kiếm, một cách tổng quát, có thể hiểu là tìm một đối tượng thỏa mãn một số đòi hỏi nào đó, trong một tập hợp rộng lớn các đối tượng. Chúng ta có thể kể ra rất nhiều vấn đề mà việc giải quyết nó được quy về vấn đề tìm kiếm.

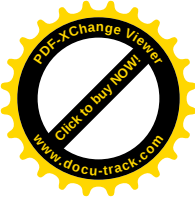
Các trò chơi, chẳng hạn cờ vua, cờ carô có thể xem như vấn đề tìm kiếm. Trong số rất nhiều nước đi được phép thực hiện, ta phải tìm ra các nước đi dẫn tới tình thế kết cuộc mà ta là người thắng.

Chứng minh định lý cũng có thể xem như vấn đề tìm kiếm. Cho một tập các tiên đề và các luật suy diễn, trong trường hợp này mục tiêu của ta là tìm ra một chứng minh (một dãy các luật suy diễn được áp dụng) để đưa đến công thức mà ta cần chứng minh.

Trong các lĩnh vực nghiên cứu của **Trí Tuệ Nhân Tạo**, chúng ta thường xuyên phải đối đầu với vấn đề tìm kiếm. Đặc biệt trong lập kế hoạch và học máy, tìm kiếm đóng vai trò quan trọng.

Trong phần này chúng ta sẽ nghiên cứu các kỹ thuật tìm kiếm cơ bản được áp dụng để giải quyết các vấn đề và được áp dụng rộng rãi trong các lĩnh vực nghiên cứu khác của **Trí Tuệ Nhân Tạo**. Chúng ta lần lượt nghiên cứu các kỹ thuật sau:

- Các kỹ thuật tìm kiếm mù, trong đó chúng ta không có hiểu biết gì về các đối tượng để hướng dẫn tìm kiếm mà chỉ đơn thuần là xem xét theo một hệ thống nào đó tất cả các đối tượng để phát hiện ra đối tượng cần tìm.
- Các kỹ thuật tìm kiếm kinh nghiệm (tìm kiếm heuristic) trong đó chúng ta dựa vào kinh nghiệm và sự hiểu biết của chúng ta về vấn đề cần giải quyết để xây dựng nên hàm đánh giá hướng dẫn sự tìm kiếm.
- Các kỹ thuật tìm kiếm tối ưu.
- Các phương pháp tìm kiếm có đối thủ, tức là các chiến lược tìm kiếm nước đi trong các trò chơi hai người, chẳng hạn cờ vua, cờ tướng, cờ carô.



CHƯƠNG 1

CÁC CHIẾN LƯỢC TÌM KIẾM MÙ

Trong chương này, chúng ta sẽ nghiên cứu các chiến lược tìm kiếm mù (blind search): tìm kiếm theo bề rộng (breadth-first search) và tìm kiếm theo độ sâu (depth-first search). Hiệu quả của các phương pháp tìm kiếm này cũng sẽ được đánh giá.

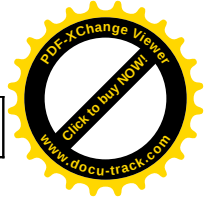
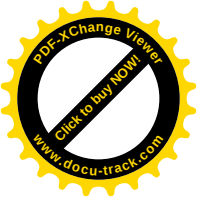
1.1 . BIỂU DIỄN VẤN ĐỀ TRONG KHÔNG GIAN TRẠNG THÁI

Một khi chúng ta muốn giải quyết một vấn đề nào đó bằng tìm kiếm, đầu tiên ta phải xác định không gian tìm kiếm. Không gian tìm kiếm bao gồm tất cả các đối tượng mà ta cần quan tâm tìm kiếm. Nó có thể là không gian liên tục, chẳng hạn không gian các vectơ thực n chiều; nó cũng có thể là không gian các đối tượng rời rạc.

Trong mục này ta sẽ xét việc biểu diễn một vấn đề trong không gian trạng thái sao cho việc giải quyết vấn đề được quy về việc tìm kiếm trong không gian trạng thái.

Một phạm vi rộng lớn các vấn đề, đặc biệt các câu đố, các trò chơi, có thể mô tả bằng cách sử dụng khái niệm trạng thái và toán tử (phép biến đổi trạng thái). Chẳng hạn, một khách du lịch có trong tay bản đồ mạng lưới giao thông nối các thành phố trong một vùng lãnh thổ (hình 1.1), du khách đang ở thành phố A và anh ta muốn tìm đường đi tới thăm thành phố B. Trong bài toán này, các thành phố có trong bản đồ là các trạng thái, thành phố A là trạng thái ban đầu, B là trạng thái kết thúc. Khi đang ở một thành phố, chẳng hạn ở thành phố D anh ta có thể đi theo các con đường để tới các thành phố C, F và G. Các con đường nối các thành phố sẽ được biểu diễn bởi các toán tử. Một toán tử biến đổi một trạng thái thành một trạng thái khác. Chẳng hạn, ở trạng thái D sẽ có ba toán tử dẫn trạng thái D tới các trạng thái C, F và G. Vấn đề của du khách bây giờ sẽ là tìm một dãy toán tử để đưa trạng thái ban đầu A tới trạng thái kết thúc B.

Một ví dụ khác, trong trò chơi cờ vua, mỗi cách bố trí các quân trên bàn cờ là một trạng thái. Trạng thái ban đầu là sự sắp xếp các quân lúc bắt đầu cuộc chơi. Mỗi nước đi hợp lệ là một toán tử, nó biến đổi một cảnh huống trên bàn cờ thành một cảnh huống khác.

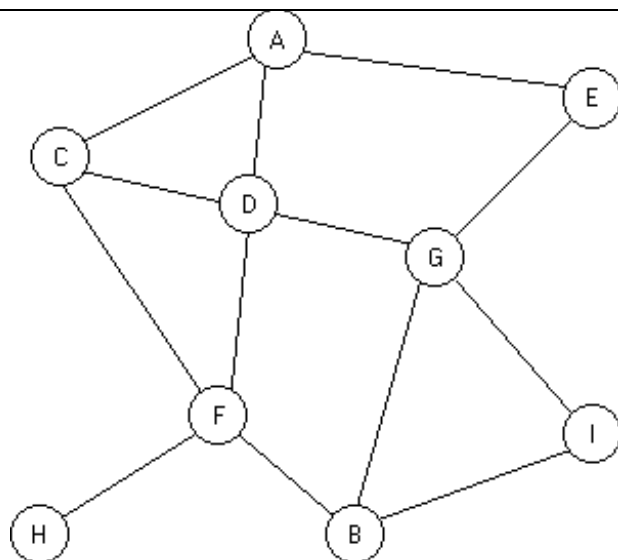


Như vậy muốn biểu diễn một vấn đề trong không gian trạng thái, ta cần xác định các yếu tố sau:

- Trạng thái ban đầu.
- Một tập hợp các toán tử. Trong đó mỗi toán tử mô tả một hành động hoặc một phép biến đổi có thể đưa một trạng thái tới một trạng thái khác.
Tập hợp tất cả các trạng thái có thể đạt tới từ trạng thái ban đầu bằng cách áp dụng một dãy toán tử, lập thành không gian trạng thái của vấn đề.
Ta sẽ ký hiệu không gian trạng thái là U , trạng thái ban đầu là u_0 ($u_0 \in U$). Mỗi toán tử R có thể xem như một ánh xạ $R: U \rightarrow U$. Nói chung R là một ánh xạ không xác định khắp nơi trên U .
- Một tập hợp T các trạng thái kết thúc (trạng thái đích). T là tập con của không gian U . Trong vấn đề của du khách trên, chỉ có một trạng thái đích, đó là thành phố B. Nhưng trong nhiều vấn đề (chẳng hạn các loại cờ) có thể có nhiều trạng thái đích và ta không thể xác định trước được các trạng thái đích. Nói chung trong phần lớn các vấn đề hay, ta chỉ có thể mô tả các trạng thái đích là các trạng thái thỏa mãn một số điều kiện nào đó.

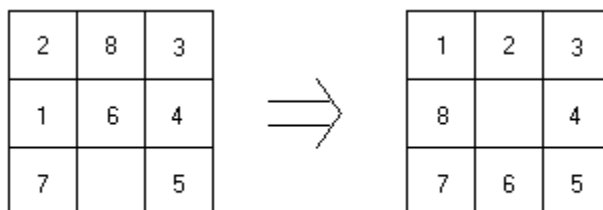
Khi chúng ta biểu diễn một vấn đề thông qua các trạng thái và các toán tử, thì việc tìm nghiệm của bài toán được quy về việc tìm đường đi từ trạng thái ban đầu tới trạng thái đích. (Một đường đi trong không gian trạng thái là một dãy toán tử dẫn một trạng thái tới một trạng thái khác).

Chúng ta có thể biểu diễn không gian trạng thái bằng đồ thị định hướng, trong đó mỗi đỉnh của đồ thị tương ứng với một trạng thái. Nếu có toán tử R biến đổi trạng thái u thành trạng thái v , thì có cung gán nhãn R đi từ đỉnh u tới đỉnh v . Khi đó một đường đi trong không gian trạng thái sẽ là một đường đi trong đồ thị này.



Hình 1.1 Tìm đường đi từ A đến B

Sau đây chúng ta sẽ xét một số ví dụ về các không gian trạng thái được xây dựng cho một số vấn đề.

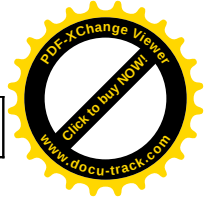
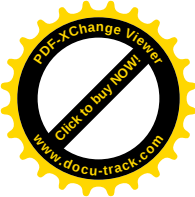


Hình 1.2 Trạng thái ban đầu và trạng thái kết thúc của bài toán số.

Ví dụ 1: Bài toán 8 số. Chúng ta có bảng 3x3 ô và tám quân mang số hiệu từ 1 đến 8 được xếp vào tám ô, còn lại một ô trống, chẳng hạn như trong hình 1.2 bên trái. Trong trò chơi này, bạn có thể chuyển dịch các quân ở cách ô trống tới ô trống đó. Vấn đề của bạn là tìm ra một dãy các chuyển dịch để biến đổi cảnh huống ban đầu (hình 1.2 bên trái) thành một cảnh huống xác định nào đó, chẳng hạn cảnh huống trong hình 1.2 bên phải.

Trong bài toán này, trạng thái ban đầu là cảnh huống ở bên trái hình 1.2, còn trạng thái kết thúc ở bên phải hình 1.2. Tương ứng với các quy tắc chuyển dịch các quân, ta có bốn toán tử: **up** (đẩy quân lên trên), **down** (đẩy quân xuống dưới), **left** (đẩy quân sang trái), **right** (đẩy quân sang phải). Rõ ràng là, các toán tử này chỉ là các toán tử bộ phận; chẳng hạn, từ trạng thái ban đầu (hình 1.2 bên trái), ta chỉ có thể áp dụng các toán tử **down**, **left**, **right**.

Trong các ví dụ trên việc tìm ra một biểu diễn thích hợp để mô tả các trạng thái của vấn đề là khá dễ dàng và tự nhiên. Song trong nhiều vấn đề



việc tìm được biểu diễn thích hợp cho các trạng thái của vấn đề là hoàn toàn không đơn giản. Việc tìm ra dạng biểu diễn tốt cho các trạng thái đóng vai trò hết sức quan trọng trong quá trình giải quyết một vấn đề. Có thể nói rằng, nếu ta tìm được dạng biểu diễn tốt cho các trạng thái của vấn đề, thì vấn đề hầu như đã được giải quyết.

Ví dụ 2: Vấn đề triệu phú và kẻ cướp. Có ba nhà triệu phú và ba tên cướp ở bên bờ tả ngạn một con sông, cùng một chiếc thuyền chở được một hoặc hai người. Hãy tìm cách đưa mọi người qua sông sao cho không để lại ở bên bờ sông kẻ cướp nhiều hơn triệu phú. Đương nhiên trong bài toán này, các toán tử tương ứng với các hành động chở 1 hoặc 2 người qua sông. Nhưng ở đây ta cần lưu ý rằng, khi hành động xảy ra (lúc thuyền đang bơi qua sông) thì ở bên bờ sông thuyền vừa dời chỗ, số kẻ cướp không được nhiều hơn số triệu phú. Tiếp theo ta cần quyết định cái gì là trạng thái của vấn đề. Ở đây ta không cần phân biệt các nhà triệu phú và các tên cướp, mà chỉ số lượng của họ ở bên bờ sông là quan trọng. Để biểu diễn các trạng thái, ta sử dụng bộ ba (a, b, k) , trong đó a là số triệu phú, b là số kẻ cướp ở bên bờ tả ngạn vào các thời điểm mà thuyền ở bờ này hoặc bờ kia, $k = 1$ nếu thuyền ở bờ tả ngạn và $k = 0$ nếu thuyền ở bờ hữu ngạn. Như vậy, không gian trạng thái cho bài toán triệu phú và kẻ cướp được xác định như sau:

- Trạng thái ban đầu là $(3, 3, 1)$.
- Các toán tử. Có năm toán tử tương ứng với hành động thuyền chở qua sông 1 triệu phú, hoặc 1 kẻ cướp, hoặc 2 triệu phú, hoặc 2 kẻ cướp, hoặc 1 triệu phú và 1 kẻ cướp.
- Trạng thái kết thúc là $(0, 0, 0)$.

1.2. CÁC CHIẾN LƯỢC TÌM KIẾM

Như ta đã thấy trong mục 1.1, để giải quyết một vấn đề bằng tìm kiếm trong không gian trạng thái, đầu tiên ta cần tìm dạng thích hợp mô tả các trạng thái của vấn đề. Sau đó cần xác định:

- Trạng thái ban đầu.
- Tập các toán tử.
- Tập T các trạng thái kết thúc. (T có thể không được xác định cụ thể gồm các trạng thái nào mà chỉ được chỉ định bởi một số điều kiện nào đó).

Giả sử u là một trạng thái nào đó và R là một toán tử biến đổi u thành v . Ta sẽ gọi v là trạng thái kế u , hoặc v được sinh ra từ trạng thái u bởi toán tử

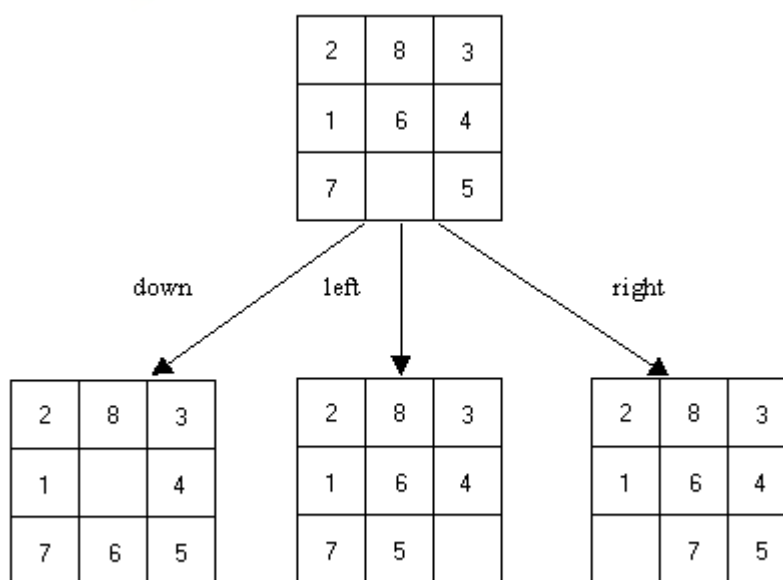
R. Quá trình áp dụng các toán tử để sinh ra các trạng thái kế u được gọi là phát triển trạng thái u. Chẳng hạn, trong bài toán tám số, phát triển trạng thái ban đầu (hình 1.2 bên trái), ta nhận được ba trạng thái kế (hình 1.3).

Khi chúng ta biểu diễn một vấn đề cần giải quyết thông qua các trạng thái và các toán tử thì việc tìm lời giải của vấn đề được quy về việc tìm đường đi từ trạng thái ban đầu tới một trạng thái kết thúc nào đó.

Có thể phân các chiến lược tìm kiếm thành hai loại:

- Các chiến lược tìm kiếm mù. Trong các chiến lược tìm kiếm này, không có một sự hướng dẫn nào cho sự tìm kiếm, mà ta chỉ phát triển các trạng thái ban đầu cho tới khi gặp một trạng thái đích nào đó. Có hai kỹ thuật tìm kiếm mù, đó là tìm kiếm theo bề rộng và tìm kiếm theo độ sâu.

Tư tưởng của tìm kiếm theo bề rộng là các trạng thái được phát triển theo thứ tự mà chúng được sinh ra, tức là trạng thái nào được sinh ra trước sẽ được phát triển trước. Còn trong tìm kiếm theo độ sâu, trạng thái sinh ra



Hình 1.3 Phát triển một trạng thái

sau cùng sẽ được phát triển trước.

Trong nhiều vấn đề, dù chúng ta phát triển các trạng thái theo hệ thống nào (theo bề rộng hoặc theo độ sâu) thì số lượng các trạng thái được sinh ra trước khi ta gặp trạng thái đích thường là cực kỳ lớn. Do đó các thuật toán tìm kiếm mù kém hiệu quả, đòi hỏi rất nhiều không gian và thời gian. Trong thực tế, nhiều vấn đề không thể giải quyết được bằng tìm kiếm mù.