هوش مصنوعی پیشرفته

درس ۲

جستجو با مشاهدات جزئی

Searching with Partial Observations

کاظم فولادی

دانشکده مهندسی برق و کامپیوتر

دانشگاه تهران

http://courses.fouladi.ir/aai

جستجو با مشاهدات جزئی

SEARCHING WITH PARTIAL OBSERVATIONS

در مسائلی با مشاهده‌پذیری جزئی، ادراکات عامل برای تعیین حالت دقیق ناکافی است.
یک کنش می‌تواند منجر به یکی از چندین حالت ممکن شود، حتی اگر محیط قطعی باشد.

مفهوم کلیدی = **حالت باور (belief state)**
بازنمایی باور فعلی عامل در مورد حالت‌های فیزیکی که عامل ممکن است در آنها باشد
[با داشتن دنباله‌ی کنش‌ها و ادراکات]

عامل بدون حسگر	عامل با حسگر	
بدون مشاهده	با مشاهده‌ی جزئی	با مشاهده‌ی کامل

جستجو بدون مشاهده

SEARCHING WITH NO OBSERVATIONS

عامل‌های بدون حسگر نیز می‌توانند اغلب به طور موفق کنش کنند
و دارای **برتری** هستند
(به دلیل عدم تکیه بر اطلاعات حسگری بالقوه غیرقابل اتکا و پرهزینه)

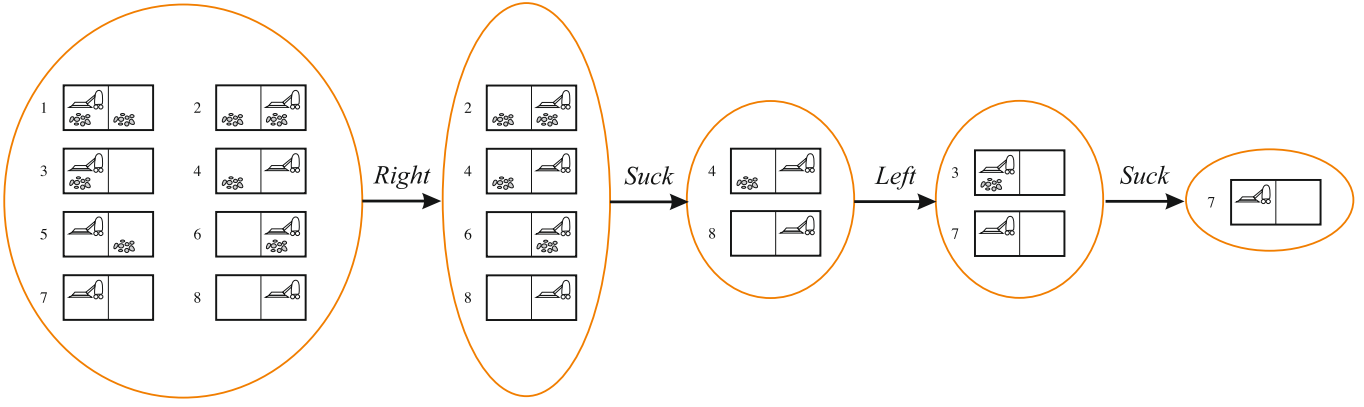
جستجو بدون مشاهده

مثال: دنیای جاروبرقی بدون حسگر

SENSORLESS VACUUM WORLD

عامل **جغرافیای خودش** را می‌شناسد، اما **موقعیت** خود یا توزیع **آلودگی** را نمی‌داند.

حالت آغازین می‌تواند هر یک از حالت‌های هشت‌گانه باشد



با استفاده از دنباله کنش‌های [*Right*, *Suck*, *Left*, *Suck*] تضمین می‌شود که به حالت هدف 7 برسیم صرف‌نظر از اینکه حالت آغازین چه باشد!

اصطلاحاً می‌گوییم عامل می‌تواند دنیا را به حالت 7 **ناگزیر** کند (**coerce**).

جستجو بدون مشاهده

جستجو در فضای باور

SEARCHING IN BELIEF STATE

برای حل مسائل بدون حسگر در فضای حالت‌های باور جستجو می‌کنیم
(به‌جای جستجو در فضای حالت‌های فیزیکی)

فضای باور مسئله **مشاهده‌پذیر کامل** است
(زیرا عامل همیشه حالت باور خود را به طور کامل می‌داند)

راه‌حل در صورت وجود، همیشه دنباله‌ای از کنش‌هاست
(زیرا ادراکات همیشه تهی هستند ⇐ کاملاً قابل پیش‌بینی هستند ⇐ هیچ اقتضائاتی برای طرح / plan وجود ندارد)

* این چهارچوب برای محیطهای غیرقطعی هم درست است.

فرمول‌بندی مسئله

مؤلفه‌های پنج‌گانه‌ی تعریف مسئله‌ی جستجو در فضای حالت‌های باور

مؤلفه‌های پنج‌گانه‌ی تعریف مسئله

حالت‌های باور
Belief states

کل فضای حالت باور: شامل همه‌ی مجموعه‌ی ممکن از حالت‌های فیزیکی
اگر مسئله‌ی فیزیکی P دارای N حالت باشد، مسئله‌ی بدون حسگر حداکثر 2^N حالت دارد.

حالت آغازین
Initial state

حالتی که عامل از آن شروع می‌کند.
(معمولاً مجموعه‌ی همه‌ی حالت‌ها در P: گاهی عامل دانایی بیشتری دارد)

کنش‌های ممکن
Available actions

کنش‌های ممکن عامل در هر حالت s: $\text{ACTIONS}(s)$

مدل گذر
Transition model

در هر حالت s، هر کنش a چه می‌کند؟: $\text{RESULT}(s,a)$
(مجموعه‌ی برآمدهای ممکن به‌جای یک برآمد واحد)

آزمون هدف
Goal test

تعیین کننده‌ی اینکه حالت جاری هدف است یا خیر:
صریح (explicit): بیان مجموعه حالات هدف ضمنی (implicit): بیان ویژگی هدف

تابع هزینه‌ی مسیر
Path cost function

تابعی که به هر مسیر یک هزینه‌ی عددی نسبت می‌دهد.
هزینه‌ی گام (step cost): هزینه‌ی گذر از یک حالت به حالت دیگر با یک کنش $c(s,\ a,\ s')$

فضای حالت مسئله
problem state-space

مجموعه‌ی حالت‌های دسترس‌پذیر از حالت آغازین با حداقل یک دنباله از کنش‌ها

راه‌حل: یک دنباله از کنش‌ها

فرمول‌بندی مسئله

مؤلفه‌های پنج‌گانه‌ی تعریف مسئله‌ی جستجو در فضای حالت‌های باور: کنش‌ها

فرض کنید عامل در فضای باور $b = \{s_1, s_2\}$ قرار دارد، اما

$$\text{ACTIONS}_P(s_1) \neq \text{ACTIONS}_P(s_2)$$

در این صورت عامل مطمئن نیست که کدام کنش‌ها مجاز است.

اگر کنش‌های غیرمجاز تأثیری بر محیط نداشته باشد از **اجتماع** استفاده می‌کنیم

$$\text{ACTIONS}(b) = \bigcup_{s \in b} \text{ACTIONS}_P(s)$$

اگر یک کنش در یک حالت صدمه‌زننده باشد، از **اشتراک** استفاده می‌کنیم

$$\text{ACTIONS}(b) = \bigcap_{s \in b} \text{ACTIONS}_P(s)$$

فرمول‌بندی مسئله

مؤلفه‌های پنج‌گانه‌ی تعریف مسئله‌ی جستجو در فضای حالت‌های باور: مدل گذر

نتیجه‌ی یک کنش، مجموعه‌ی همه‌ی حالت‌های فیزیکی حاصل از انجام آن کنش بر روی همه‌ی حالت‌های فیزیکی در حالت باور فعلی است.

برای کنش‌های **قطعی**، این مجموعه عبارت است از:

$$b' = \text{RESULT}(b,a) = \{s' : s' = \text{RESULT}_P(s,a) \text{ and } s \in b\}$$

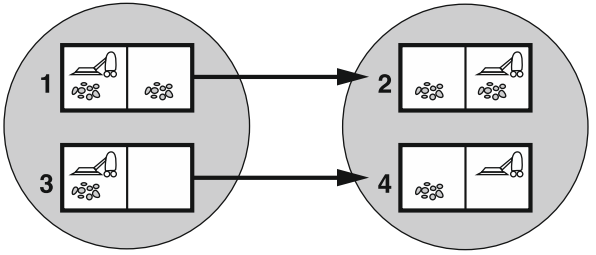
برای کنش‌های **غیرقطعی**، این مجموعه عبارت است از:

$$\begin{aligned} b' = \text{RESULT}(b,a) &= \{s' : s' \in \text{RESULTS}_P(s,a) \text{ and } s \in b\} \\ &= \bigcup_{s \in b} \text{RESULTS}_P(s,a) \ , \end{aligned}$$

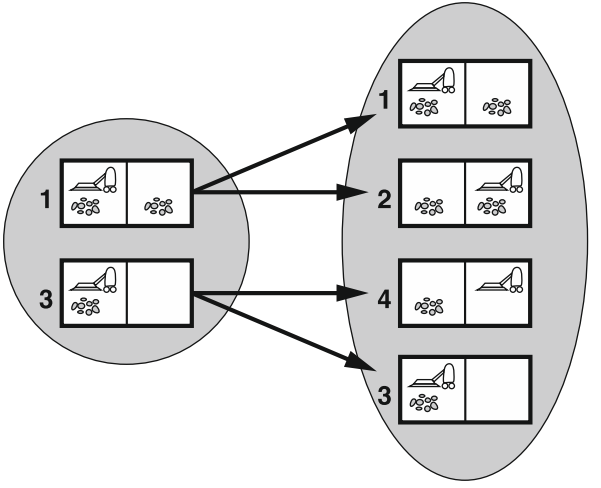
فرآیند تولید یک حالت باور جدید پس از یک کنش، گام پیش‌بینی $\text{PREDICT}_P(b,a)$ نام دارد.

فرمول‌بندی مسئله

مؤلفه‌های پنج‌گانه‌ی تعریف مسئله‌ی جستجو در فضای حالت‌های باور: مدل گذر: مثال



پیش‌بینی حالت بعدی
برای دنیای جاروبرقی بدون حسگر
با کنش **قطعی** *Right*



پیش‌بینی حالت بعدی
برای دنیای جاروبرقی لغزنده
با کنش **غیرقطعی** *Right*

فرمول‌بندی مسئله

مؤلفه‌های پنج‌گانه‌ی تعریف مسئله‌ی جستجو در فضای حالت‌های باور: آزمون هدف

حالت‌های هدف در فضای باور تنها حالت‌هایی هستند که در آن
همه‌ی حالت‌های فیزیکی GOAL-TEST_P را ارضا می‌کنند.

یک عامل ممکن است تصادفاً زودتر به هدف برسد، اما این را نخواهد دانست.

فرمول‌بندی مسئله

مؤلفه‌های پنج‌گانه‌ی تعریف مسئله‌ی جستجو در فضای حالت‌های باور: هزینه‌ی مسیر

فرض می‌کنیم هر کنش در همه‌ی حالت‌ها هزینه‌ی یکسانی دارد.

اگر یک کنش یکسان برای حالت‌های فیزیکی متفاوت در یک حالت باور،
هزینه‌های مختلفی داشته باشد،
تعریف هزینه کمی دشوار می‌شود.

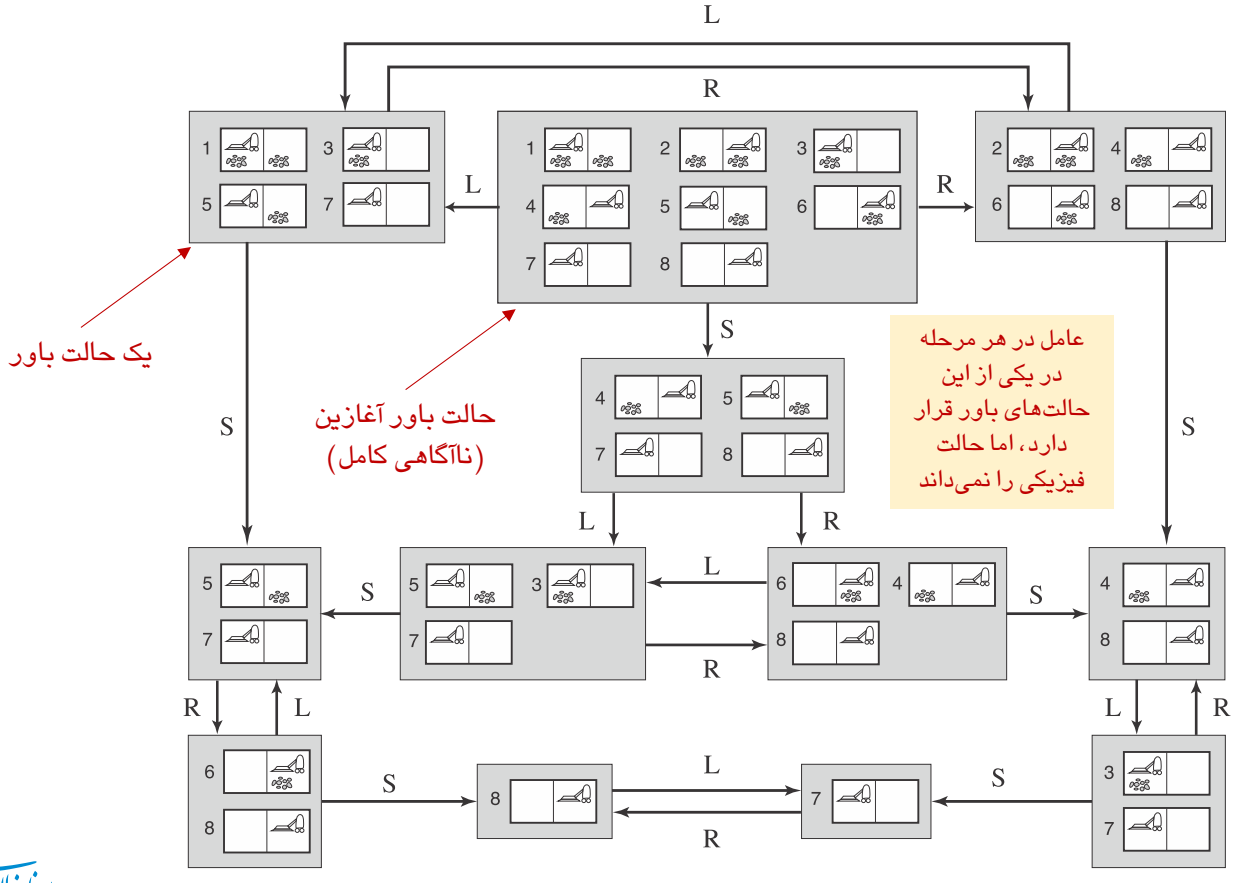
جستجو در فضای باور

تعریف‌های انجام شده، امکان ساخت خودکار مسئله‌ی فضای باور را ایجاد می‌کند.

در این مرحله می‌توان از هر روش جستجوی موجود برای حل این مسئله استفاده کرد.

جستجو در فضای باور

مثال: بخش دسترس‌پذیر فضای حالت باور برای دنیای جاروبرقی بدون حسگر قطعی



Prepared by Kazim Fouladi | Spring 2017 | 2nd Edition
کاظم فولادی

جستجو در فضای باور

کارآیی

حل مسئله‌ی بدون حسگر، به‌ندرت با الگوریتم‌های تاکنون تشریح‌شده امکان‌پذیر است.
(زیرا اندازه‌ی فضای باور بسیار بزرگ است.)

برای مثال:
حالت باور آغازین برای یک دنیای جاروبرقی 10 * 10:
حاوی $100 \times 2^{100} \cong 10^{32}$ حالت فیزیکی است.

راه حل:

الف) بازنمایی حالت باور در یک توصیف متراکم‌تر (مثل بازنمایی صوری/منطق)
ب) اجتناب از الگوریتم‌های جستجوی استاندارد و عدم برخورد با حالت باور به‌صورت جعبه سیاه
⇐ **الگوریتم‌های جستجوی حالت باور افزایشی** (incremental belief-state search algorithms)

جستجو با مشاهدات

SEARCHING WITH OBSERVATION

برای یک مسئله‌ی مشاهده‌پذیر جزئی عمومی
باید مشخص کنیم که محیط چگونه ادراکات را برای عامل تولید می‌کند:

PERCEPT(s) تابع ادراک

جستجو با مشاهدات

تابع ادراک $\text{PERCEPT}(s)$

SEARCHING WITH OBSERVATION

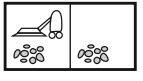
تابع ادراک $\text{PERCEPT}(s)$	
حس کردن قطعی *Deterministic Sensing*	**حس کردن غیرقطعی** *Non-deterministic Sensing*
ادراک عامل برای حالت s	مجموعه‌ی ادراک‌های ممکن عامل برای حالت s

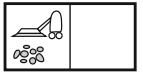
برای مسائل مشاهده‌پذیر کامل: $\text{PERCEPT}(s) = s$

برای مسائل بدون حسگر: $\text{PERCEPT}(s) = null$

وقتی مشاهدات جزئی هستند، یک ادراک می‌تواند توسط حالت‌های متعدد تولید شود.

مثال: ادراک $[A, dirty]$ در دنیای جاروبرقی می‌تواند توسط دو حالت 1 و 3 تولید شود:

1 

3 

جستجو با مشاهدات

گذر در فضای باور

گذر از یک حالت باور به یک حالت باور دیگر به **سه مرحله** تقسیم می‌شود:

۱ پیش‌بینی *Prediction*

با داشتن یک کنش a در حالت باور b، حالت باور پیش‌بینی شده:

$$\hat{b} = \text{PREDICT}(b, a)$$

۲ پیش‌بینی مشاهده *Observation Prediction*

تعیین مجموعه‌ی ادراک‌های ممکن در حالت باور پیش‌بینی شده

$$\text{POSSIBLE-PERCEPTS}(\hat{b}) = \{o : o = \text{PERCEPT}(s) \text{ and } s \in \hat{b}\}$$

۳ به‌هنگام‌سازی *Update*

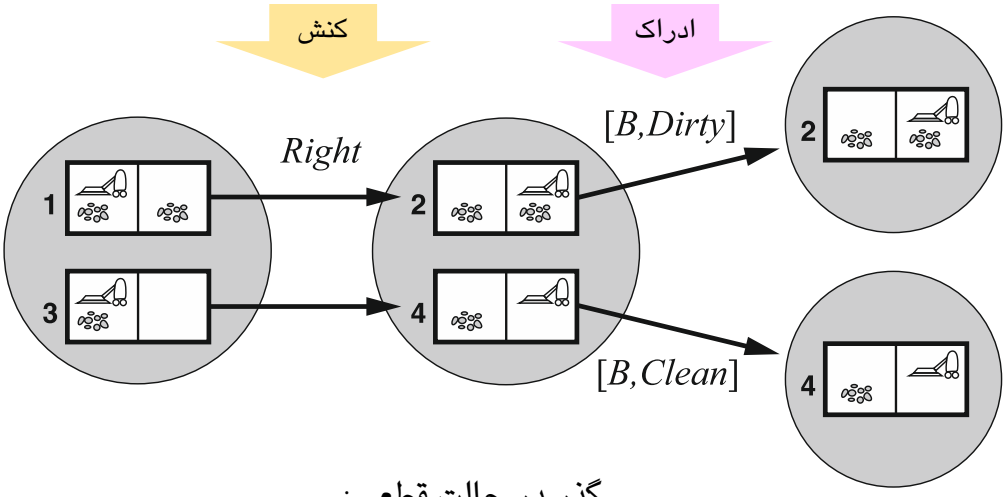
تعیین حالت باور جدید حاصل از یک ادراک

$$b_o = \text{UPDATE}(\hat{b}, o) = \{s : o = \text{PERCEPT}(s) \text{ and } s \in \hat{b}\}$$

حالت باور جدید، مجموعه‌ی حالت‌ها در حالت باور پیش‌بینی شده است که می‌تواند آن ادراک را تولید کند.

جستجو با مشاهدات

گذر در فضای باور: مثال (گذر در حالت قطعی)



گذر در حالت قطعی:

کنش *Right* در حالت باور اولیه اعمال می‌شود

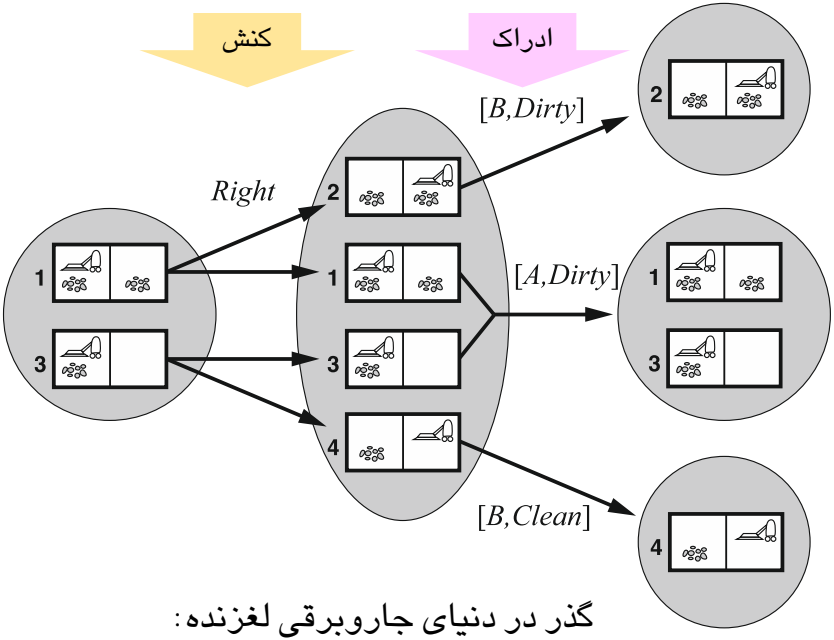
که منجر به یک حالت باور جدید با دو حالت فیزیکی ممکن وابسته به ادراکات زیر می‌شود.

$[B,\ dirty], \quad [B,\ Clean]$

* در حس کردن قطعی، حالت‌های باور برای ادراکات مختلف به‌طور جداگانه،
یک افراز از حالت باور پیش‌بینی شده‌ی اصلی را تشکیل می‌دهند.

جستجو با مشاهدات

گذر در فضای باور: مثال (گذر در حالت غیرقطعی)



گذر در دنیای جاروبرقی لغزنده:

اعمال کنش *Right* در حالت باور اولیه، چهار حالت فیزیکی می‌دهد
که منجر به سه حالت باور جدید وابسته به ادراکات زیر می‌شود:

$[B,\ dirty], \quad [A,\ dirty], \quad [B,\ Clean]$

* حالت باور به‌هنگام‌شده نمی‌تواند بزرگ‌تر از حالت باور پیش‌بینی‌شده باشد،
زیرا مشاهدات کمک می‌کنند عدم اطمینان در مقایسه با حالت بدون حسگر کاهش یابد.

جستجو با مشاهدات

حل مسائل مشاهده‌پذیر جزئی

حالت‌های باور ممکن حاصل از یک کنش و ادراک متوالی بعدی:
(با قرار دادن سه‌مرحله‌ی پیش‌بینی، پیش‌بینی مشاهده و به‌هنگام‌سازی در کنار هم)

$$\text{RESULTS}(b,a) = \{b_o : b_o = \text{UPDATE}(\text{PREDICT}(b,a),o) \text{ and } o \in \text{POSSIBLE-PERCEPTS}(\text{PREDICT}(b,a))\}$$

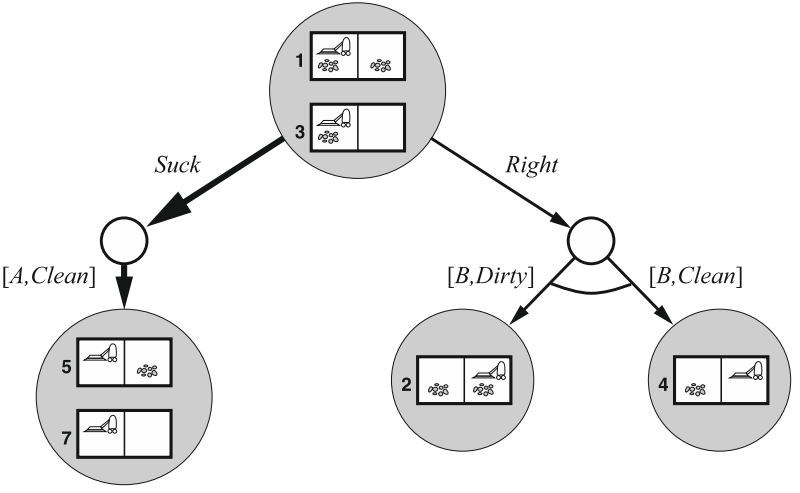
حل مسائل مشاهده‌پذیر جزئی

با داشتن این فرمول‌بندی، جستجوی AND-OR می‌تواند برای یافتن یک راه‌حل مستقیماً اعمال شود.
(برخورد با حالت‌های باور به صورت جعبه سیاه)

حل مسائل مشاهده‌پذیر جزئی

مثال

سطح اول یک درخت جستجوی AND-OR برای ادراک آغازین $[A,\ dirty]$
در دنیای جاروبرقی با حس کردن محلی



راه حل یک طرح شرطی است:

$$[Suck,\ Right,\ \textbf{if}\ BState = \{6\}\ \textbf{then}\ Suck\ \textbf{else}\ [\,]]$$

یک عامل برای محیط‌های مشاهده‌پذیر جزئی

عامل حل مسئله برای محیط‌های مشاهده‌پذیر جزئی:

۱) فرمول‌بندی مسئله
۲) فراخوانی الگوریتم جستجو (مثل جستجوی گرافی AND-OR) و حل مسئله
۳) اجرای راه‌حل

تفاوت‌ها با عامل حل مسئله‌ی ساده

- راه‌حل یک طرح اقتضائی است (به‌جای یک دنباله)
- عامل نیاز دارد حالت باور خودش را (در حال اجرای کنش‌ها و دریافت ادراکات) نگه دارد.

نگهداری حالت باور، مشابه گام به‌هنگام‌سازی در مرحله‌ی جستجو است:

$$b' = \text{UPDATE}(\text{PREDICT}(b, a), o)$$

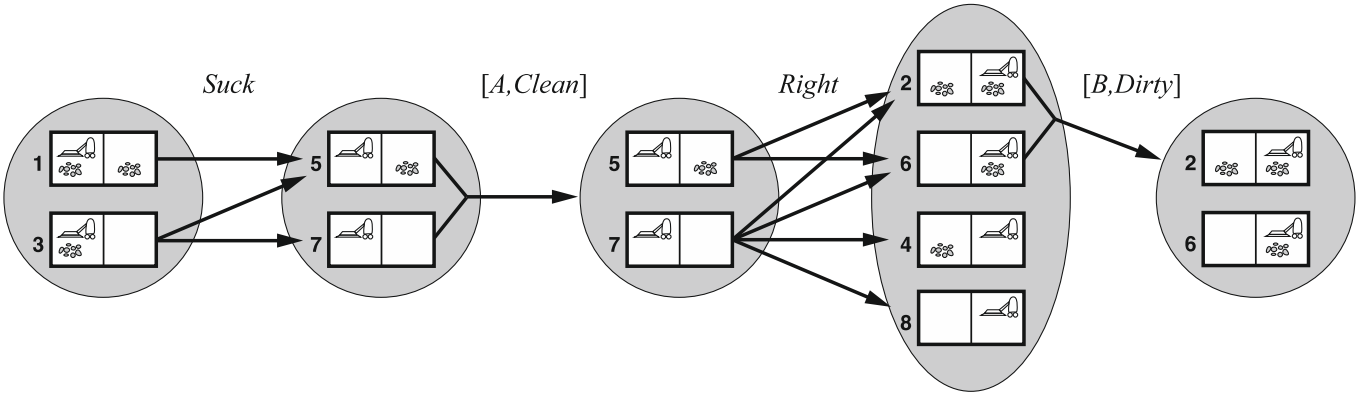
حالت باور جدید — حالت باور — کنش — مشاهده

یک عامل برای محیط‌های مشاهده‌پذیر جزئی

مثال: دنیای جاروبرقی کودکستان

KINDERGARTEN VACUUM WORLD

یک خانه می‌تواند مجدداً آلوده شود، مگر اینکه عامل به‌طور فعالانه در همان لحظه آن را تمیز کند.



حالت‌های باور حاصل برای دو چرخه‌ی پیش‌بینی–به‌هنگام‌سازی
برای یک عامل در دنیای جاروبرقی کودکستان

یک عامل برای محیط‌های مشاهده‌پذیر جزئی

نگهداری حالت باور

MAINTAINING BELIEF STATE

نگهداری حالت باور، یک کارکرد مرکزی برای همه‌ی سیستم‌های هوشمند در یک محیط مشاهده‌پذیر جزئی است.

(شامل اکثر مسائل دنیای واقعی)

نام‌های دیگر

نظارت *Monitoring*	فیلتر کردن *Filtering*	تخمین حالت *Estimation*

گام **پیش‌بینی–به‌هنگام‌سازی** باید به‌اندازه‌ی ورود ادراک‌ها **سریع** باشد؛
در غیر این صورت عامل عقب می‌افتد.

* در محیط‌های پیچیده‌تر محاسبه‌ی دقیق امکان‌ناپذیر می‌شود و تقریب‌ها باید استفاده شوند.

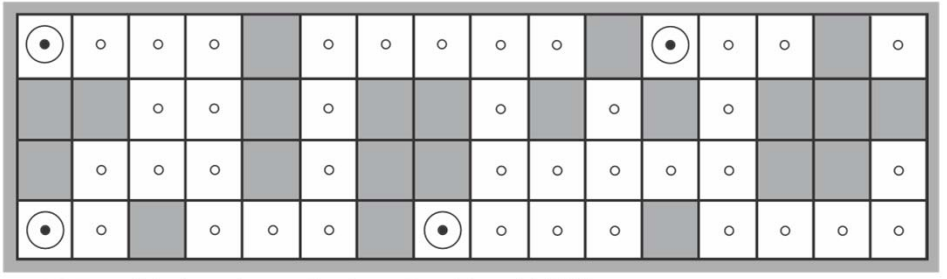
یک عامل برای محیط‌های مشاهده‌پذیر جزئی

نگهداری حالت باور: مثال (مکان‌یابی)

LOCALIZATION

مکان‌یابی، یک مسئله با مشاهدات جزئی است.

مثلاً: تعیین موقعیت یک ربات خطادار در یک راهرو با ۴ حسگر بیان‌کننده‌ی حالت سلول‌ها در ۴ همسایه‌ی آن (آزاد/دیوار)

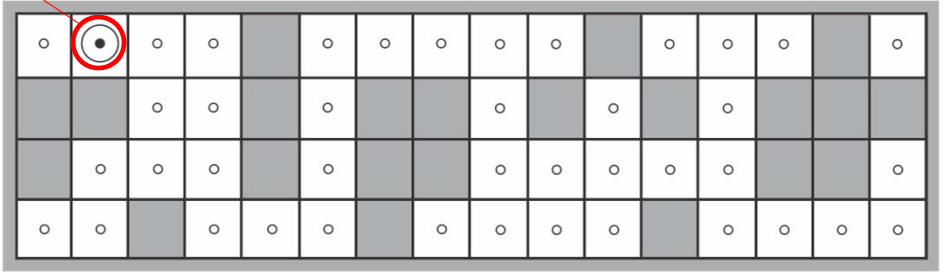


$E_1 =$ W (N, S)

(a) Possible locations of robot after E_1 = NSW

Move
(nondeterministic)

ربات باید اینجا باشد پس از E_1, Move, E_2

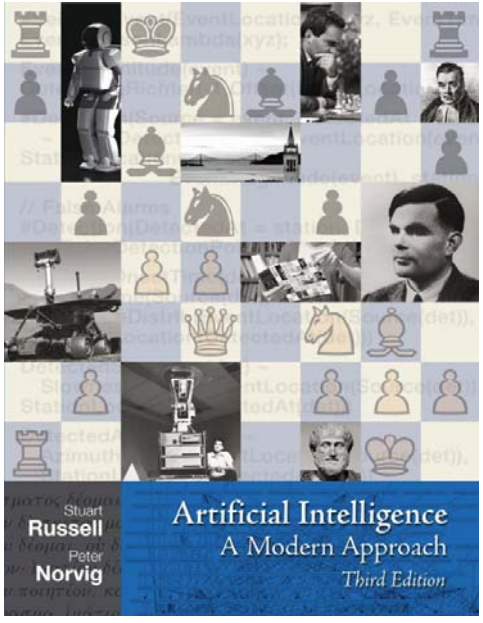


$E_2 =$ (N, S)

(b) Possible locations of robot After E_1 = NSW, E_2 = NS

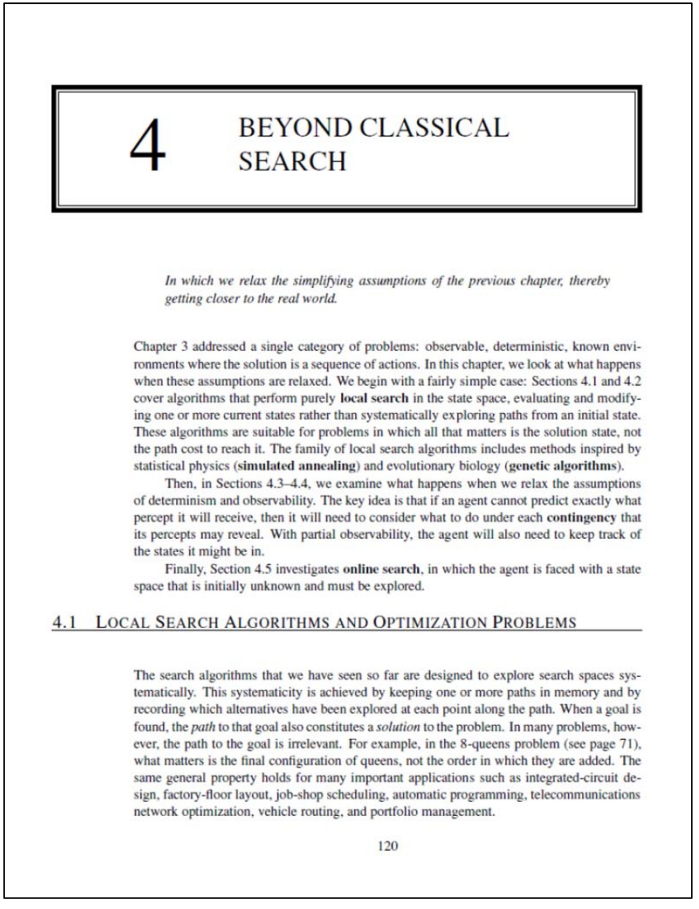
UPDATE(PREDICT(UPDATE(b,NSW),*Move*),NS)

منبع اصلی



Stuart Russell and Peter Norvig,
Artificial Intelligence: A Modern Approach,
3rd Edition, Prentice Hall, 2010.

Chapter 4 (4.4)



4 BEYOND CLASSICAL SEARCH

In which we relax the simplifying assumptions of the previous chapter, thereby getting closer to the real world.

Chapter 3 addressed a single category of problems: observable, deterministic, known environments where the solution is a sequence of actions. In this chapter, we look at what happens when these assumptions are relaxed. We begin with a fairly simple case: Sections 4.1 and 4.2 cover algorithms that perform purely **local search** in the state space, evaluating and modifying one or more current states rather than systematically exploring paths from an initial state. These algorithms are suitable for problems in which all that matters is the solution state, not the path cost to reach it. The family of local search algorithms includes methods inspired by statistical physics (**simulated annealing**) and evolutionary biology (**genetic algorithms**).

Then, in Sections 4.3–4.4, we examine what happens when we relax the assumptions of determinism and observability. The key idea is that if an agent cannot predict exactly what percept it will receive, then it will need to consider what to do under each **contingency** that its percepts may reveal. With partial observability, the agent will also need to keep track of the states it might be in.

Finally, Section 4.5 investigates **online search**, in which the agent is faced with a state space that is initially unknown and must be explored.

4.1 Local Search Algorithms and Optimization Problems

The search algorithms that we have seen so far are designed to explore search spaces systematically. This systematicity is achieved by keeping one or more paths in memory and by recording which alternatives have been explored at each point along the path. When a goal is found, the *path* to that goal also constitutes a *solution* to the problem. In many problems, however, the path to the goal is irrelevant. For example, in the 8-queens problem (see page 71), what matters is the final configuration of queens, not the order in which they are added. The same general property holds for many important applications such as integrated-circuit design, factory-floor layout, job-shop scheduling, automatic programming, telecommunications network optimization, vehicle routing, and portfolio management.

120