

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC MỎ - ĐỊA CHẤT

BÁO CÁO SEMINAR

TÊN ĐỀ TÀI
TÁC TỬ THÔNG MINH

Người thực hiện: PGS. TS. Lê Văn Hưng (Mã cán bộ: 0801-01)

Đơn vị: Bộ môn Công nghệ phần mềm
Khoa Công nghệ Thông tin

Hà Nội - 2023

Mục lục

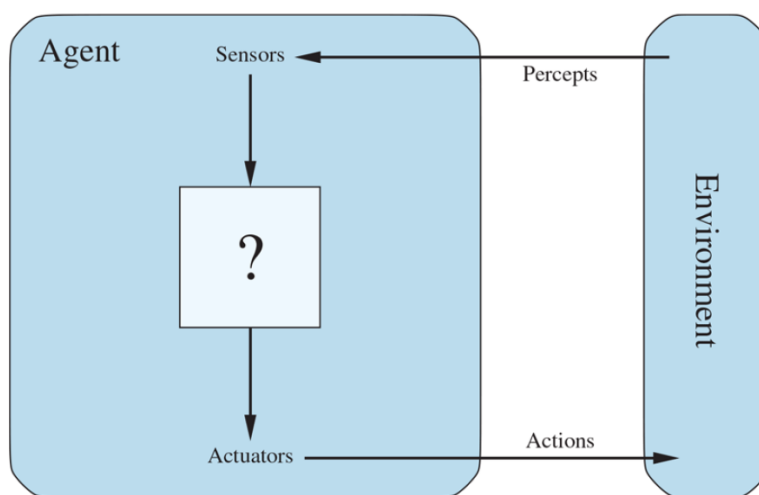
1	Mở đầu	2
2	Tác tử và môi trường.....	2
3	Hành vi tốt: Khái niệm về Tính hợp lý	4
3.1	Các chỉ số hiệu suất	4
3.2	Tính hợp lý.....	4
4	Tính chất của môi trường	6
4.1	Xác định môi trường nhiệm vụ.....	6
4.2	Các thuộc tính của môi trường nhiệm vụ.....	7
5	Cấu trúc của tác tử	10
5.1	Chương trình tác tử.....	11
5.2	Tác tử phản xạ đơn giản	11
5.3	Tác tử phản xạ dựa trên mô hình.....	13
5.4	Tác tử dựa trên mục tiêu.....	15
5.5	Tác tử dựa trên tính hữu ích	16
5.6	Tác tử có khả năng học.....	17
5.7	Cách thức hoạt động của các thành phần trong chương trình tác tử.....	20
6	Kết luận.....	21
	Tài liệu tham khảo	22

1 Mở đầu

Báo cáo này trình bày về bản chất của tác tử thông minh, môi trường và mối quan hệ giữa chúng. Qua quan sát, nhận thấy có những tác tử vượt trội so với những tác tử khác, từ đó xuất phát ý tưởng về tác tử hợp lý - tức là tác tử có hành vi tốt nhất có thể. Báo cáo sẽ trình bày một số nguyên tắc thiết kế để xây dựng các tác tử hợp lý, tức là các hệ thống được coi là thông minh. Hiệu suất hoạt động của tác tử phụ thuộc vào đặc điểm của môi trường, trong đó có những môi trường khó hơn những môi trường khác. Báo cáo cũng sẽ cung cấp một phân loại sơ bộ về các loại môi trường và cách thuộc tính của môi trường ảnh hưởng đến việc thiết kế tác tử phù hợp cho từng môi trường, đồng thời cũng sẽ giới thiệu một số mô hình tác tử cơ bản.

2 Tác tử và môi trường

Một tác tử là một thực thể có khả năng cảm nhận môi trường thông qua cảm biến và thực hiện hành động tác động lên môi trường. Ý tưởng đơn giản này được minh họa trong Hình 1. Tác tử con người sử dụng mắt, tai và các cơ quan khác để làm cảm biến, và tay, chân, miệng và các bộ phận khác để thực hiện hành động. Một tác tử robot có thể trang bị camera và cảm biến hồng ngoại để cảm nhận môi trường, và các động cơ để thực hiện hành động. Một tác tử phần mềm có thể nhận dữ liệu từ tệp, gói dữ liệu mạng và thông tin nhập từ con người (bàn phím/chuột/màn hình cảm ứng/âm thanh) làm cảm biến và tác động lên môi trường bằng cách ghi ra tệp, gửi gói dữ liệu mạng và hiển thị thông tin hoặc tạo ra âm thanh. Môi trường có thể là bất cứ thứ gì. Trong thực tế, nó chỉ là một phần của vũ trụ mà chúng ta quan tâm đến trạng thái khi thiết kế tác tử - phần mà tác tử cảm nhận và bị tác động bởi các hành động của nó.

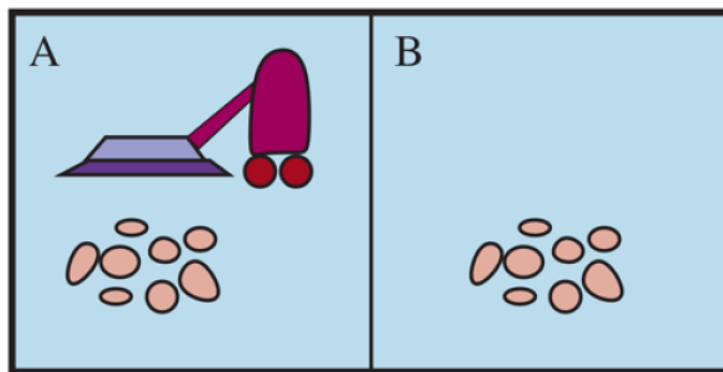


Hình 1. Tác tử và môi trường

Ta sử dụng thuật ngữ "cảm nhận" để chỉ nội dung mà các cảm biến của tác tử cảm nhận được. Dãy cảm nhận của tác tử là toàn bộ lịch sử của mọi thứ mà tác tử đã từng cảm nhận. Nói chung, việc tác tử lựa chọn hành động tại bất kỳ thời điểm nào phụ thuộc vào kiến thức tích hợp sẵn trong nó và toàn bộ dãy cảm nhận đã quan sát cho đến thời điểm hiện tại. Theo quy tắc toán học, ta nói rằng hành vi của một tác tử được mô tả bằng hàm tác tử, ánh xạ một dãy cảm nhận bất kỳ cho trước thành một hành động.

Ta có thể tạo một bảng hàm tác tử để mô tả hoạt động của bất kỳ tác tử nào; đối với hầu hết tác tử, bảng này sẽ rất lớn - vô hạn, trừ khi ta giới hạn độ dài của dãy cảm nhận. Bảng này sẽ ánh xạ mọi dãy cảm nhận sang hành động mà tác tử thực hiện. Bên trong, hàm tác tử sẽ được thực hiện thông qua một chương trình tác tử. Đây là hai khái niệm khác nhau. Hàm tác tử là một mô tả toán học trừu tượng; chương trình tác tử là một cài đặt cụ thể, chạy trên một hệ thống vật lý.

Để minh họa những ý tưởng này, ta sử dụng một ví dụ đơn giản - máy hút bụi, bao gồm một tác tử máy hút bụi (robot) trong một môi trường gồm các ô vuông bị bẩn hoặc sạch. Hình 2 hiển thị một cấu hình với chỉ hai ô vuông. Tác tử máy hút bụi cảm nhận được ô vuông mà nó đang ở và có bụi trong ô đó hay không. Các hành động có thể thực hiện là di chuyển sang phải, di chuyển sang trái, hút bụi hoặc không làm gì cả. Một hàm tác tử rất đơn giản có thể là như sau: nếu ô hiện tại bị bẩn, thì hút bụi; ngược lại, di chuyển sang ô khác. Một phần của bảng hàm tác tử này được thể hiện trong Hình 3 và một chương trình tác tử thực hiện nó nằm trong Hình 4.



Hình 2. Tác tử máy hút bụi đơn giản

Percept sequence	Action
[A, Clean]	Right
[A, Dirty]	Suck
[B, Clean]	Left
[B, Dirty]	Suck
[A, Clean], [A, Clean]	Right
[A, Clean], [A, Dirty]	Suck
⋮	⋮
[A, Clean], [A, Clean], [A, Clean]	Right
[A, Clean], [A, Clean], [A, Dirty]	Suck
⋮	⋮

Hình 3. Bảng hàm tác tử máy hút bụi đơn giản

function REFLEX-VACUUM-AGENT([location,status]) **returns** an action

if status = Dirty **then return** Suck
else if location = A **then return** Right
else if location = B **then return** Left

Hình 4. Chương trình tác tử hút bụi đơn giản

3 Hành vi tốt: Khái niệm về Tính hợp lý

Một tác tử hợp lý là tác tử luôn làm điều đúng.

3.1 Các chỉ số hiệu suất

Chủ nghĩa kết quả, một nguyên tắc phổ biến trong lĩnh vực Trí tuệ Nhân tạo, đánh giá hành vi của tác tử dựa trên kết quả mà nó mang lại. Khi đặt một tác tử vào một môi trường, nó sẽ tạo ra một chuỗi hành động dựa trên cảm nhận của nó. Chuỗi hành động này dẫn đến một chuỗi trạng thái thay đổi của môi trường. Nếu chuỗi trạng thái này đúng như mong muốn, tác tử được coi là hoạt động tốt. Khái niệm mong muốn được biểu thị thông qua một chỉ số hiệu suất dùng để đánh giá từng chuỗi trạng thái của môi trường.

Con người có những mong muốn và sở thích riêng, do đó, khái niệm về tính hợp lý áp dụng cho con người liên quan đến việc thành công trong việc lựa chọn hành động tạo ra chuỗi trạng thái môi trường mà từ góc nhìn của họ là mong muốn. Máy móc, ngược lại, không có những mong muốn và sở thích riêng của chúng; chỉ số hiệu suất ban đầu nằm trong suy nghĩ của người thiết kế hoặc trong suy nghĩ của người dùng. Ta sẽ thấy rằng một số thiết kế tác tử có biểu diễn rõ ràng về chỉ số hiệu suất, trong khi ở các thiết kế khác, chỉ số hiệu suất được ngầm định hoàn toàn - tác tử có thể hoạt động đúng, nhưng không biết lý do tại sao.

Có thể thấy rằng việc xây dựng một chỉ số hiệu suất chính xác là khá khó khăn. Ví dụ, đối với tác tử máy hút bụi đã được đề cập ở phần trước, ta có thể đo lường hiệu suất bằng lượng bụi được làm sạch trong một ca làm việc tám giờ. Tuy nhiên, một tác tử hợp lý có thể tối đa hóa chỉ số hiệu suất này bằng cách làm sạch bụi, sau đó đổ toàn bộ bụi xuống sàn, sau đó lại làm sạch lại, và tiếp tục như vậy. Vì vậy, cần thiết lập một chỉ số hiệu suất phù hợp hơn, ví dụ như thưởng cho tác tử nếu sàn nhà được giữ sạch. Một cách thực hiện điều này đó là cho điểm cho mỗi ô vuông sạch tại mỗi bước thời gian (có thể áp dụng một khoản phạt cho việc tiêu thụ điện năng và tạo ra tiếng ồn). Quy tắc chung là thiết kế chỉ số hiệu suất dựa trên những gì ta thực sự muốn tác tử đạt được, thay vì dựa trên những gì ta nghĩ rằng tác tử nên làm.

3.2 Tính hợp lý

Tính hợp lý tại một thời điểm nhất định phụ thuộc vào bốn yếu tố sau:

1. Chỉ số hiệu suất xác định tiêu chí thành công.
2. Kiến thức có sẵn của tác tử về môi trường.
3. Các hành động mà tác tử thực hiện.
4. Chuỗi cảm nhận tác tử nhận được cho đến thời điểm hiện tại.

Điều này dẫn đến định nghĩa sau về tác tử hợp lý:

Đối với mỗi chuỗi cảm nhận, tác tử hợp lý chọn hành động để tối đa hóa chỉ số hiệu suất của nó, dựa trên các bằng chứng được cung cấp bởi chuỗi cảm nhận và kiến thức tích hợp sẵn trong tác tử.

Xét tác tử máy hút bụi đơn giản, làm sạch một ô vuông nếu nó bẩn và di chuyển sang ô vuông khác nếu ngược lại. Giả sử những giả định sau đây là đúng:

- Chỉ số hiệu suất sẽ thưởng một điểm cho mỗi ô sạch tại mỗi bước thời gian, trong suốt vòng đời gồm 1000 bước thời gian.

- Cấu hình của môi trường đã biết trước như ở Hình 2 nhưng chưa biết phân bố bụi và vị trí ban đầu của tác tử. Các ô sạch vẫn luôn sạch và hút bụi làm sạch ô hiện tại. Các hành động di chuyển sang trái/sang phải làm cho tác tử di chuyển một ô trừ khi điều này làm cho tác tử di chuyển ra ngoài phạm vi môi trường, trong trường hợp đó tác tử sẽ ở tại chỗ.

- Tác tử chỉ có các hành động di chuyển sang trái, di chuyển sang phải và hút bụi.

- Tác tử nhận biết chính xác vị trí của mình và ở vị trí đó có bụi hay không.

Với các điều kiện như trên, tác tử máy hút bụi đúng là một tác tử hợp lý.

Dễ nhận thấy rằng cùng một tác tử có thể trở nên không hợp lý trong các tình huống khác nhau. Ví dụ, một khi tất cả bụi ở các ô được làm sạch, tác tử sẽ lặp đi lặp lại các hành động một cách vô ích; và nếu chỉ số hiệu suất bao gồm việc phạt một điểm cho mỗi hành động di chuyển, tác tử sẽ không đạt được kết quả tốt. Một tác tử tốt hơn cho trường hợp này sẽ không làm gì sau khi đã chắc chắn rằng tất cả các ô đều sạch. Nếu các ô sạch có thể bị bẩn trở lại, tác tử sẽ định kỳ kiểm tra và làm sạch lại nếu cần thiết. Nếu tác tử chưa biết địa hình của môi trường, nó sẽ cần phải khám phá.

Dễ nhận thấy rằng cùng một tác tử có thể trở nên không hợp lý trong các tình huống khác nhau. Ví dụ, khi tất cả bụi ở các ô đã được làm sạch, tác tử sẽ lặp đi lặp lại các hành động một cách vô ích; và nếu chỉ số hiệu suất bao gồm việc phạt một điểm cho mỗi hành động di chuyển, tác tử sẽ không đạt được kết quả tốt. Một tác tử tốt hơn cho trường hợp này sẽ không thực hiện bất kỳ hành động nào sau khi đã chắc chắn rằng tất cả các ô đều sạch. Nếu các ô sạch có thể bị bẩn lại, tác tử sẽ định kỳ kiểm tra và làm sạch lại nếu cần thiết. Trong trường hợp tác tử chưa biết địa hình của môi trường, nó sẽ cần phải tiến hành khám phá.

Định nghĩa của ta yêu cầu một tác tử hợp lý không chỉ thu thập thông tin mà còn học hỏi càng nhiều càng tốt từ những gì nó cảm nhận được. Cấu hình ban đầu của tác tử có thể phản ánh một số kiến thức có trước về môi trường, nhưng khi tác tử tích lũy kinh nghiệm, những kiến thức này có thể được điều chỉnh và bổ sung. Có những trường hợp trong đó môi trường được biết trước hoàn toàn và có thể dự đoán được hoàn toàn. Trong những trường hợp như vậy, tác tử không cần phải cảm nhận hoặc học hỏi; nó chỉ đơn giản là hành động đúng.

Nếu một tác tử phụ thuộc vào kiến thức tích hợp trước của người thiết kế hơn là từ những gì nó cảm nhận được và quá trình học của nó, ta nói rằng tác tử đó thiếu tính tự chủ. Một tác tử hợp lý nên có tính tự chủ - nó nên học hỏi để bù đắp cho những kiến thức có sẵn không được đầy đủ hoặc không chính xác. Ví dụ, nếu một tác tử máy hút bụi học được cách dự đoán nơi và thời điểm bụi sẽ sinh ra sẽ làm tốt hơn so với một tác tử không làm được như vậy.

Trong thực tế, hiếm khi ta yêu cầu tính tự chủ hoàn toàn ngay từ đầu: khi tác tử chưa có kinh nghiệm hoặc ít kinh nghiệm, nó sẽ hoạt động ngẫu nhiên trừ khi nhà thiết kế cung cấp một số hỗ trợ. Tương tự như tiến hóa cung cấp cho động vật những phản xạ có sẵn đủ để chúng tồn tại đủ lâu để tự học, việc cung cấp kiến thức ban đầu và khả năng học cho một tác tử trí tuệ nhân tạo cũng là điều hợp lý. Sau khi có đủ kinh nghiệm về môi trường, hành vi của tác tử có thể trở nên độc lập, hiệu quả hơn so với kiến thức trước đó. Do đó, việc tích hợp khả năng học cho phép ta thiết kế tác tử hợp lý sẽ thành công trong nhiều môi trường khác nhau.

4 Tính chất của môi trường

Trong phần này, trước tiên, ta sẽ nói về "môi trường nhiệm vụ" (task environment), đó chính là những "vấn đề" mà các tác tử hợp lý đóng vai trò "giải pháp". Ta sẽ minh họa quá trình xác định một môi trường nhiệm vụ thông qua một số ví dụ. Môi trường nhiệm vụ có nhiều loại khác nhau. Tính chất của môi trường nhiệm vụ trực tiếp ảnh hưởng đến thiết kế phù hợp cho chương trình tác tử.

4.1 Xác định môi trường nhiệm vụ

Khi thảo luận về tính hợp lý của tác tử máy hút bụi đơn giản, ta cần xác định chỉ số hiệu suất, môi trường, cảm biến và bộ điều khiển của nó. Tất cả các yếu tố này được tổng hợp lại thành môi trường nhiệm vụ. Ta gọi điều này là mô tả PEAS (Performance, Environment, Actuators, Sensors). Trong quá trình thiết kế tác tử, bước đầu tiên luôn là xác định môi trường nhiệm vụ một cách cụ thể nhất có thể.

Máy hút bụi là một ví dụ đơn giản; ta xem xét một vấn đề phức tạp hơn: một taxi tự hành. Hình 5 tóm tắt mô tả PEAS cho môi trường nhiệm vụ của taxi tự hành. Ta sẽ thảo luận về từng yếu tố một cách chi tiết hơn.

Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi driver	Safe, fast, legal, comfortable trip, maximize profits, minimize impact on other road users	Roads, other traffic, police, pedestrians, customers, weather	Steering, accelerator, brake, signal, horn, display, speech	Cameras, radar, speedometer, GPS, engine sensors, accelerometer, microphones, touchscreen

Hình 5. Mô tả PEAS về môi trường nhiệm vụ cho một taxi tự hành

Trước hết, chỉ số hiệu suất mà ta mong muốn taxi tự hành đạt được bao gồm: đến đúng đích; giảm tiêu thụ nhiên liệu và hao mòn; giảm thời gian hoặc chi phí chuyển đi; giảm vi phạm luật giao thông và gây phiền hà cho các tài xế khác; tối đa hóa an toàn và sự thoải mái của hành khách; tối đa hóa lợi nhuận. Rõ ràng, một số mục tiêu này có xung đột với nhau và do đó cần phải có sự thỏa hiệp.

Tiếp theo, môi trường lái xe mà taxi tự hành sẽ đối mặt bao gồm sự đa dạng về loại đường, từ các con đường nông thôn và những ngõ hẻm đô thị đến các tuyến đường cao tốc nhiều làn. Trên các con đường này, taxi sẽ gặp phải các yếu tố khác nhau như người đi bộ, động vật hoang dã, các công trường đang thi công, xe cảnh sát, vũng nước và ổ gà. Ngoài ra, taxi tự hành cũng phải tương tác với các khách hàng tiềm năng và thực tế. Cũng có một số lựa chọn tùy chọn. Taxi có thể hoạt động ở Nam California, nơi tuyết hiếm khi rơi xuống, hoặc ở Alaska, nơi hiếm khi không có tuyết. Nó cũng có thể luôn đi bên phải hoặc đủ linh hoạt để đi bên trái khi ở Anh hoặc Nhật Bản. Rõ ràng rằng, môi trường càng bị hạn chế, thì việc thiết kế càng dễ dàng hơn.

Các bộ điều khiển cho một chiếc taxi tự hành bao gồm các bộ điều khiển tương tự như

của người lái: điều khiển động cơ thông qua chân ga, điều khiển vô lăng và phanh. Ngoài ra, nó cần có đầu ra để hiển thị trên màn hình hoặc tổng hợp giọng nói để nói chuyện với hành khách và có thể có một số phương thức giao tiếp với các phương tiện khác.

Các cảm biến cơ bản cho taxi tự hành bao gồm một hoặc nhiều camera hình ảnh, cũng như các cảm biến lidar và siêu âm để xác định khoảng cách đến các xe khác và chướng ngại vật. Để tránh việc bị phạt vi phạm tốc độ, taxi nên có đồng hồ đo tốc độ và để điều khiển phương tiện một cách chính xác, đặc biệt là trên các đoạn đường cong, nó nên có cảm biến gia tốc. Để xác định tình trạng cơ khí của chiếc xe, nó sẽ cần các cảm biến thông thường của động cơ, nhiên liệu và hệ thống điện. Ngoài ra, nó có thể muốn truy cập tín hiệu GPS để không bị lạc đường. Cuối cùng, nó sẽ cần màn hình cảm ứng hoặc đầu vào bằng giọng nói để hành khách yêu cầu điểm đến.

Hình 6 phác thảo các yếu tố cơ bản PEAS cho một số loại tác tử khác. Các ví dụ này bao gồm cả môi trường vật lý và môi trường ảo.

Agent Type	Performance Measure	Environment	Actuators	Sensors
Medical diagnosis system	Healthy patient, reduced costs	Patient, hospital, staff	Display of questions, tests, diagnoses, treatments	Touchscreen/voice entry of symptoms and findings
Satellite image analysis system	Correct categorization of objects, terrain	Orbiting satellite, downlink, weather	Display of scene categorization	High-resolution digital camera
Part-picking robot	Percentage of parts in correct bins	Conveyor belt with parts; bins	Jointed arm and hand	Camera, tactile and joint angle sensors
Refinery controller	Purity, yield, safety	Refinery, raw materials, operators	Valves, pumps, heaters, stirrers, displays	Temperature, pressure, flow, chemical sensors
Interactive English tutor	Student's score on test	Set of students, testing agency	Display of exercises, feedback, speech	Keyboard entry, voice

Hình 6. Các ví dụ về các loại tác tử và mô tả PEAS của chúng

4.2 Các thuộc tính của môi trường nhiệm vụ

Phạm vi của các môi trường nhiệm vụ trong lĩnh vực Trí tuệ Nhân tạo rất rộng lớn. Tuy nhiên, ta có thể phân loại chúng dựa trên một số yếu tố quan trọng. Những yếu tố này sẽ định rõ thiết kế phù hợp cho tác tử cũng như sự áp dụng của các nhóm kỹ thuật chính trong việc triển khai tác tử. Đầu tiên, ta sẽ liệt kê các yếu tố này, sau đó phân tích một số môi trường nhiệm vụ cụ thể để minh họa ý tưởng trên.

QUAN SÁT ĐƯỢC ĐẦY ĐỦ và QUAN SÁT ĐƯỢC MỘT PHẦN: Nếu các cảm biến của một tác tử cho phép nó tiếp cận được toàn bộ trạng thái của môi trường tại mỗi thời điểm, ta nói rằng môi trường nhiệm vụ là quan sát được đầy đủ. Một môi trường nhiệm vụ được coi là quan sát được đầy đủ nếu các cảm biến phát hiện được tất cả các khía cạnh quan trọng

(phụ thuộc vào chỉ số hiệu suất) liên quan đến việc lựa chọn hành động. Môi trường quan sát được đầy đủ rất tiện lợi vì tác tử không cần duy trì bất kỳ trạng thái bên trong nào để theo dõi thế giới xung quanh. Một môi trường có thể chỉ quan sát được một phần do các cảm biến không chính xác hoặc nhiễu, hoặc vì một số phần của trạng thái bị thiếu trong dữ liệu cảm biến - ví dụ, một tác tử máy hút bụi chỉ có cảm biến bụi tại chỗ không thể biết được có bụi ở các ô khác, và một chiếc taxi tự hành không thể biết được suy nghĩ của các tài xế khác. Nếu tác tử không có cảm biến nào, thì môi trường không thể quan sát được. Mặc dù có thể nghĩ rằng trong những trường hợp như vậy, tác tử là không có hy vọng, nhưng đôi khi, tác tử vẫn có thể đạt được mục tiêu.

ĐƠN TÁC TỬ và ĐA TÁC TỬ: Sự phân biệt giữa môi trường đơn tác tử và môi trường đa tác tử đôi khi có vẻ đơn giản. Ví dụ, khi một tác tử giải câu đố chữ một mình, rõ ràng nó đang hoạt động trong một môi trường đơn tác tử, trong khi một tác tử chơi cờ vua hoạt động trong một môi trường đa tác tử. Tuy nhiên, có một số trường hợp tinh vi hơn. Đầu tiên, ta đã mô tả cách một thực thể có thể được coi là một tác tử, nhưng ta chưa giải thích những thực thể nào cần được coi là tác tử. Liệu tác tử A (ví dụ như một chiếc taxi tự hành) có cần coi đối tượng B (một phương tiện khác) là một tác tử, hay chỉ xem nó là một đối tượng tuân theo các quy luật vật lý, tương tự như sóng biển hay lá cuốn trong gió? Cách phân biệt đó là liệu hành vi của B có thể được mô tả tốt nhất bằng cách tối đa hóa một chỉ số hiệu suất mà giá trị của nó phụ thuộc vào hành vi của tác tử A.

Ví dụ, trong cờ vua, thực thể đối thủ B cố gắng tối đa hóa chỉ số hiệu suất của mình, trong khi theo quy tắc của cờ vua, điều này làm giảm chỉ số hiệu suất của tác tử A. Do đó, cờ vua được coi là một môi trường đa tác tử cạnh tranh. Trong khi đó, trong môi trường lái xe taxi tự động, việc tránh va chạm tối đa hóa chỉ số hiệu suất của tất cả các tác tử, vì vậy đây là một môi trường đa tác tử hợp tác một phần. Nó cũng có tính cạnh tranh một phần, ví dụ như khi tranh giành chỗ đỗ xe.

Thiết kế tác tử trong môi trường đa tác tử thường có những khác biệt đáng kể so với môi trường đơn tác tử; ví dụ, giao tiếp thường trở thành một hành vi hợp lý trong môi trường đa tác tử; trong một số môi trường cạnh tranh, hành vi ngẫu nhiên trở thành lựa chọn hợp lý vì nó giúp tránh những rủi ro bị dự đoán trước hành vi.

Xác định và không xác định. Nếu trạng thái tiếp theo của môi trường được xác định hoàn toàn bởi trạng thái hiện tại và hành động thực hiện bởi tác tử (hoặc các tác tử), chúng ta nói rằng môi trường là xác định; ngược lại, nó là không xác định. Theo lý thuyết, tác tử không cần quan tâm đến sự không chắc chắn trong một môi trường quan sát được đầy đủ và xác định. Tuy nhiên, nếu môi trường chỉ quan sát được một phần, thì nó có thể trở thành môi trường không xác định.

Hầu hết các tình huống thực tế đều phức tạp đến mức không thể theo dõi được tất cả các khía cạnh của môi trường; do đó, chúng phải được coi là không xác định. Trong trường hợp lái xe taxi tự hành, rõ ràng là không xác định theo nghĩa này, vì không thể dự đoán chính xác hành vi của các thực thể tham gia giao thông; hơn nữa, lốp xe có thể bất ngờ bị nổ và động cơ có thể hỏng mà không có cảnh báo trước. Môi trường của máy hút bụi, như đã mô tả, là xác định, nhưng có thể tồn tại các yếu tố không xác định như bụi xuất hiện ngẫu nhiên.

Cuối cùng, ta cần lưu ý rằng từ "stochastic" được một số người sử dụng như một từ đồng nghĩa với "không xác định". Tuy nhiên, ta phân biệt hai thuật ngữ này như sau: một mô

hình của môi trường là "stochastic" nếu nó xử lý các xác suất cụ thể (ví dụ: "có 25% khả năng mưa vào ngày mai"), và "không xác định" nếu các khả năng chỉ được liệt kê mà không được định lượng (ví dụ: "có khả năng mưa vào ngày mai").

PHÂN ĐOẠN và TUẦN TỰ: Trong môi trường nhiệm vụ phân đoạn, hành vi của tác tử được chia thành các bước. Tại mỗi bước, tác tử thu nhận thông tin từ môi trường và sau đó thực hiện một hành động duy nhất. Đáng chú ý là hành động ở bước tiếp theo không phụ thuộc vào các hành động được thực hiện tại các bước trước đó. Nhiều bài toán phân lớp là phân đoạn. Ví dụ, một tác tử có nhiệm vụ phát hiện các bộ phận lỗi trên một dây chuyền lắp ráp đưa ra quyết định dựa trên bộ phận hiện tại, không quan tâm đến các quyết định ở các bước trước đó; hơn nữa, quyết định hiện tại không ảnh hưởng đến việc bộ phận tiếp theo có lỗi hay không. Ngược lại, trong môi trường tuần tự, quyết định hiện tại có thể ảnh hưởng đến các quyết định trong tương lai. Cờ vua và lái xe taxi tự hành là những môi trường tuần tự: trong cả hai trường hợp, hành động tại thời điểm hiện tại có thể có ảnh hưởng lâu dài. Môi trường phân đoạn đơn giản hơn rất nhiều so với môi trường tuần tự vì tác tử không cần phải suy nghĩ về tương lai.

TĨNH và ĐỘNG: Nếu môi trường có thể thay đổi trong quá trình tác tử xem xét các lựa chọn hành động, ta gọi môi trường đó là động đối với tác tử; nếu không, nó là tĩnh. Môi trường tĩnh dễ xử lý vì tác tử không cần tiếp tục quan sát thế giới trong quá trình đưa ra quyết định hành động và không cần lo lắng về thời gian trôi qua. Môi trường động, ngược lại, liên tục yêu cầu tác tử đưa ra quyết định hành động; nếu tác tử chưa đưa ra quyết định, ta coi đó là quyết định không làm gì cả. Nếu môi trường không thay đổi theo thời gian nhưng điểm hiệu suất của tác tử thay đổi, ta nói rằng môi trường là bán động. Lái xe taxi tự động là một ví dụ rõ ràng về môi trường động: các xe khác và chính chiếc taxi vẫn di chuyển trong khi thuật toán lái xe đang xem xét về quyết định hành động tiếp theo. Cờ vua, khi chơi với đồng hồ tính giờ, là một ví dụ về môi trường bán động. Trò chơi ô chữ là môi trường tĩnh.

RỜI RẠC và LIÊN TỤC: Sự phân biệt giữa rời rạc và liên tục áp dụng cho trạng thái của môi trường, cách xử lý thời gian, cảm nhận cũng như hành động của tác tử. Ví dụ, môi trường chơi cờ vua có một số hữu hạn trạng thái phân biệt (không tính đồng hồ). Cờ vua cũng có một tập hợp cảm nhận và hành động rời rạc. Lái taxi tự động là một bài toán liên tục về trạng thái và thời gian: tốc độ và vị trí của taxi cùng với các phương tiện khác thay đổi một cách liên tục và mượt mà theo thời gian. Các hành động lái taxi cũng là liên tục (góc đánh lái, v.v.). Đầu vào từ máy ảnh kỹ thuật số, dù đúng ra là rời rạc, thường được xem như biến thiên liên tục.

BIẾT RÕ và CHƯA BIẾT: Phân biệt này không áp dụng cho môi trường mà áp dụng cho trạng thái hiểu biết của tác tử (hoặc người thiết kế) về "quy luật vận động" của môi trường. Trong một môi trường đã biết rõ, kết quả (hoặc xác suất kết quả nếu môi trường là không xác định) cho tất cả các hành động đã được xác định trước. Rõ ràng, nếu môi trường là chưa biết, tác tử sẽ phải học cách môi trường hoạt động để đưa ra những quyết định tốt. Sự phân biệt giữa môi trường đã biết rõ và chưa biết không giống như sự phân biệt giữa môi trường quan sát được hoàn toàn và môi trường quan sát được một phần. Hoàn toàn có thể một môi trường đã biết rõ vẫn chỉ có thể quan sát được một phần - ví dụ, trong trò chơi bài solitaire, ta đã biết các luật chơi nhưng vẫn không thể nhìn thấy các lá bài chưa lật. Ngược lại, một môi trường chưa biết có thể quan sát được toàn bộ - trong một trò chơi video mới,

màn hình có thể hiển thị toàn bộ trạng thái trò chơi nhưng ta vẫn chưa biết chức năng của từng nút bấm cho đến khi thử chúng.

Chỉ số đánh giá hiệu suất có thể không được biết đến do người thiết kế không chắc chắn cách viết nó đúng, hoặc do người dùng cuối – mà sở thích của họ là quan trọng - chưa được xác định. Ví dụ, một tài xế taxi thường không biết khách hàng mới thích một chuyến đi thư giãn hay nhanh chóng, phong cách lái an toàn hay quyết liệt. Một trợ lý cá nhân ảo ban đầu cũng không biết gì về những sở thích cá nhân của chủ sở hữu mới. Trong những trường hợp như vậy, tác tử có thể học thêm về chỉ số hiệu suất dựa trên sự tương tác tiếp theo với người thiết kế hoặc người sử dụng. Điều này cho thấy môi trường nhiệm vụ cần được xem như một môi trường đa tác tử.

Môi trường mà chỉ có thể quan sát một phần, đa tác tử, không xác định, tuần tự, động, liên tục và chưa biết là trường hợp khó nhất. Lái taxi là khó khăn theo các khía cạnh này, trừ việc tài xế đã biết phần lớn về môi trường lái xe của mình. Điều khiển một chiếc xe thuê ở một quốc gia mới với địa lý không quen thuộc, luật giao thông khác và hành khách căng thẳng sẽ mang lại trải nghiệm thú vị.

Hình 7 liệt kê các tính chất của một số môi trường quen thuộc. Lưu ý rằng các tính chất này không phải lúc nào cũng rõ ràng. Chúng không bao gồm cột "biết rõ/không biết" vì, như đã giải thích từ trước, đây không phải là một tính chất của môi trường. Đối với một số môi trường như cờ vua và poker, việc cung cấp cho tác tử kiến thức đầy đủ về các luật chơi khá dễ dàng, nhưng việc để cho một tác tử có thể học chơi những trò chơi này mà không có những kiến thức đó vẫn là một điều thú vị.

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Crossword puzzle	Fully	Single	Deterministic	Sequential	Static	Discrete
Chess with a clock	Fully	Multi	Deterministic	Sequential	Semi	Discrete
Poker	Partially	Multi	Stochastic	Sequential	Static	Discrete
Backgammon	Fully	Multi	Stochastic	Sequential	Static	Discrete
Taxi driving	Partially	Multi	Stochastic	Sequential	Dynamic	Continuous
Medical diagnosis	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Image analysis	Fully	Single	Deterministic	Episodic	Semi	Continuous
Part-picking robot	Partially	Single	Stochastic	Episodic	Dynamic	Continuous
Refinery controller	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
English tutor	Partially	Multi	Stochastic	Sequential	Dynamic	Discrete

Hình 7. Các ví dụ về môi trường nhiệm vụ và các tính chất của chúng

5 Cấu trúc của tác tử

Ta đã nói về các tác tử bằng cách mô tả hành vi - hành động được thực hiện sau mỗi chuỗi cảm nhận. Bây giờ, ta sẽ nói về cách các thành phần bên trong tác tử hoạt động như thế nào. Nhiệm vụ của trí tuệ nhân tạo là thiết kế một chương trình tác tử thực thi hàm tác tử - ánh xạ từ các cảm nhận sang các hành động. Ta giả định rằng chương trình này sẽ chạy trên một thiết bị tính toán nào đó với các cảm biến vật lý và bộ điều khiển - ta gọi thiết bị này là kiến trúc (architecture) tác tử.

agent = architecture + program

Rõ ràng, chương trình mà ta chọn phải phù hợp với kiến trúc. Nếu chương trình đề xuất các hành động như Đi bộ, thì kiến trúc tốt nhất phải có chân. Kiến trúc có thể chỉ là một máy tính cá nhân thông thường, hoặc có thể là một chiếc ô tô robot với nhiều máy tính tích hợp, camera và các cảm biến khác. Nói chung, kiến trúc cho phép các cảm nhận từ các cảm biến được truyền đến chương trình, chạy chương trình và cung cấp các lựa chọn hành động của chương trình cho các bộ điều khiển.

5.1 Chương trình tác tử

Các chương trình tác tử được thiết kế trong cuốn sách này có cấu trúc giống nhau: chúng nhận cảm nhận hiện tại từ các cảm biến và trả về hành động cho các bộ điều khiển. Sự khác biệt giữa chương trình tác tử và hàm tác tử là chương trình tác tử chỉ có cảm nhận hiện tại làm đầu vào, trong khi hàm tác tử có thể phụ thuộc vào toàn bộ chuỗi cảm nhận. Chương trình tác tử chỉ nhận cảm nhận hiện tại làm đầu vào vì không có thông tin nào khác được cung cấp từ môi trường; nếu việc xác định hành động của tác tử cần dựa trên toàn bộ chuỗi cảm nhận, tác tử sẽ phải ghi nhớ các cảm nhận trước đó. Thách thức chính đối với Trí tuệ Nhân tạo là tìm cách viết các chương trình tác tử sao cho chúng tạo ra hành vi hợp lý từ một chương trình nhỏ thay vì từ một bảng lớn.

Trong mục này, ta sẽ trình bày bốn loại chương trình tác tử cơ bản, thể hiện những nguyên tắc cơ bản của hầu hết các hệ thống thông minh:

- Chương trình tác tử phản xạ đơn giản;
- Chương trình tác tử phản xạ dựa trên mô hình;
- Chương trình tác tử dựa trên mục tiêu;
- Chương trình tác tử dựa trên lợi ích.

Mỗi loại chương trình tác tử kết hợp các thành phần theo các cách cụ thể để tạo ra hành động. Tất cả các tác tử này đều có thể chuyển đổi thành các tác tử có khả học, từ đó cải thiện hiệu suất của các thành phần và đưa ra các hành động tốt hơn.

5.2 Tác tử phản xạ đơn giản

Tác tử phản xạ đơn giản là loại tác tử đơn giản nhất. Những tác tử này lựa chọn hành động dựa trên cảm nhận hiện tại, bỏ qua phần còn lại của chuỗi cảm nhận. Ví dụ, tác tử hút bụi có hàm tác tử được trình bày trong Hình 3 là một tác tử phản xạ đơn giản, vì quyết định của nó chỉ dựa trên vị trí hiện tại và vị trí đó có bụi hay không. Một chương trình tác tử cho tác tử này được trình bày trong Hình 4. Mặc dù chương trình tác tử được viết bằng các câu lệnh if-then-else, nó đơn giản đến mức có thể được cài đặt dưới dạng một mạch logic.

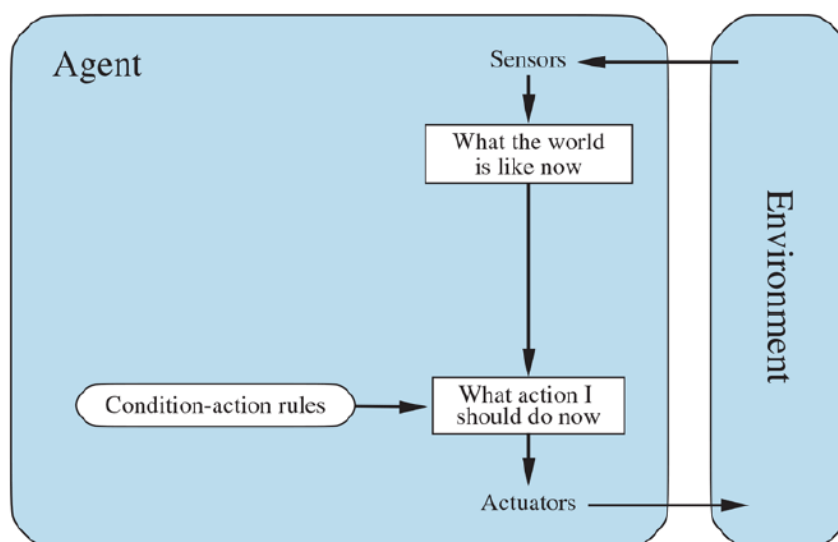
Các hành vi phản xạ đơn giản xảy ra ngay cả trong môi trường phức tạp hơn, ví dụ như trong trường hợp của taxi tự hành. Khi xe phía trước phanh và đèn phanh sáng lên, taxi tự hành sẽ nhận biết điều này và bắt đầu phanh xe. Điều này được thực hiện thông qua việc xử lý đầu vào hình ảnh để xác định điều kiện "xe phía trước đang phanh" xảy ra. Sau đó, điều kiện này sẽ kích hoạt hành động "bắt đầu phanh" thông qua một kết nối đã được thiết lập trong chương trình tác tử. Ta gọi một kết nối như vậy là một luật "điều kiện-hành động" được biểu diễn như sau:

if car-in-front-is-braking then initiate-braking.

Các luật này còn được gọi là luật "tình huống-hành động", luật sản xuất hay luật "nếu-thì".

Con người cũng có nhiều kết nối như vậy, một số là phản xạ thông qua việc học (như khi lái xe) và một số khác là phản xạ bản năng (như nháy mắt khi có vật bay vào). Trong lĩnh vực trí tuệ nhân tạo, chúng ta có nhiều phương pháp khác nhau để học và thực hiện những kết nối như vậy.

Chương trình trong Hình 4 dành riêng cho máy hút bụi trong một môi trường cụ thể. Một cách tiếp cận tổng quát và linh hoạt hơn là xây dựng trước một trình thông dịch đa năng cho các luật "điều kiện-hành động", sau đó tạo ra các tập luật cho các môi trường nhiệm vụ cụ thể. Hình 8 trình bày sơ đồ cấu trúc tổng quát cho tác tử phản xạ đơn giản, cho thấy cách các luật "điều kiện-hành động" cho phép tác tử thiết lập kết nối từ cảm nhận đến hành động.



Hình 8. Sơ đồ cấu trúc tổng quát của một tác tử phản xạ đơn giản

Ở đây, các hình chữ nhật được sử dụng để biểu diễn trạng thái bên trong hiện tại của quá trình ra quyết định của tác tử, và các hình oval biểu diễn thông tin nền được sử dụng trong quá trình này.

Một chương trình tác tử cho cấu trúc trong Hình 8 được hiển thị trong Hình 9.

function SIMPLE-REFLEX-AGENT(*percept*) **returns** an action
persistent: *rules*, a set of condition–action rules

```
state ← INTERPRET-INPUT(percept)
rule ← RULE-MATCH(state, rules)
action ← rule.ACTION
return action
```

Hình 9. Chương trình tác tử phản xạ đơn giản tổng quát

Hàm INTERPRET-INPUT tạo ra một mô tả trừu tượng về trạng thái hiện tại từ cảm

nhận, và hàm RULEMATCH trả về luật đầu tiên trong tập luật khớp với mô tả trạng thái đã cho. Nó hoạt động dựa trên một luật mà điều kiện của nó khớp với trạng thái hiện tại, được xác định bởi cảm nhận.

Chương trình tác tử phản xạ trên rất đơn giản, tuy nhiên chúng có khả năng thông minh hạn chế. Tác tử trong Hình 9 chỉ hoạt động tốt khi có thể đưa ra các quyết định chính xác chỉ dựa trên cảm nhận hiện tại - điều này có nghĩa là môi trường phải là quan sát được hoàn toàn.

Dù chỉ có một chút không quan sát được cũng có thể gây ra rắc rối nghiêm trọng. Ví dụ, luật phanh đưa ra ở trên giả định rằng điều kiện "xe phía trước đang phanh" có thể được xác định từ cảm nhận hiện tại - một khung hình video duy nhất. Điều này chỉ hoạt động tốt nếu xe phía trước có một đèn phanh được lắp ở vị trí dễ nhận ra. Thật không may, các mẫu xe có cấu hình khác nhau về đèn hậu, đèn phanh và đèn tín hiệu rẽ, và không phải lúc nào cũng có thể biết từ một bức hình duy nhất rằng xe đang phanh hay chỉ đơn giản là đèn hậu đang sáng. Một tác tử phản xạ đơn giản điều khiển xe đằng sau xe như vậy sẽ hoặc phanh liên tục một cách không cần thiết.

Ta có thể thấy vấn đề tương tự xảy ra đối với máy hút bụi đơn giản. Giả sử tác tử máy hút bụi bị mất cảm biến vị trí và chỉ còn cảm biến bụi. Một tác tử như vậy chỉ có thể thu được hai cảm nhận: [Bụi] và [Sạch]. Khi gặp cảm nhận [Bụi], tác tử sẽ Hút bụi; nhưng khi gặp cảm nhận [Sạch], tác tử sẽ phải làm gì? Di chuyển sang trái (mãi mãi) sẽ thất bại nếu nó bắt đầu ở ô A, và di chuyển sang phải (mãi mãi) thất bại nếu nó bắt đầu ở ô B. Vòng lặp vô hạn thường không thể tránh được đối với các tác tử phản xạ đơn giản hoạt động trong môi trường không quan sát được hoàn toàn.

Có thể thoát khỏi vòng lặp vô hạn nếu tác tử có thể ngẫu nhiên hóa các hành động của mình. Ví dụ, nếu tác tử máy hút bụi thu được cảm nhận [Sạch], nó có thể tung đồng xu để chọn giữa Di chuyển sang trái và Di chuyển sang phải. Dễ dàng thấy rằng tác tử sẽ đến được ô còn lại trong trung bình hai bước. Sau đó, nếu ô đó có bụi, tác tử sẽ làm sạch và hoàn thành nhiệm vụ. Do đó, một tác tử phản xạ đơn giản có yếu tố ngẫu nhiên có thể vượt trội hơn so với tác tử phản xạ đơn giản không có yếu tố ngẫu nhiên.

Thực tế, trong một số môi trường đa tác tử, hành vi ngẫu nhiên đúng đắn có thể được coi là hợp lý. Tuy nhiên, trong môi trường đơn tác tử, việc ngẫu nhiên hóa thường không được coi là hợp lý. Đây chỉ là một kỹ thuật hữu ích giúp tác tử phản xạ đơn giản vượt qua một số tình huống cụ thể, tuy nhiên, trong hầu hết các trường hợp, có thể đạt được hiệu quả tốt hơn nhiều thông qua việc sử dụng các tác tử xác định phức tạp hơn.

5.3 Tác tử phản xạ dựa trên mô hình

Cách hiệu quả nhất để xử lý tình trạng môi trường không quan sát được hoàn toàn là tác tử phải theo dõi được phần của thế giới mà nó không thể quan sát. Tác tử cần duy trì một trạng thái bên trong dựa trên chuỗi cảm nhận, phản ánh một số khía cạnh chưa được quan sát của trạng thái hiện tại. Đối với trường hợp phanh xe, trạng thái bên trong không quá phức tạp - chỉ là khung hình trước đó từ camera, cho phép tác tử phát hiện khi nào hai đèn đỏ ở cạnh hông của xe bật hoặc tắt đồng thời. Đối với các tác vụ lái xe khác như chuyển làn đường, tác tử cần theo dõi vị trí của những chiếc xe khác nếu nó không thể nhìn thấy tất cả chúng cùng một lúc.

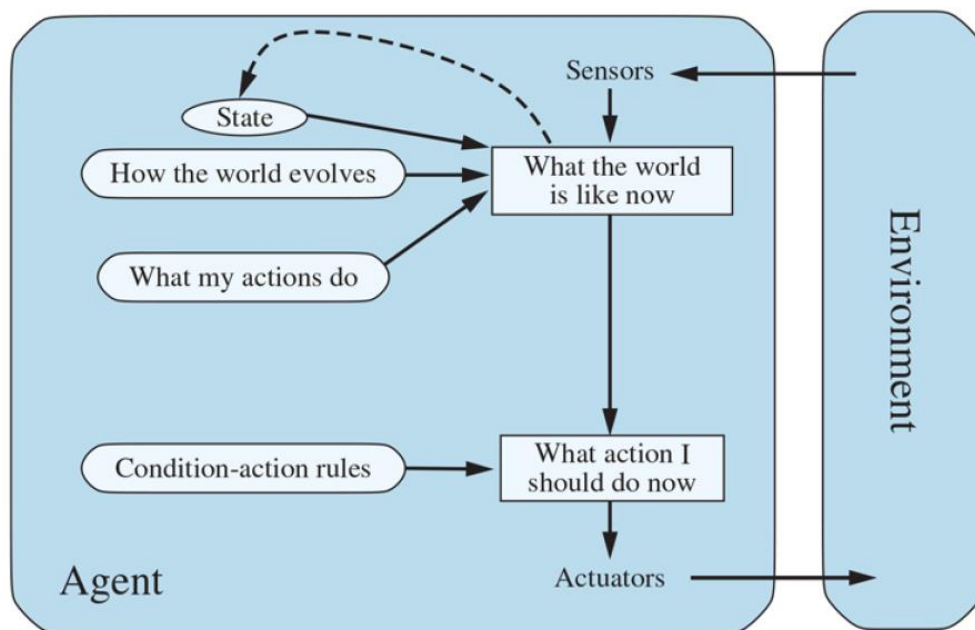
Cập nhật thông tin trạng thái bên trong này theo thời gian đòi hỏi hai loại kiến thức cần

phải được mã hóa trong chương trình tác tử dưới một dạng nào đó. Thứ nhất, ta cần thông tin về cách môi trường thay đổi theo thời gian, có thể chia thành hai phần chính: tác động của hành động của tác tử đối với môi trường và cách môi trường biến đổi độc lập với tác tử. Ví dụ, khi tác tử xoay vô lăng theo chiều kim đồng hồ, ô tô sẽ rẽ sang phải, và khi trời mưa, camera của ô tô có thể bị ướt. Kiến thức về "cách môi trường hoạt động" này - dù được cài đặt bằng cách nào - được gọi là một mô hình chuyển đổi của môi trường.

Thứ hai, ta cần một số thông tin về cách trạng thái của môi trường được phản ánh trong cảm nhận của tác tử. Ví dụ, khi xe phía trước bắt đầu phanh, một hoặc nhiều vùng sáng đỏ xuất hiện trong hình ảnh của camera phía trước, và khi camera bị ướt, các đối tượng hình giọt nước xuất hiện trong hình ảnh che khuất một phần đường đi. Loại kiến thức như vậy được gọi là một mô hình cảm biến.

Mô hình chuyển đổi và mô hình cảm biến là hai yếu tố cần thiết để tác tử có thể theo dõi trạng thái của môi trường - trong phạm vi có thể nhìn thấy theo khả năng của các cảm biến. Tác tử sử dụng cả hai mô hình này được gọi là tác tử dựa trên mô hình.

Hình 10 trình bày cấu trúc của tác tử phản xạ dựa trên mô hình với trạng thái bên trong, cho thấy cách kết hợp cảm nhận hiện tại với trạng thái bên trong cũ để tạo ra mô tả về trạng thái hiện tại, dựa trên các mô hình về cách môi trường hoạt động.



Hình 10. Tác tử phản xạ dựa trên mô hình

Chương trình tác tử được trình bày trong Hình 2.12. Tác tử này theo dõi trạng thái hiện tại của thế giới bằng cách sử dụng một trạng thái bên trong. Hàm UPDATE-STATE có trách nhiệm tạo ra mô tả về trạng thái bên trong mới. Sau đó, tác tử chọn một hành động theo cách tương tự như tác tử phản xạ đơn giản. Cách biểu diễn chi tiết về mô hình và trạng thái thay đổi tùy thuộc vào loại môi trường và công nghệ cụ thể được sử dụng trong thiết kế tác tử.

function MODEL-BASED-REFLEX-AGENT(*percept*) **returns** an action
persistent: *state*, the agent's current conception of the world state
transition_model, a description of how the next state depends on
the current state and action
sensor_model, a description of how the current world state is reflected
in the agent's percepts
rules, a set of condition–action rules
action, the most recent action, initially none

```

state ← UPDATE-STATE(state, action, percept, transition_model, sensor_model)
rule ← RULE-MATCH(state, rules)
action ← rule.ACTION
return action

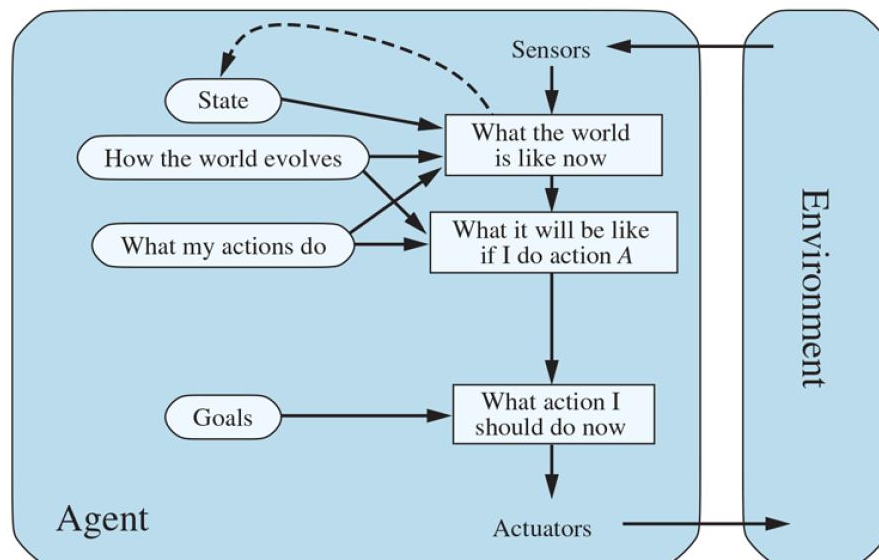
```

Hình 11. Chương trình tác tử phản xạ dựa trên mô hình

Dù dùng bất kỳ cách biểu diễn nào, hiếm khi tác tử có thể xác định chính xác trạng thái hiện tại của một môi trường không quan sát được hoàn toàn. Thay vào đó, ô được ghi là "what the world is like now" (Hình 10) biểu diễn "phỏng đoán tốt nhất" của tác tử (hoặc đôi khi là những phỏng đoán tốt nhất, nếu tác tử xem xét nhiều khả năng). Ví dụ, một xe taxi tự hành có thể không thể nhìn xung quanh chiếc xe tải lớn đang dừng ở phía trước và chỉ có thể phỏng đoán về nguyên nhân gây ra việc này. Do đó, sự không chắc chắn về trạng thái hiện tại có thể không thể tránh khỏi, nhưng tác tử vẫn phải đưa ra quyết định.

5.4 Tác tử dựa trên mục tiêu

Chỉ biết thông tin về trạng thái hiện tại của môi trường không phải lúc nào cũng đủ để quyết định hành động. Ví dụ, tại một ngã tư đường, xe taxi có thể rẽ trái, rẽ phải hoặc đi thẳng. Quyết định đúng phụ thuộc vào địa điểm mà xe taxi muốn đến. Ngoài việc mô tả trạng thái hiện tại, tác tử cần có thông tin về mục tiêu, mô tả các tình huống mong muốn, ví dụ như đến một địa điểm cụ thể. Chương trình tác tử kết hợp thông tin này với mô hình (giống thông tin được sử dụng trong tác tử phản xạ dựa trên mô hình) để chọn hành động nhằm đạt được mục tiêu. Hình 12 mô tả cấu trúc của tác tử dựa trên mục tiêu.



Hình 12. Tác tử dựa trên mô hình và mục tiêu

Tác tử theo dõi trạng thái của môi trường cũng như lưu một tập các mục tiêu mà nó đang cố gắng hoàn thành, và chọn một hành động mà sẽ (cuối cùng) dẫn đến việc đạt được các mục tiêu đó.

Đôi khi việc lựa chọn hành động dựa trên mục tiêu là đơn giản - ví dụ, khi việc đạt được mục tiêu xảy ra ngay sau một hành động duy nhất. Tuy nhiên, đôi khi nó sẽ phức tạp hơn - ví dụ, khi tác tử phải xem xét các chuỗi hành động phức tạp để tìm cách đạt được mục tiêu. Tìm kiếm và lập kế hoạch là những lĩnh vực con của Trí tuệ Nhân tạo tập trung vào việc tìm ra chuỗi hành động để đạt được mục tiêu của tác tử.

Chú ý rằng việc đưa ra quyết định ở đây hoàn toàn khác với các luật “điều kiện-hành động” đã được đề cập ở trên, bởi vì nó liên quan đến việc xem xét tương lai - cả “What will happen if I do such-and-such?” (“Những gì sẽ xảy ra nếu tôi làm cái này?”) và “Will that make me happy?” (“Điều đó sẽ làm tôi hạnh phúc chăng?”). Trong thiết kế của tác tử phản xạ, thông tin này không được biểu diễn một cách tường minh, vì các luật tích hợp sẵn ánh xạ trực tiếp từ cảm nhận đến hành động. Tác tử phản xạ phanh khi thấy đèn phanh, đơn giản như vậy. Nó không biết tại sao. Một tác tử dựa trên mục tiêu phanh khi nhìn thấy đèn phanh vì đó là hành động duy nhất mà nó dự đoán sẽ đạt được mục tiêu là không va chạm với các xe khác.

Mặc dù tác tử dựa trên mục tiêu có vẻ không hiệu quả hơn, nhưng nó linh hoạt hơn vì kiến thức hỗ trợ ra quyết định được biểu diễn một cách tường minh và có thể được thay đổi. Ví dụ, hành vi của tác tử dựa trên mục tiêu có thể dễ dàng thay đổi để đi đến một đích đến khác, chỉ cần mô tả đích đến đó là mục tiêu. Các luật tích hợp của tác tử phản xạ về việc khi nào rẽ và khi nào tiếp tục đi thẳng chỉ hoạt động cho một đích đến duy nhất; chúng phải được thay thế toàn bộ để đi đến một địa điểm mới.

5.5 Tác tử dựa trên tính hữu ích

Sử dụng chỉ mục tiêu mà không đủ để tạo ra hành vi chất lượng cao trong hầu hết các môi trường. Ví dụ, có nhiều chuỗi hành động có thể đưa taxi đến điểm đến của nó (đạt được mục tiêu), nhưng có một số chuỗi hành động nhanh hơn, an toàn hơn, đáng tin cậy hơn hoặc tiết kiệm hơn so với những chuỗi khác. Mục tiêu chỉ tạo ra sự phân biệt đơn giản giữa trạng thái "hạnh phúc" (đạt được mục tiêu) và "không hạnh phúc" (không đạt được mục tiêu). Để đánh giá chính xác các trạng thái khác nhau của môi trường dựa trên mức độ hạnh phúc mà chúng mang lại cho tác tử, cần có một chỉ số hiệu suất tổng quát hơn. Vì thuật ngữ "hạnh phúc" nghe không có vẻ khoa học lắm, nên các nhà kinh tế và khoa học máy tính thường sử dụng thuật ngữ "tính hữu ích" (utility) để thay thế.

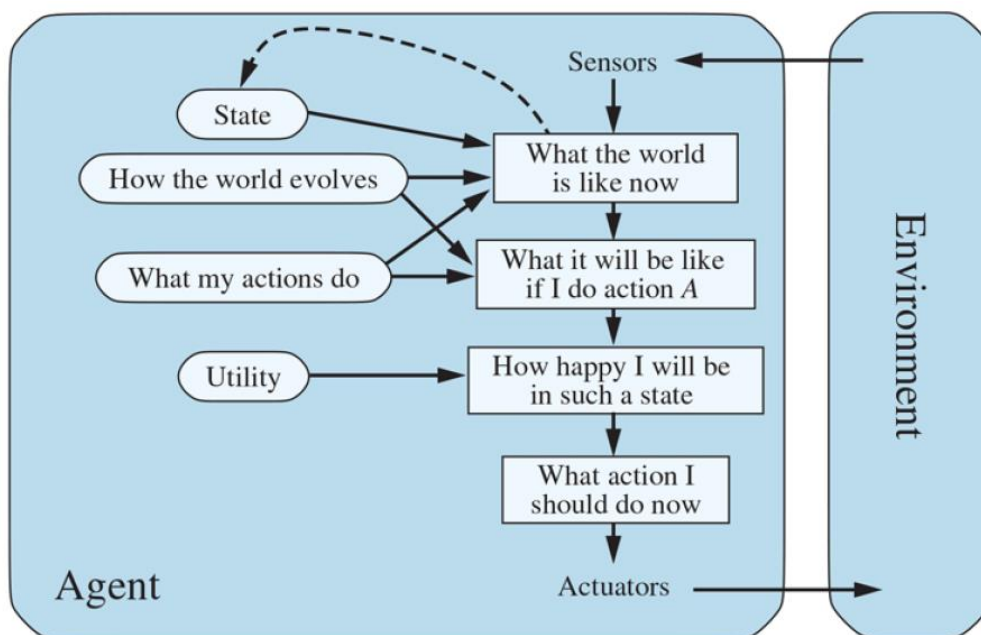
Chỉ số hiệu suất gán điểm cho từng chuỗi trạng thái môi trường, do đó nó có thể phân biệt dễ dàng giữa các con đường khác nhau để đến đích của xe taxi, từ cách tốt hơn đến cách kém mong muốn hơn. Hàm hữu ích (utility function) của tác tử là một biểu diễn của chỉ số hiệu suất. Tác tử lựa chọn hành động để tối đa hóa tính hữu ích sẽ được coi là hợp lý theo chỉ số hiệu suất.

Chú ý rằng đây không phải là cách duy nhất để có tính hợp lý - ta đã thấy một chương trình tác tử hợp lý cho máy hút bụi (Hình 4) mà không biết hàm hữu ích của nó là gì - nhưng tương tự như các tác tử dựa trên mục tiêu, tác tử dựa trên tính hữu ích có nhiều lợi thế về tính linh hoạt và khả năng học hỏi. Hơn nữa, trong hai trường hợp sau đây, mục tiêu là không đủ,

nhưng một tác tử dựa trên tính hữu ích vẫn có thể đưa ra những quyết định hợp lý. Đầu tiên, khi có những mục tiêu xung đột nhau, chỉ một số trong chúng là có thể được đạt (ví dụ, giữa tốc độ và an toàn), hàm hữu ích xác định sự cân đối thích hợp. Thứ hai, khi có nhiều mục tiêu mà tác tử có thể hướng đến và không có mục tiêu nào đảm bảo được đạt được một cách chắc chắn, hàm hữu ích cung cấp cách để đánh giá khả năng thành công dựa trên mức độ quan trọng của từng mục tiêu.

Quan sát được không đầy đủ và tính không xác định thường xảy ra trong thực tế, vì vậy, ra quyết định trong các tình huống không chắc chắn cũng là điều tất yếu. Theo quan điểm kỹ thuật, một tác tử hợp lý dựa trên tính hữu ích sẽ lựa chọn hành động để tối đa hóa giá trị hữu ích kỳ vọng của kết quả hành động - tức là giá trị hữu ích trung bình mà tác tử kỳ vọng nhận được, dựa trên xác suất và giá trị hữu ích của từng kết quả. Tác tử sở hữu một hàm hữu ích rõ ràng có thể đưa ra quyết định hợp lý bằng một thuật toán tổng quát mà không phụ thuộc vào hàm hữu ích cụ thể đang được tối đa hóa.

Cấu trúc của tác tử dựa trên tính hữu ích được trình bày trong Hình 13. Nó sử dụng một mô hình của môi trường cùng với một hàm hữu ích đo lường sự ưa thích giữa các trạng thái của môi trường. Sau đó, nó lựa chọn hành động dẫn đến giá trị hữu ích kỳ vọng tốt nhất, trong đó giá trị hữu ích kỳ vọng được tính toán bằng cách lấy trung bình trên tất cả các trạng thái kết quả có thể với trọng số là xác suất của mỗi kết quả.



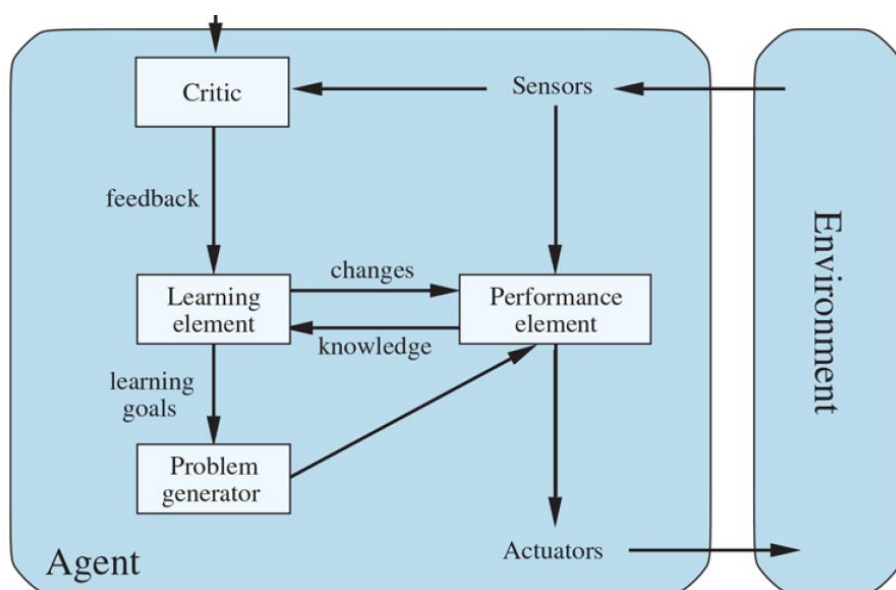
Hình 13. Tác tử dựa trên mô hình và tính hữu ích

5.6 Tác tử có khả năng học

Ta đã mô tả các chương trình tác tử với các phương pháp lựa chọn hành động khác nhau. Cho đến nay, ta chưa giải thích cách xây dựng các chương trình tác tử này. Trong bài báo nổi tiếng của mình, Turing (1950) đã đề xuất phương pháp xây dựng các hệ thống có khả năng học và sau đó dạy chúng. Trong nhiều lĩnh vực của Trí tuệ Nhân tạo, đây hiện là phương pháp được ưa thích để tạo ra các hệ thống hàng đầu. Bất kỳ loại tác tử nào (dựa trên mô hình, dựa trên mục tiêu, dựa trên tính hữu ích, v.v.) đều có thể được xây dựng dưới dạng một tác tử có khả năng học.

Học tập có một lợi điểm khác, như chúng tôi đã đề cập trước đó: nó cho phép tác tử hoạt động trong các môi trường ban đầu không biết và trở nên thành thạo hơn so với kiến thức ban đầu mà nó có thể có. Trong phần này, chúng tôi tóm lược những ý tưởng chính của các tác tử học.

Một tác tử học có thể được chia thành bốn thành phần khái niệm, như được hiển thị trong Hình 14. Phân biệt quan trọng nhất là giữa thành phần học tập, có trách nhiệm cải thiện, và thành phần hiệu suất, có trách nhiệm lựa chọn hành động bên ngoài. Thành phần hiệu suất là những gì chúng ta đã xem xét trước đây là toàn bộ tác tử: nó nhận thông tin từ quan sát và quyết định hành động. Thành phần học tập sử dụng phản hồi từ nhà phê bình về cách tác tử hoạt động và xác định cách thay đổi thành phần hiệu suất để hoạt động tốt hơn trong tương lai. Hộp "thành phần hiệu suất" đại diện cho những gì chúng ta trước đây đã xem là toàn bộ chương trình tác tử. Bây giờ, hộp "thành phần học tập" được phép thay đổi chương trình đó để cải thiện hiệu suất của nó.



Hình 14. Tác tử học tổng quát

Thiết kế thành phần học tập phụ thuộc rất nhiều vào thiết kế thành phần hiệu suất. Khi cố gắng thiết kế một tác tử có khả năng học một khả năng nhất định, câu hỏi đầu tiên không phải là "Làm thế nào để tôi giúp nó học điều này?" mà là "Loại thành phần hiệu suất nào mà tác tử của tôi sẽ sử dụng để thực hiện điều này sau khi nó đã học được?" Với một thiết kế cho thành phần hiệu suất, các cơ chế học tập có thể được xây dựng để cải thiện mọi phần của tác tử.

Người đánh giá cho thành phần học tập biết được tốt đến mức nào tác tử đang hoạt động dựa trên một tiêu chuẩn hiệu suất cố định. Người đánh giá là cần thiết vì các thông tin cảm quan không cung cấp bất kỳ chỉ dẫn nào về sự thành công của tác tử. Ví dụ, một chương trình cờ vua có thể nhận được thông tin cảm quan cho biết nó đã chiếu vua đối thủ, nhưng nó cần một tiêu chuẩn hiệu suất để biết rằng đây là một điều tốt; thông tin cảm quan chưa nói lên điều đó. Quan trọng là tiêu chuẩn hiệu suất phải được cố định. Về mặt khái niệm, người ta nên coi đó như là bên ngoài tác tử hoàn toàn, vì tác tử không được thay đổi để phù hợp với hành vi của nó.

Thành phần cuối cùng của tác tử học tập là bộ tạo vấn đề. Nó có trách nhiệm đề xuất các hành động sẽ dẫn đến những trải nghiệm mới và có giá trị thông tin. Nếu chỉ theo ý của thành phần hiệu suất, nó sẽ tiếp tục thực hiện các hành động tốt nhất dựa trên những gì nó biết, nhưng nếu tác tử sẵn lòng khám phá một chút và thực hiện một số hành động có thể không tối ưu trong ngắn hạn, nó có thể khám phá ra những hành động tốt hơn cho dài hạn. Nhiệm vụ của bộ tạo vấn đề là đề xuất những hành động khám phá này. Điều này giống như các nhà khoa học khi tiến hành thực nghiệm. Galileo không nghĩ rằng việc thả các viên đá từ đỉnh một tháp ở Pisa có giá trị đối với nó. Ông không cố gắng phá vỡ những viên đá hay thay đổi não bộ của những người đi bộ bất hạnh. Mục tiêu của ông là thay đổi bản thân mình bằng cách xác định một lý thuyết tốt hơn về chuyển động của các vật thể.

Thành phần học tập có thể thay đổi bất kỳ thành phần "kiến thức" nào được hiển thị trong sơ đồ tác tử (Hình trên). Trường hợp đơn giản nhất là học trực tiếp từ chuỗi nhận thức. Quan sát cặp trạng thái liên tiếp của môi trường có thể cho phép tác tử học "Hành động của tôi làm gì" và "Thế giới thay đổi như thế nào" đáp ứng hành động của nó. Ví dụ, nếu xe taxi tự động áp dụng một áp suất phanh nhất định khi lái xe trên đường ướt, sau đó nó sẽ sớm tìm hiểu được mức giảm tốc thực sự đạt được và liệu nó có trượt khỏi đường hay không. Bộ tạo vấn đề có thể xác định các phần của mô hình cần được cải thiện và đề xuất các thử nghiệm, như thử phanh trên các bề mặt đường khác nhau trong các điều kiện khác nhau.

Việc cải thiện các thành phần mô hình của một tác tử dựa trên mô hình để chúng phù hợp hơn với thực tế là ý tưởng tốt gần như luôn luôn, bất kể tiêu chuẩn hiệu suất bên ngoài. (Trong một số trường hợp, từ quan điểm tính toán, có thể tốt hơn để có một mô hình đơn giản nhưng hơi không chính xác hơn là một mô hình hoàn hảo nhưng phức tạp đáng kinh tởm.) Thông tin từ tiêu chuẩn bên ngoài là cần thiết khi cố gắng học một thành phần phản xạ hoặc hàm tiện ích.

Ví dụ, giả sử tác tử lái taxi không nhận được tiền boa từ những hành khách đã bị lắc lư trong suốt chuyến đi. Tiêu chuẩn hiệu suất bên ngoài phải thông báo cho tác tử rằng việc mất tiền boa là một đóng góp tiêu cực đối với hiệu suất tổng thể của nó; sau đó, tác tử có thể học được rằng những động tác mãnh liệt không đóng góp vào tiện ích của chính nó. Theo một khía cạnh, tiêu chuẩn hiệu suất phân biệt một phần của nhận thức đang vào làm phần thưởng (hoặc phạt) cung cấp phản hồi trực tiếp về chất lượng hành vi của tác tử. Các tiêu chuẩn hiệu suất được cài đặt sẵn như đau đớn và đói trong động vật có thể được hiểu theo cách này.

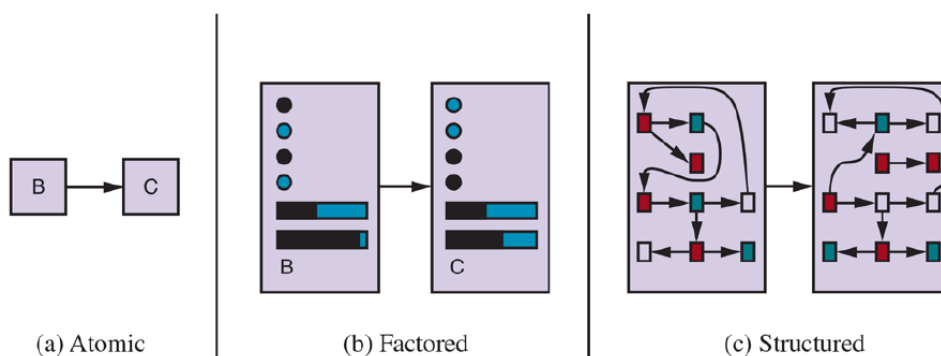
Nói chung hơn, những lựa chọn của con người có thể cung cấp thông tin về sở thích của con người. Ví dụ, giả sử tài xế taxi không biết rằng mọi người thường không thích tiếng ồn to, và quyết định thổi còi liên tục như một cách để đảm bảo người đi bộ biết rằng xe đang đến. Hành vi của con người kết quả - che tai, sử dụng ngôn ngữ tục tĩu và có thể cất dây còi - sẽ cung cấp chứng cứ cho tác tử để cập nhật hàm tiện ích của nó.

Tóm lại, các tác tử có nhiều thành phần khác nhau, và những thành phần đó có thể được biểu diễn theo nhiều cách khác nhau trong chương trình tác tử, do đó có sự đa dạng lớn trong các phương pháp học. Tuy nhiên, có một chủ đề thống nhất duy nhất. Học trong các tác tử thông minh có thể được tóm tắt là quá trình sửa đổi từng thành phần của tác tử để đưa các thành phần này vào sự tương thích gần hơn với thông tin phản hồi có sẵn, từ đó cải thiện hiệu suất tổng thể của tác tử.

5.7 Cách thức hoạt động của các thành phần trong chương trình tác tử

Chúng tôi đã mô tả các chương trình tác tử (ở mức độ rất tổng quát) bao gồm các thành phần khác nhau, có chức năng trả lời những câu hỏi như: "Thế giới hiện tại như thế nào?" "Hành động nào tôi nên thực hiện ngay bây giờ?" "Hành động của tôi có tác dụng gì?" Câu hỏi tiếp theo đối với sinh viên học trí tuệ nhân tạo là "Làm thế nào các thành phần này hoạt động?" Để trả lời đúng câu hỏi đó, cần khoảng một ngàn trang sách, nhưng ở đây chúng tôi muốn nhấn mạnh sự khác biệt cơ bản giữa các cách mà các thành phần có thể đại diện cho môi trường mà tác tử đang sống.

Nói chung, chúng ta có thể đặt các biểu diễn theo một trục tăng dần độ phức tạp và sức mạnh biểu hiện - nguyên tử, rời rạc và cấu trúc. Để minh họa ý tưởng này, nó có ích khi xem xét một thành phần cụ thể của tác tử, chẳng hạn như thành phần xử lý "Hành động của tôi có tác dụng gì". Thành phần này mô tả các thay đổi có thể xảy ra trong môi trường khi thực hiện một hành động, và Hình 15 cung cấp biểu đồ tương đối về cách các chuyển đổi đó có thể được biểu diễn. (a) Biểu diễn nguyên tử: một trạng thái (như B hoặc C) là một hộp đen không có cấu trúc nội tại; (b) Biểu diễn phân tán: một trạng thái bao gồm một vector các giá trị thuộc tính; giá trị có thể là Boolean, có giá trị thực hoặc thuộc một tập hợp cố định các ký hiệu. (c) Biểu diễn có cấu trúc: một trạng thái bao gồm các đối tượng, mỗi đối tượng có thể có các thuộc tính riêng cũng như mối quan hệ với các đối tượng khác.



Hình 15. Ba cách biểu diễn trạng thái

Trong biểu diễn nguyên tử, mỗi trạng thái của thế giới không thể chia thành các phần tử riêng biệt - nó không có cấu trúc nội tại. Hãy xem xét nhiệm vụ tìm tuyến đường lái xe từ một đầu của một quốc gia đến đầu kia thông qua một chuỗi các thành phố. Với mục đích giải quyết vấn đề này, có thể đủ để giảm trạng thái của thế giới chỉ xuống thành tên của thành phố chúng ta đang ở - một nguyên tử kiến thức đơn lẻ, một "hộp đen" chỉ có thuộc tính duy nhất là giống hoặc khác với một hộp đen khác.

Trong biểu diễn phân tán, mỗi trạng thái được chia thành một tập hợp cố định các biến hoặc thuộc tính, mỗi biến có thể có một giá trị. Hãy xem xét một mô tả chi tiết hơn cho cùng vấn đề lái xe, nơi chúng ta cần quan tâm đến nhiều hơn chỉ vị trí nguyên tử ở một thành phố nào đó; chúng ta có thể cần quan tâm đến lượng xăng còn trong bình, tọa độ GPS hiện tại, việc đèn cảnh báo dầu có hoạt động hay không, số tiền chúng ta có để trả phí, đài phát thanh đang nghe là gì và nhiều hơn nữa. Trong khi hai trạng thái nguyên tử khác nhau không có gì chung nhau - chúng chỉ là những hộp đen khác nhau - hai trạng thái phân tán khác nhau có thể chia sẻ một số thuộc tính (như đang ở một vị trí GPS cụ thể) và không chia sẻ các thuộc

tính khác (như có nhiều xăng hoặc không có xăng); điều này làm cho việc biến đổi từ một trạng thái thành trạng thái khác dễ dàng hơn.

Đối với nhiều mục đích, chúng ta cần hiểu rằng thế giới có những vật thể trong đó có mối quan hệ với nhau, không chỉ là biến số với giá trị. Ví dụ, chúng ta có thể nhận thấy rằng một chiếc xe tải lớn phía trước đang lùi vào lối vào trang trại chăn nuôi, nhưng có một con bò lạc đường đang cản trở đường đi của chiếc xe. Một biểu diễn phân tán không có khả năng trang bị sẵn thuộc tính

"TruckAheadBackingIntoDairyFarmDrivewayBlockedByLooseCow"

với giá trị true hoặc false. Thay vào đó, chúng ta cần một biểu diễn có cấu trúc, trong đó các đối tượng như bò và xe tải và các mối quan hệ khác nhau và thay đổi của chúng có thể được mô tả một cách rõ ràng (xem Hình 15(c)).

Như đã đề cập trước đó, trực mà các biểu diễn nguyên tử, phân tán và có cấu trúc nằm trên đó là trực của tính biểu hiện gia tăng. Nói một cách đại khái, một biểu diễn biểu hiện mạnh mẽ hơn có thể bắt được, ít nhất là một cách ngắn gọn, mọi thứ mà một biểu diễn yếu hơn có thể bắt được, cộng thêm một số thứ khác. Thường thì ngôn ngữ biểu diễn mạnh mẽ hơn có thể ngắn gọn hơn rất nhiều; ví dụ, các luật chơi cờ có thể được viết trong một hoặc hai trang của một ngôn ngữ biểu diễn có cấu trúc như logic bậc một (first-order logic) nhưng cần hàng ngàn trang khi viết bằng một ngôn ngữ biểu diễn phân tán như logic xác suất và khoảng vài trang khi viết bằng một ngôn ngữ nguyên tử như của tự động hóa hữu hạn trạng thái. Tuy nhiên, quá trình suy luận và học tập trở nên phức tạp hơn khi sức mạnh biểu diễn gia tăng. Để đạt được lợi ích từ các biểu diễn mạnh mẽ mà tránh các hạn chế của chúng, các hệ thống thông minh trong thế giới thực có thể cần hoạt động ở tất cả các điểm trên trục đồng thời.

Một trực biểu diễn khác liên quan đến ánh xạ của khái niệm vào các vị trí trong bộ nhớ vật lý, có thể là trong máy tính hoặc trong não. Nếu có một ánh xạ một-một giữa các khái niệm và các vị trí bộ nhớ, chúng ta gọi đó là một biểu diễn localist. Ngược lại, nếu biểu diễn của một khái niệm được phân tán trên nhiều vị trí bộ nhớ, và mỗi vị trí bộ nhớ được sử dụng như một phần của biểu diễn của nhiều khái niệm khác nhau, chúng ta gọi đó là một biểu diễn phân tán. Biểu diễn phân tán có tính bền vững hơn đối với nhiễu và mất thông tin. Với biểu diễn localist, ánh xạ từ khái niệm đến vị trí bộ nhớ là tùy ý, và nếu có lỗi truyền thông làm mất một số bit, chúng ta có thể nhầm lẫn giữa khái niệm Truck với khái niệm không liên quan Truce. Nhưng với biểu diễn phân tán, bạn có thể tưởng tượng rằng mỗi khái niệm đại diện cho một điểm trong không gian đa chiều, và nếu một số bit bị lỗi, bạn sẽ di chuyển đến một điểm gần đó trong không gian đó, có ý nghĩa tương tự.

6 Kết luận

Báo cáo này đã đưa chúng ta đi qua một cuộc hành trình vòng quay về Trí tuệ nhân tạo, mà chúng ta coi đó là khoa học về thiết kế các hệ thống tác tử. Các điểm chính cần nhớ là như sau:

Một tác tử là một thực thể có khả năng nhận thức và thực hiện hành động trong một môi trường. Hàm tác tử đối với một tác tử xác định hành động được thực hiện bởi tác tử dựa trên chuỗi cảm nhận nào đó.

Đo lường hiệu suất đánh giá hành vi của tác tử trong một môi trường. Một tác tử hợp

lý hành động nhằm tối đa hóa giá trị kỳ vọng của đo lường hiệu suất, dựa trên chuỗi cảm nhận mà nó đã thấy cho đến thời điểm hiện tại.

Một môi trường nhiệm vụ bao gồm đo lường hiệu suất, môi trường bên ngoài, bộ điều khiển và các cảm biến. Trong việc thiết kế một tác tử, bước đầu tiên luôn luôn là xác định môi trường nhiệm vụ càng cụ thể càng tốt.

Môi trường nhiệm vụ có sự biến đổi theo một số chiều quan trọng. Nó có thể là toàn diện hoặc một phần có thể quan sát được, đơn đặt hàng hoặc nhiều tác tử, xác định hoặc không xác định, từng đoạn hoặc tuần tự, tĩnh hoặc động, rời rạc hoặc liên tục, và đã biết hoặc không biết.

Trong những trường hợp mà đo lường hiệu suất không được biết đến hoặc khó để xác định chính xác, có một rủi ro đáng kể rằng tác tử sẽ tối ưu hóa mục tiêu sai. Trong những trường hợp như vậy, thiết kế tác tử phải phản ánh sự không chắc chắn về mục tiêu thực sự.

Chương trình tác tử thực thi chức năng của tác tử. Có một loạt các thiết kế chương trình tác tử cơ bản phản ánh loại thông tin được làm rõ và được sử dụng trong quyết định. Các thiết kế này khác nhau về hiệu suất, tính gọn và tính linh hoạt. Thiết kế chương trình tác tử phù hợp phụ thuộc vào bản chất của môi trường.

Các tác tử phản xạ đơn giản phản ứng trực tiếp với cảm nhận, trong khi các tác tử phản xạ dựa trên mô hình duy trì trạng thái nội tại để theo dõi các khía cạnh của thế giới không rõ ràng trong cảm nhận hiện tại. Các tác tử dựa trên mục tiêu hành động để đạt được mục tiêu của họ, và các tác tử dựa trên tiện ích cố gắng tối đa hóa "hạnh phúc" kỳ vọng của chính họ.

Tất cả các tác tử đều có thể cải thiện hiệu suất của mình thông qua việc học.

Tài liệu tham khảo

1. Russell and Norvig. Artificial Intelligence: A Modern Approach. Prentice Hall, 4th Edition. 2021
2. Từ Minh Phương. Giáo trình Nhập môn trí tuệ nhân tạo. NXB Thông tin và Truyền thông. 2016