

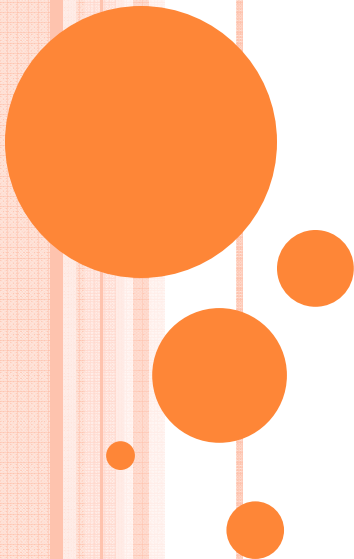
هوش مصنوعی

۷- عوامل‌های منطقی

ارائه دهنده:

احمد نیک آبادی

E-mail: nickabadi@aut.ac.ir



فهرست مطالب

- عامل‌های مبتنی بر دانش
- دنیای WUMPUS
- منطق
- منطق گزاره‌ای
- اثبات قضیه گزاره‌ای
- بررسی مدل گزاره‌ای
- عامل‌های مبتنی بر منطق گزاره‌ای

عوامل‌های مبتنی بر دانش

○ عامل مبتنی بر دانش (knowledge-based agent):

- عاملی که عملکرد آن وابسته به استنتاج بر مبنای پایگاه دانشی است که درون عامل تعبیه شده است.

○ پایگاه دانش (knowledge base (KB):

- اصلی‌ترین جزء یک عامل مبتنی بر دانش است که از مجموعه‌ای از جملات تشکیل شده است.

○ جمله (sentence):

- بیانگر ادعایی در مورد دنیای پیرامون است.
- هر جمله پایگاه دانش را یک اصل موضوعه (axiom) نیز می‌گویند.

○ زبان ارائه دانش (knowledge representation language):

- زبانی که جملات پایگاه دانش با استفاده از آن بیان می‌شوند.

عامل‌های مبتنی بر دانش

- برای افزودن جملات جدید به پایگاه دانش از یک تابع TELL و برای سوال از پایگاه دانش از تابع ASK استفاده می‌شود.
- ادراکات عامل از طریق تابع TELL به عامل منتقل شده و عمل انتخابی عامل از طریق تابع ASK مشخص می‌شود.

○ استنتاج (inference):

- استخراج جملات جدید از جملات موجود در پایگاه داده.
- جملات استخراج شده می‌بایست از جملات موجود در پایگاه دانش تبعیت کنند.

○ دو سطح مختلف بررسی عامل:

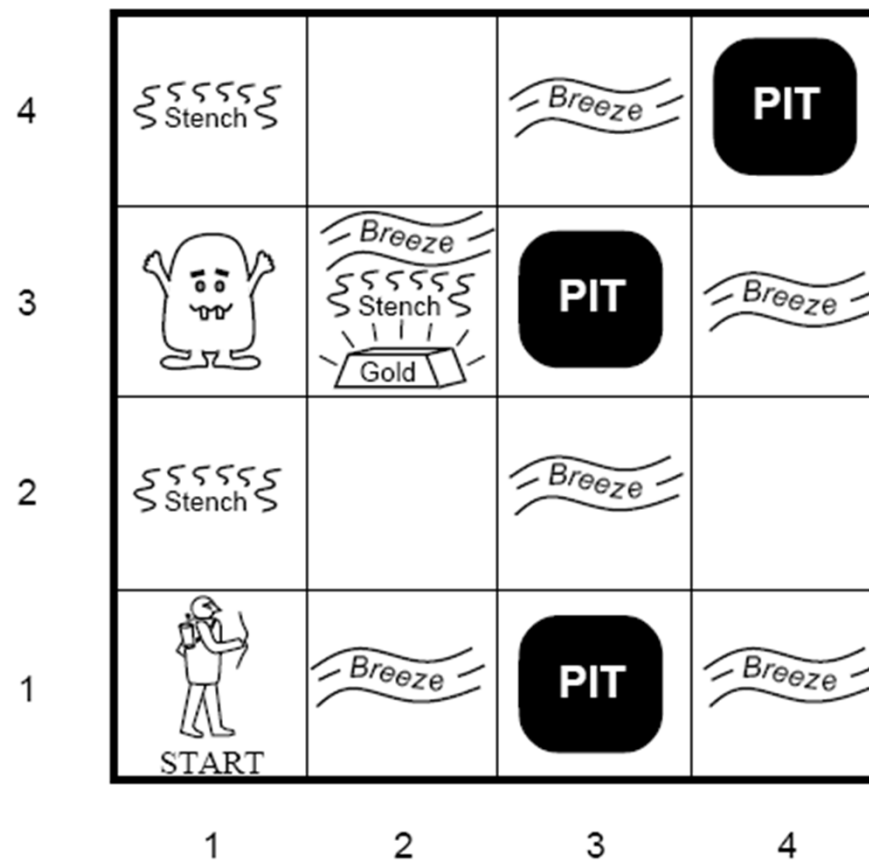
- سطح دانش (knowledge level): بیان این که عامل چه چیزی می‌داند و اهداف آن چیست.
- سطح پیاده سازی (implementation level): نحوه پیاده سازی بخش‌های مختلف عامل.

عامل‌های مبتنی بر دانش

○ روش‌های ایجاد عامل‌های مبتنی بر دانش:

- روش اعلانی (declarative):
 - ایجاد عامل با گفتن (TELL) آنچه مورد نیاز عامل است به آن.
 - در ابتدای کار پایگاه دانش عامل تهی است و به تدریج کامل می‌شود.
- روش رویه‌ای (procedural):
 - رفتارهای مطلوب عامل مستقیماً در کد عامل پیاده سازی می‌شود.
- روش یادگیری (learning):
 - مکانیزم‌های لازم برای یادگیری در اختیار عامل قرار داده می‌شود.
 - پایگاه دانش بر اساس دریافت‌های عامل کامل می‌شود.
 - عامل‌های یادگیرنده می‌توانند به طور کاملاً خودکار عمل کنند.

دنیای WUMPUS



دنیای WUMPUS

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2	3,2	4,2
OK			
1,1	2,1	3,1	4,1
A			
OK	OK		

A = Agent
B = Breeze
G = Glitter, Gold
OK = Safe square
P = Pit
S = Stench
V = Visited
W = Wumpus

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2	3,2	4,2
OK	P?		
1,1	2,1	3,1	4,1
V	A	P?	
OK	B		
	OK		

1,4	2,4	3,4	4,4
1,3	W!	3,3	4,3
1,2	A	3,2	4,2
S			
OK	OK		
1,1	2,1	3,1	4,1
V	B	P!	
OK	V		
	OK		

A = Agent
B = Breeze
G = Glitter, Gold
OK = Safe square
P = Pit
S = Stench
V = Visited
W = Wumpus

1,4	2,4	3,4	4,4
	P?		
1,3	W!	2,3	3,3
		A	P?
		S G	
		B	
1,2	S	3,2	4,2
V			
OK	V		
	OK		
1,1	2,1	3,1	4,1
V	B	P!	
OK	V		
	OK		

منطق

○ هر زبان ارائه دانش از دو بخش تشکیل می‌شود:

• **نحو (syntax):**

○ ساختار کلیه جملات زبان را مشخص می‌کند.

○ مثال: جمله $x+y=4$ از نظر نحوی درست و جمله $x4y+=$ از نظر نحوی غلط است.

• **معنی (semantic):**

○ درستی هر جمله را بر اساس هر جهان ممکن (possible world) مشخص می‌کند.

○ مثال: جمله $x+y=4$ در جهانی که $x=1$ و $y=3$ باشد درست و در جهانی که $x=1$ و $y=5$ باشد نادرست است.

○ هر جهان ممکن را یک **مدل (model)** نیز می‌نامند. مدل مفهوم مجردتری نسبت به جهان است.

○ در صورتی که جمله α در مدل m درست باشد، گفته می‌شود که m مدلی از α است.

○ مجموعه کلیه مدل‌های α با نماد $M(\alpha)$ نمایش داده می‌شود.

منطق

○ استلزام (ایجاب) (entailment):

- رابطه‌ای میان جملات α و β که بیان می‌کند

- جمله α مستلزم جمله β است.

- اگر جمله α درست باشد، جمله β هم حتماً درست است.

- جمله β از جمله α نتیجه می‌شود.

- مجموعه مدل‌های α زیر مجموعه‌ای از مجموعه مدل‌های β است.

$$\alpha \models \beta \text{ if and only if } M(\alpha) \subseteq M(\beta)$$

- با نماد $\alpha \models \beta$ نمایش داده می‌شود.

- مثال:

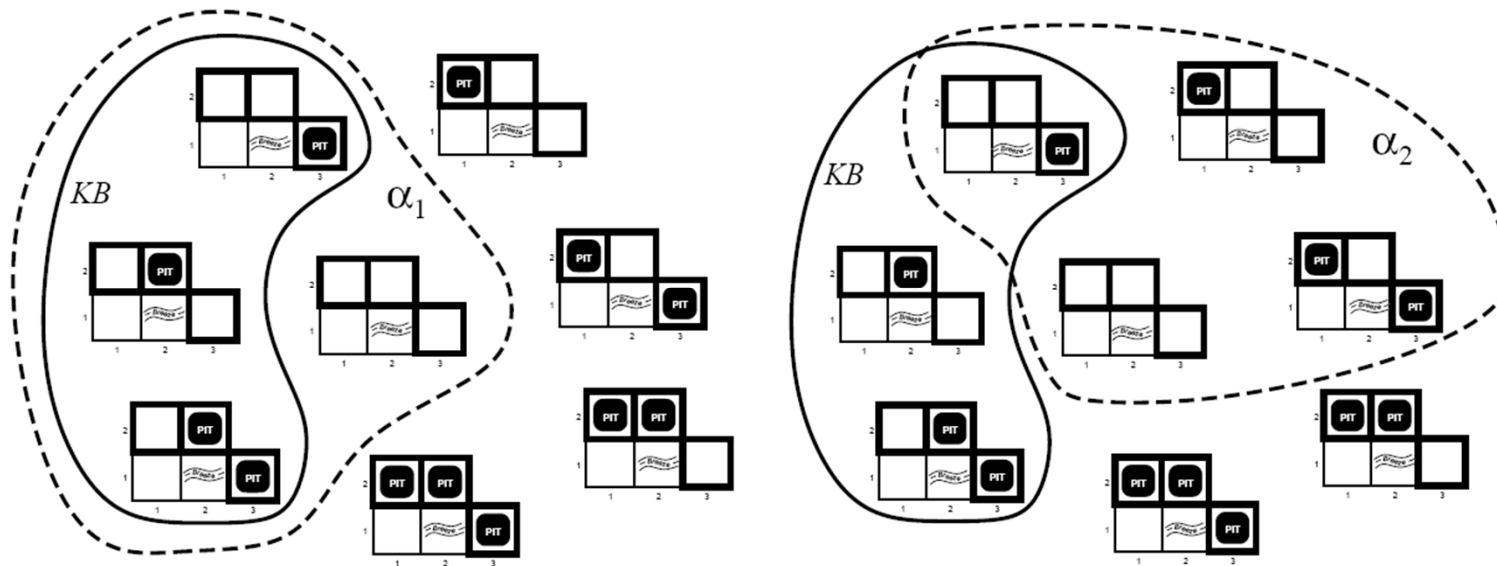
- α : x یک گربه سان است.

- β : x یک پستاندار است.

منطق

مثال: استلزام در دنیای WUMPUS

- α_1 : گودالی در خانه $[1,2]$ نیست. $KB \models \alpha_1$
- α_2 : گودالی در خانه $[2,2]$ نیست. $KB \not\models \alpha_2$
- $KB =$ مجموعه دانش اولیه عامل در مورد محیط + مجموعه دریافت‌ها

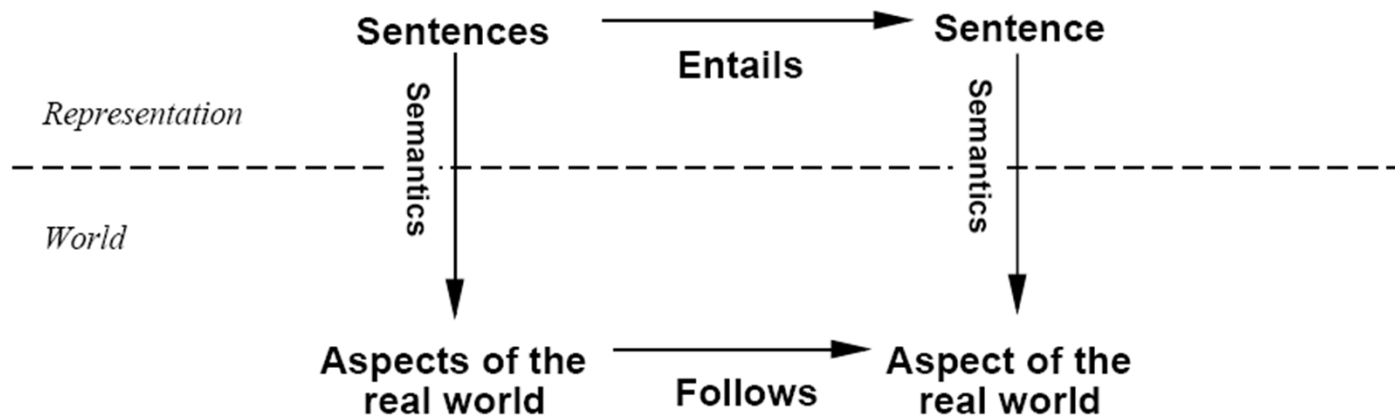


منطق

- استنتاج منطقی (logical inference):
 - استخراج نتایج (جملات) جدید از دانش موجود
- استلزام بیانگر کلیه نتایج قابل حصول از پایگاه دانش و استنتاج یافتن یک جمله خاص است.
- الگوریتم بررسی مدل (model checking):
 - یک الگوریتم استنتاج که بررسی قابل استنتاج بودن یک جمله از پایگاه دانش را با برشمردن کلیه مدل‌های ممکن انجام می‌دهد.
- الگوریتم استنتاج صحیح (sound) یا نگهدارنده درستی (truth-preserving):
 - الگوریتم استنتاجی که تنها جملاتی را استخراج کند که پایگاه دانش موجود مستلزم آنها باشد.
- الگوریتم استنتاج کامل (complete):
 - الگوریتم استنتاجی که به صورت سیستماتیک بتواند بررسی کند که آیا یک جمله از پایگاه دانش موجود نتیجه می‌شود یا خیر.

منطق

- در صورتی که پایگاه دانش در دنیای واقعی درست باشد، باید هر جمله نتیجه شده از آن نیز در دنیای واقعی درست باشد.



منطق گزاره‌ای

○ ساختار جملات (syntax) منطق گزاره‌ای:

• جملات ساده (اتمیک)

- از یک نماد گزاره (proposition symbol) تنها تشکیل می‌شوند.
- معمولاً با حروف بزرگ نمایش داده می‌شوند.
- مثال: $P, Q, W_{1,2}, \dots$

• جملات پیچیده

- از ترکیب جملات اتمیک با استفاده از پرانتز و رابط‌های منطقی ایجاد می‌شوند.
- رابط‌های منطقی:
 - نقیض (negation)
 - به یک جمله اتمیک یا نقیض آن جمله لیترال (literal) گفته می‌شود.
 - ترکیب عطفی (conjunction)
 - ترکیب فصلی (disjunction)
 - استلزام (شرطی)
 - از دو بخش مقدمه (شرط) و تالی (نتیجه) تشکیل می‌شود.
 - دوشروطی (هم ارزی)

منطق گزاره‌ای

○ ساختار جملات منطق گزاره‌ای:

$Sentence \rightarrow AtomicSentence \mid ComplexSentence$

$AtomicSentence \rightarrow \mathbf{True} \mid \mathbf{False} \mid Symbol$

$Symbol \rightarrow \mathbf{P} \mid \mathbf{Q} \mid \mathbf{R} \mid \dots$

$ComplexSentence \rightarrow \neg Sentence$

$\mid (Sentence \wedge Sentence)$

$\mid (Sentence \vee Sentence)$

$\mid (Sentence \Rightarrow Sentence)$

$\mid (Sentence \Leftrightarrow Sentence)$

منطق گزاره‌ای

○ معنی (semantic) جملات منطق گزاره‌ای:

- مجموعه قوانینی برای تعیین درستی هر یک از جملات با توجه به یک مدل خاص
- هر مدل یک مقدار درستی (truth value) به هر نماد گزاره‌ای نسبت می‌دهد.

• جملات اتمیک:

- جمله True در همه مدلها درست و جمله False در همه مدلها نادرست است.
- درستی هر نماد گزاره باید مستقیماً در مدل مشخص شود.

• جملات پیچیده:

- نقیض P ($\neg P$) در صورتی درست است که P نادرست باشد و بالعکس.
- ترکیب عطفی $P \wedge Q$ تنها در صورتی درست است که P و Q هر دو درست باشند.
- ترکیب فصلی $P \vee Q$ در صورتی درست است که حداقل یکی از دو جمله P یا Q درست باشد.
- جمله شرطی $P \rightarrow Q$ همواره درست است مگر این که P درست و Q نادرست باشد.
- جمله دوشروطی $P \leftrightarrow Q$ در صورتی درست است که P و Q هر دو درست یا هر دو نادرست باشند.

منطق گزاره‌ای

○ معنی (semantic) جملات منطق گزاره‌ای:

- برای نشان دادن قوانین درستی جملات می‌توان از **جدول درستی** (truth table) نیز استفاده کرد.

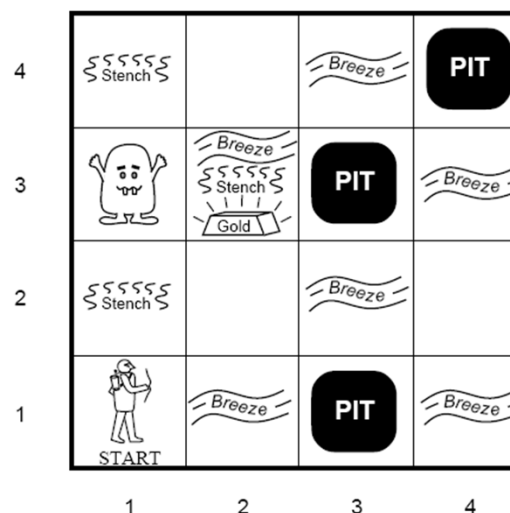
P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
false	false	true	false	false	true	true
false	true	true	false	true	true	false
true	false	false	false	true	false	false
true	true	false	true	true	true	true

منطق گزاره‌ای

○ **مثال:** ایجاد یک پایگاه دانش گزاره‌ای بر اساس دانش قبلی و دریافت‌های عامل

- گودالی در خانه $[1,1]$ وجود ندارد.
- دریافت نسیم در یک خانه به معنای وجود گودال در یکی از خانه‌های مجاور آن است.
- عامل در خانه $[1,1]$ نسیم دریافت نمی‌کند.
- عامل در خانه $[2,1]$ نسیم دریافت می‌کند.

- $R_1: \neg P_{1,1}$
- $R_2: B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
- $R_3: B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- $R_4: \neg B_{1,1}$
- $R_5: B_{2,1}$



منطق گزاره‌ای

○ مثال: یک استنتاج ساده بر مبنای پایگاه دانش عامل

- KB در سه تا از ۱۲۸ مدل ممکن درست است.
- در همه مدل‌هایی که KB درست است $P_{1,2}$ نادرست است.

$B_{1,1}$	$B_{2,1}$	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$P_{2,2}$	$P_{3,1}$	R_1	R_2	R_3	R_4	R_5	KB
false	false	false	false	false	false	false	true	true	true	true	false	false
false	false	false	false	false	false	true	true	true	false	true	false	false
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
false	true	false	false	false	false	false	true	true	false	true	true	false
false	true	false	false	false	false	true	true	true	true	true	true	<u>true</u>
false	true	false	false	false	true	true	true	true	true	true	true	<u>true</u>
false	true	false	false	true	false	false	true	false	false	true	true	false
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
true	true	true	true	true	true	true	false	true	true	false	true	false

استدلال در منطق گزاره‌ای

- استنتاج بر اساس مدلهای ممکن در صورت زیاد بودن تعداد مدل‌ها کارا نیست.
- می‌توان قوانین استنتاج را مستقیماً به جملات موجود در پایگاه دانش اعمال و اثباتی برای جمله مورد نظر ایجاد کرد.
- هم ارزی منطقی (logical equivalence):
 - دو جمله α و β هم ارز هستند در صورتی که در مجموعه یکسانی از مدل‌ها درست باشند.

$$\alpha \equiv \beta \text{ if and only if } \alpha \models \beta \text{ and } \beta \models \alpha$$

استدلال در منطق گزاره‌ای

○ روابط هم ارزی:

$$(\alpha \wedge \beta) \equiv (\beta \wedge \alpha) \quad \text{commutativity of } \wedge$$

$$(\alpha \vee \beta) \equiv (\beta \vee \alpha) \quad \text{commutativity of } \vee$$

$$((\alpha \wedge \beta) \wedge \gamma) \equiv (\alpha \wedge (\beta \wedge \gamma)) \quad \text{associativity of } \wedge$$

$$((\alpha \vee \beta) \vee \gamma) \equiv (\alpha \vee (\beta \vee \gamma)) \quad \text{associativity of } \vee$$

$$\neg(\neg\alpha) \equiv \alpha \quad \text{double-negation elimination}$$

$$(\alpha \Rightarrow \beta) \equiv (\neg\beta \Rightarrow \neg\alpha) \quad \text{contraposition}$$

$$(\alpha \Rightarrow \beta) \equiv (\neg\alpha \vee \beta) \quad \text{implication elimination}$$

$$(\alpha \Leftrightarrow \beta) \equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)) \quad \text{biconditional elimination}$$

$$\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta) \quad \text{de Morgan}$$

$$\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta) \quad \text{de Morgan}$$

$$(\alpha \wedge (\beta \vee \gamma)) \equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma)) \quad \text{distributivity of } \wedge \text{ over } \vee$$

$$(\alpha \vee (\beta \wedge \gamma)) \equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma)) \quad \text{distributivity of } \vee \text{ over } \wedge$$

استدلال در منطق گزاره‌ای

○ معتبر بودن (validity):

- یک جمله در صورتی معتبر (valid) است که در همه مدل‌ها درست باشد.
- جمله معتبر جمله همیشه درست (tautology) نیز گفته می‌شود.
- هر جمله معتبر هم ارز جمله True است.

○ برای هر دو جمله α و β ، $\alpha \models \beta$ اگر و تنها اگر $\alpha \rightarrow \beta$ یک جمله معتبر باشد.

- برای اثبات این که β از α نتیجه می‌شود کافی است نشان داده شود که $\alpha \rightarrow \beta$ یک جمله معتبر است.

استدلال در منطق گزاره‌ای

○ صدق پذیری (satisfiability):

- یک جمله در صورتی صدق پذیر است که در برخی مدل‌ها درست باشد.
- صدق پذیری یک جمله را با بررسی درستی آن در کلیه مدل‌های ممکن می‌توان بررسی کرد.
- اگر یک جمله غیرصدق پذیر باشد، نقیض آن یک جمله معتبر (همواره درست) است.
- نتیجه سودمند:

$\alpha \models \beta$ if and only if $(\alpha \wedge \neg \beta)$ is unsatisfiable

- اثبات به روش فوق، همان برهان خلف است.

استدلال در منطق گزاره‌ای

○ اثبات (proof):

- دنباله‌ای از نتایج که به هدف مورد نظر منجر می‌شوند.
- برای یافتن یک اثبات می‌توان از الگوریتم‌های جستجو به همراه مجموعه‌ای از قوانین استنتاج استفاده کرد.

○ قوانین استنتاج:

- قیاس استثنائی (Modus Ponens):

$$\frac{\alpha \Rightarrow \beta, \alpha}{\beta}$$

- حذف عطف (And-Elimination):

$$\frac{\alpha \wedge \beta}{\alpha}$$

- کلیه قوانین هم ارزی

استدلال در منطق گزاره‌ای

○ مثال: استنتاج بر اساس قوانین استنتاج و قواعد هم ارزی

- $R_1: \neg P_{1,1}$
- $R_2: B_{1,1} \leftrightarrow (P_{1,2} \vee P_{2,1})$
- $R_3: B_{2,1} \leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- $R_4: \neg B_{1,1}$
- $R_5: B_{2,1}$
- $R_6: (B_{1,1} \rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$
- $R_7: ((P_{1,2} \vee P_{2,1}) \rightarrow B_{1,1})$
- $R_8: (\neg B_{1,1} \rightarrow \neg (P_{1,2} \vee P_{2,1}))$
- $R_9: \neg (P_{1,2} \vee P_{2,1})$
- $R_{10}: \neg P_{1,2} \wedge \neg P_{2,1}$

استدلال در منطق گزاره‌ای

○ یکنواختی (monotonicity):

- یکی از ویژگی‌های سیستم‌های منطقی که بیان می‌کند که جملات قابل استنتاج از یک پایگاه دانش با افزایش دانش جدید تنها افزایش می‌یابد و هیچ گاه کاهش نمی‌یابد.

$$\text{if } KB \models \alpha \text{ then } KB \wedge \beta \models \alpha$$

- به دلیل یکنواختی سیستم منطقی، هر زمان که مقدمات یک قاعده استنتاج آماده باشد می‌توان از آن قاعده استفاده کرد صرف نظر از این که بعداً چه جملاتی به پایگاه دانش اضافه می‌شوند.

استدلال در منطق گزاره‌ای

○ رزولوشن (تحلیل) (resolution):

- یک قاعده استنتاج صحیح (sound) که در صورت ترکیب با یک الگوریتم جستجوی کامل یک الگوریتم استنتاج کامل ارائه می‌کند.

○ قاعده رزولوشن واحد (unit resolution):

- یک عبارت (عبارت فصلی از لیترال‌ها) و یک لیترال دریافت و یک عبارت جدید باز می‌گرداند (m نقیض لیترال l_i است):

$$\frac{l_1 \vee \dots \vee l_k, \quad m}{l_1 \vee \dots \vee l_{i-1} \vee l_{i+1} \vee \dots \vee l_k}$$

○ قاعده رزولوشن کامل:

$$\frac{l_1 \vee \dots \vee l_k, \quad m_1 \vee \dots \vee m_n}{l_1 \vee \dots \vee l_{i-1} \vee l_{i+1} \vee \dots \vee l_k \vee m_1 \vee \dots \vee m_{j-1} \vee m_{j+1} \vee \dots \vee m_n}$$

استدلال در منطق گزاره‌ای

- در قاعده رزولوشن تنها از عبارات (clause) (ترکیب فصلی لیترال‌ها) استفاده میشود.
- هر جمله‌ای از منطق گزاره‌ای معادل ترکیب عطفی تعدادی عبارت (CNF) است.

- شکل نرمال عطفی (conjunctive normal form(CNF)):
 - جملات حاصل از ترکیب عطفی تعدادی عبارت

$$(l_{1,1} \vee \dots \vee l_{1,k}) \wedge \dots \wedge (l_{n,1} \vee \dots \vee l_{n,k})$$

استدلال در منطق گزاره‌ای

function PL-RESOLUTION(KB, α) **returns** *true* or *false*

inputs: KB , the knowledge base, a sentence in propositional logic

α , the query, a sentence in propositional logic

$clauses \leftarrow$ the set of clauses in the CNF representation of $KB \wedge \neg\alpha$

$new \leftarrow \{ \}$

loop do

for each C_i, C_j **in** $clauses$ **do**

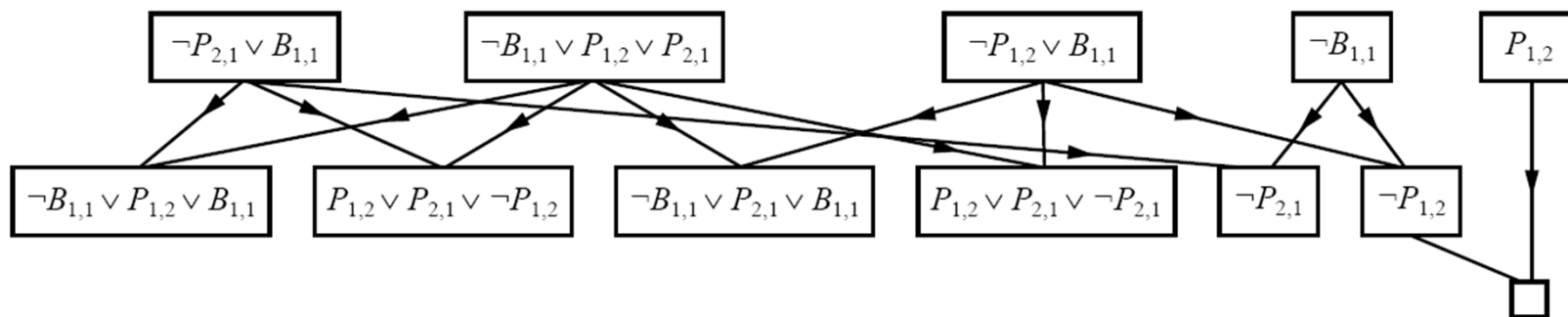
$resolvents \leftarrow$ PL-RESOLVE(C_i, C_j)

if $resolvents$ contains the empty clause **then return** *true*

$new \leftarrow new \cup resolvents$

if $new \subseteq clauses$ **then return** *false*

$clauses \leftarrow clauses \cup new$



استدلال در منطق گزاره‌ای

○ الگوریتم رزولوشن یک الگوریتم کامل است:

- بستار رزولوشن یک مجموعه از عبارات متناهی است.
- اگر یک مجموعه از عبارات صدق ناپذیر باشد، بستار رزولوشن آن حاوی عبارت تهی خواهد بود.

○ بستار رزولوشن (resolution closure):

- مجموعه کلیه عبارات قابل حصول از یک مجموعه عبارت با اعمال تکراری قاعده رزولوشن به عبارات موجود در مجموعه و عبارات حاصل شده از آن.

استدلال در منطق گزاره‌ای

○ ایده: ارائه الگوریتم‌های استنتاج محدودتر و در عین حال کارآتر از الگوریتم رزولوشن برای مواردی که جملات پایگاه داده شرایط ویژه‌ای دارند.

○ عبارات معین (definite clauses):

- عبارات فصلی از لیترال‌ها که دقیقاً یکی از لیترال‌های آن مثبت است.

- مثال: $\neg L_{1,1} \vee \neg B \vee B_{1,1}$

○ عبارات هورن (Horn clauses):

- عبارات فصلی از لیترال‌ها که حداکثر یکی از لیترال‌های آن مثبت است.

- عبارات هورن تحت عمل رزولوشن بسته هستند.

○ دلایل مورد توجه بودن پایگاه‌های دانش حاوی عبارات معین:

- هر عبارت معین را می‌توان به صورتی یک استلزام نوشت.

- مثال: $\neg L_{1,1} \vee \neg B \vee B_{1,1}$ معادل $(L_{1,1} \wedge B) \rightarrow B_{1,1}$ است.

- می‌توان از الگوریتم‌های زنجیره‌ای پیش رو یا زنجیره‌ای عقب‌گرد برای استنتاج استفاده کرد.

- تصمیم‌گیری در مورد استلزام در زمان خطی بر حسب اندازه پایگاه دانش قابل انجام است.

استدلال در منطق گزاره‌ای

○ عبارات معین و عبارات هورن:

$$CNFSentence \rightarrow Clause_1 \wedge \dots \wedge Clause_n$$

$$Clause \rightarrow Literal_1 \vee \dots \vee Literal_m$$

$$Literal \rightarrow Symbol \mid \neg Symbol$$

$$Symbol \rightarrow P \mid Q \mid R \mid \dots$$

$$HornClauseForm \rightarrow DefiniteClauseForm \mid GoalClauseForm$$

$$DefiniteClauseForm \rightarrow (Symbol_1 \wedge \dots \wedge Symbol_l) \Rightarrow Symbol$$

$$GoalClauseForm \rightarrow (Symbol_1 \wedge \dots \wedge Symbol_l) \Rightarrow False$$

استدلال در منطق گزاره‌ای

○ زنجیره پیش‌رو (Forward chaining):

- مشخص می‌کند که آیا گزاره تنهای q از پایگاه دانش KB قابل استنتاج هست یا خیر.
- از عبارات معین در پایگاه دانش استفاده می‌شود.
- به صورت تکراری، اگر کلیه شروط یک استلزام درست باشد نتیجه آن به مجموعه پایگاه داده اضافه می‌شود.
- پیچیدگی زمانی خطی دارد.
- زنجیره نتایج را می‌توان به صورت یک گراف AND-OR نمایش داد.
- الگوریتم صحیح و کاملی برای استنتاج است.
- یک الگوریتم استدلال مبتنی بر داده (data-driven) است.

○ زنجیره عقب‌گرد (backward chaining):

- مشابه الگوریتم زنجیره پیش‌رو است.
- با شروع از جمله هدف q شروع به جستجو می‌کند.
- مشابه یک الگوریتم جستجوی گراف AND-OR است.
- یک الگوریتم استدلال مبتنی بر هدف (goal-directed) است.

استدلال در منطق گزاره‌ای

○ زنجیره پیش‌رو:

```
function PL-FC-ENTAILS?(KB, q) returns true or false  
  inputs: KB, the knowledge base, a set of propositional Horn clauses  
           q, the query, a proposition symbol  
  local variables: count, a table, indexed by clause, initially the number of premises  
                    inferred, a table, indexed by symbol, each entry initially false  
                    agenda, a list of symbols, initially the symbols known to be true in KB  
  
  while agenda is not empty do  
     $p \leftarrow \text{POP}(\text{agenda})$   
    unless inferred[p] do  
       $\text{inferred}[p] \leftarrow \text{true}$   
      for each Horn clause c in whose premise p appears do  
        decrement count[c]  
        if count[c] = 0 then do  
          if HEAD[c] = q then return true  
          PUSH(HEAD[c], agenda)  
  
  return false
```

استدلال در منطق گزاره‌ای

○ زنجیره پیش‌رو و زنجیره عقب‌گرد:

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

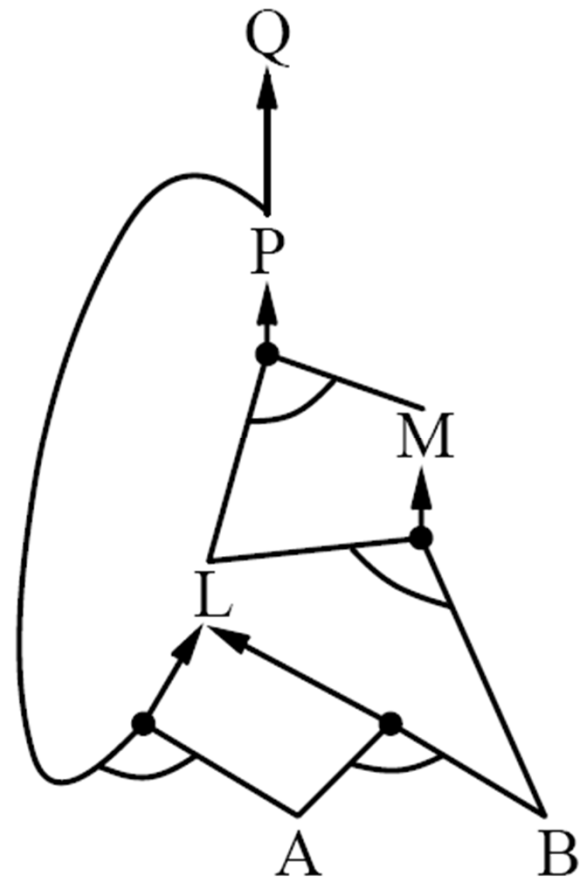
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

$$A \wedge B \Rightarrow L$$

A

B



بررسی مدل گزاره‌ای

○ حل مسأله بررسی صدق پذیری با روش‌های جستجو:

- نیازمند بررسی مدل‌ها (model checking) است.
- با عنوان مسأله SAT شناخته می‌شود.
- روشها:
- الگوریتم بازگشتی کامل (complete backtracking algorithm)
- الگوریتم‌های جستجوی محلی

بررسی مدل گزاره‌ای

○ الگوریتم DPLL:

- یک الگوریتم بازگشتی کامل برای بررسی صدق پذیری یک مجموعه عبارت است.
- یک الگوریتم جستجوی عمق اول است.
- بهبود کارایی الگوریتم با روشهای شهودی:

○ خاتمه زودرس

- در برخی موارد بررسی صدق پذیری یک مجموعه عبارت نیازمند مدل کامل نیست.
- مثال: جمله $(A \vee B) \wedge (A \vee C)$ در صورت درست بودن A درست است.

○ نمادهای خالص

- نمادهای خالص (pure symbols) نمادهایی هستند که لیترالهای آنها در کلیه عبارت مثبت یا منفی باشد.
- در صورتی که مدلی وجود داشته باشد که مجموعه عبارات داده شده در آن صدق کنند، همچنین مدلی وجود خواهد داشت که در آن کلیه نمادهای خالص دارای مقدار درست باشند.

○ عبارات واحد

- عبارت واحد (unit clause) عبارتی است که شامل تنها یک لیترال باشد یا سایر لیترالهای آن مقدار نادرست گرفته باشند.
- کلیه عبارات واحد باید دارای مقدار درست باشند.

بررسی مدل گزاره‌ای

○ الگوریتم‌های جستجوی محلی:

- از الگوریتم‌های جستجوی محلی تپه نوردی و تبرید شیبه سازی شده می‌توان برای حل مسأله SAT استفاده کرد.
- برای تابع ارزیابی می‌توان از تعداد عبارات دارای مقدار نادرست استفاده کرد.
- این الگوریتم‌ها کامل نیستند.

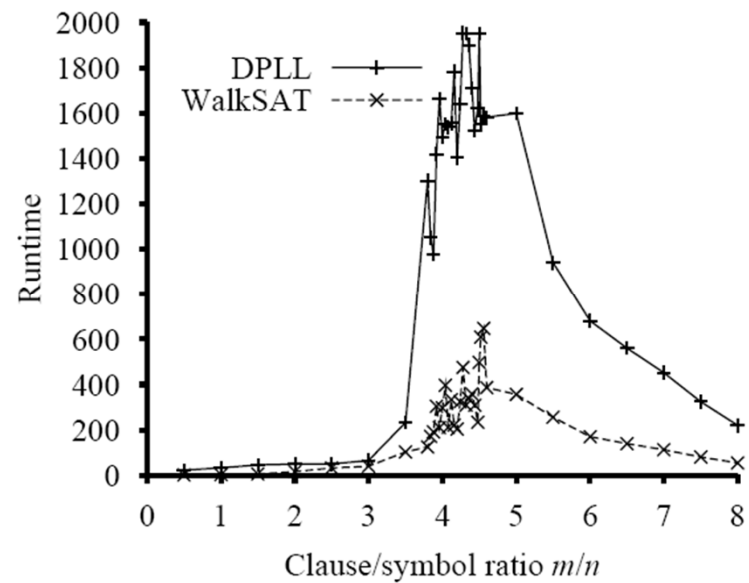
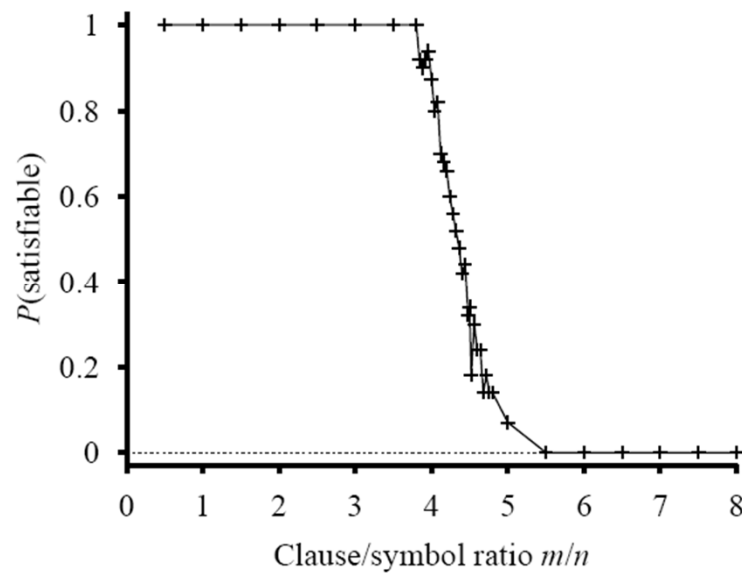
○ الگوریتم WALKSAT:

- در هر مرحله، یک عبارت نادرست انتخاب و مقدار درستی یکی از نمادهای موجود در آن را تغییر می‌دهد.
- نحوه انتخاب نماد برای تغییر:
 - کمترین تداخل: انتخاب نمادی که تعداد عبارات نادرست را در حالت جدید کمینه کند.
 - قدم تصادفی: انتخاب تصادفی یک نماد.
- در صورت بازگرداندن یک مدل یعنی مجموعه عبارات صدق پذیر است.
- در صورت عدم بازگرداندن یک مدل، یا مسأله صدق ناپذیر بوده است یا الگوریتم زمان کافی در اختیار نداشته است.

بررسی مدل گزاره‌ای

○ میزان سختی مسائل SAT:

- جملات 3-CNF (هر عبارت سه لیترال دارد)
- تعداد نمادها: $n=50$
- تعداد عبارات: m



عامل‌های مبتنی بر منطق گزاره‌ای

○ وضعیت فعلی دنیا:

- وجود نسیم در یک خانه بیانگر وجود چاله در یکی از خانه‌های مجاور است.
- وجود بو در یک خانه بیانگر وجود WUMPUS در یکی از خانه‌های مجاور است.

$$\begin{aligned}B_{1,1} &\Leftrightarrow (P_{1,2} \vee P_{2,1}) \\S_{1,1} &\Leftrightarrow (W_{1,2} \vee W_{2,1}) \\&\dots\end{aligned}$$

- یک و فقط یک WUMPUS وجود دارد.

$$W_{1,1} \vee W_{1,2} \vee \dots \vee W_{4,3} \vee W_{4,4}$$

$$\neg W_{1,1} \vee \neg W_{1,2}$$

$$\neg W_{1,1} \vee \neg W_{1,3}$$

...

$$\neg W_{4,3} \vee \neg W_{4,4}$$

عوامل‌های مبتنی بر منطق گزاره‌ای

وضعیت فعلی دنیا:

- جنبه‌های مختلف محیط
 - جنبه‌های (متغیرهای) غیر زمانی (atemporal)
 - همواره ثابت هستند و می‌توان آنها را با یک مجموعه گزاره ثابت نمایش داد.
 - مثال: وجود چاله در خانه $[2,1]$
 - جنبه‌های (متغیرهای) زمانی (fluent)
 - با گذشت زمان تغییر می‌کنند.
 - اضافه کردن مستقیم آنها به پایگاه دانش باعث بروز خطا می‌شود.
 - مثال: استشمام بوی بد
 - راه حل: استفاده از برچسب زمانی $\neg \text{Stench}^0, \text{Stench}^1$
 - ایجاد ارتباط بین متغیرهای زمانی با موقعیت مکانی آنها
- $$\begin{aligned} L_{x,y}^t &\Rightarrow (\text{Breeze}^t \Leftrightarrow B_{x,y}) \\ L_{x,y}^t &\Rightarrow (\text{Stench}^t \Leftrightarrow S_{x,y}) \end{aligned}$$
- بیان تأثیر عمل‌ها (effect axiom)

$$L_{1,1}^0 \wedge \text{FacingEast}^0 \wedge \text{Forward}^0 \Rightarrow (L_{2,1}^1 \wedge \neg L_{1,1}^1)$$

عوامل‌های مبتنی بر منطق گزاره‌ای

○ وضعیت فعلی دنیا:

• مشکل قاب (frame problem):

○ اطلاعات مربوط به بخش‌هایی از محیط که تحت تأثیر عمل انتخابی نیستند به گام بعد منتقل نمی‌شود.

• مثال: $HaveArrow^0$ ، اما در مورد $HaveArrow^1$ هیچ اظهار نظری نمی‌توان کرد.

• راه حل: استفاده از اصول قاب (frame axioms)

$$\begin{aligned} Forward^t &\Rightarrow (HaveArrow^t \Leftrightarrow HaveArrow^{t+1}) \\ Forward^t &\Rightarrow (WumpusAlive^t \Leftrightarrow WumpusAlive^{t+1}) \\ &\dots \end{aligned}$$

• مشکلات روش فوق:

○ مشکل ارائه قاب (representational frame problem)

○ پیچیدگی بالای تعداد اصول مورد نیاز ($O(mn)$)

○ مشکل استنتاج قاب (inferential frame problem)

○ پیچیدگی بالای منعکس کردن اطلاعات فریم جاری به فریم بعدی

عامل‌های مبتنی بر منطق گزاره‌ای

وضعیت فعلی دنیا:

- راه حل مشکلات قاب روش قبل:

- اصول حالت بعد (successor-state axioms) که به جای عمل‌ها بر متغیرهای زمانی تمرکز می‌کنند.

$$F^{t+1} \Leftrightarrow \text{ActionCauses}F^t \vee (F^t \wedge \neg \text{ActionCausesNot}F^t)$$

$$\text{HaveArrow}^{t+1} \Leftrightarrow (\text{HaveArrow}^t \wedge \neg \text{Shoot}^t)$$

$$\begin{aligned} L_{1,1}^{t+1} \Leftrightarrow & (L_{1,1}^t \wedge (\neg \text{Forward}^t \vee \text{Bump}^{t+1})) \\ & \vee (L_{1,2}^t \wedge (\text{South}^t \wedge \text{Forward}^t)) \\ & \vee (L_{2,1}^t \wedge (\text{West}^t \wedge \text{Forward}^t)) . \end{aligned}$$

عامل‌های مبتنی بر منطق گزاره‌ای

○ یک عامل مرکب

```
function HYBRID-WUMPUS-AGENT(percept) returns an action
inputs: percept, a list, [stench,breeze,glitter,bump,scream]
persistent: KB, a knowledge base, initially the atemporal “wumpus physics”
           t, a counter, initially 0, indicating time
           plan, an action sequence, initially empty

  TELL(KB, MAKE-PERCEPT-SENTENCE(percept, t))
  TELL the KB the temporal “physics” sentences for time t
  safe  $\leftarrow \{[x, y] : \text{ASK}(\text{KB}, OK_{x,y}^t) = \text{true}\}$ 
  if ASK(KB, Glittert) = true then
    plan  $\leftarrow [\text{Grab}] + \text{PLAN-ROUTE}(\text{current}, \{[1,1]\}, \text{safe}) + [\text{Climb}]$ 
  if plan is empty then
    unvisited  $\leftarrow \{[x, y] : \text{ASK}(\text{KB}, I_{x,y}^{t'}) = \text{false for all } t' \leq t\}$ 
    plan  $\leftarrow \text{PLAN-ROUTE}(\text{current}, \text{unvisited} \cap \text{safe}, \text{safe})$ 
  if plan is empty and ASK(KB, HaveArrowt) = true then
    possible_wumpus  $\leftarrow \{[x, y] : \text{ASK}(\text{KB}, \neg W_{x,y}) = \text{false}\}$ 
    plan  $\leftarrow \text{PLAN-SHOT}(\text{current}, \text{possible\_wumpus}, \text{safe})$ 
  if plan is empty then // no choice but to take a risk
    not_unsafe  $\leftarrow \{[x, y] : \text{ASK}(\text{KB}, \neg OK_{x,y}^t) = \text{false}\}$ 
    plan  $\leftarrow \text{PLAN-ROUTE}(\text{current}, \text{unvisited} \cap \text{not\_unsafe}, \text{safe})$ 
  if plan is empty then
    plan  $\leftarrow \text{PLAN-ROUTE}(\text{current}, \{[1, 1]\}, \text{safe}) + [\text{Climb}]$ 
  action  $\leftarrow \text{POP}(\text{plan})$ 
  TELL(KB, MAKE-ACTION-SENTENCE(action, t))
  t  $\leftarrow t + 1$ 
  return action
```

عوامل‌های مبتنی بر منطق گزاره‌ای

○ مشکل عامل قبل:

- افزایش هزینه محاسباتی تابع ASK با گذشت زمان به دلیل افزایش تعداد نمادهای گزاره‌ای و لزوم محاسبات بیشتر در فرآیند استنتاج

○ راه حل:

- استفاده از حالت باور (belief state) به منظور دانش موجود در دنباله دریافت‌های عامل

○ تخمین حالت (state estimation)

- به فرآیند بروزرسانی حالت باور بر اساس دریافت‌های جدید عامل تخمین حالت گفته می‌شود.
- نحوه ارائه حالت باور خود کار مشکلی است. برای نمایش دقیق حالت باور برای دنیایی با n نماد زمانی، به حداقل 2^n بیت حافظه نیاز دارد.

عوامل‌های مبتنی بر منطق گزاره‌ای

○ طرح ریزی با استفاده از استنتاج گزاره‌ای:

- شرایط اولیه، اصول حالت بعدی و اعلان وضعیت رسیدن به هدف را در یک جمله قرار بده.
- جمله ایجاد شده را در اختیار یک حل کننده SAT قرار بده.
- در صورتی که جوابی یافته شود، دنباله عمل‌هایی که مقدار true دارند طرح رسیدن عامل به هدف را ارائه می‌کنند.

○ الگوریتم فوق تنها برای محیط‌های کاملاً قابل مشاهده مناسب است.

○ لازم است دو دسته اصول دیگر نیز به پایگاه دانش عامل اضافه شود:

- **اصول پیش شرط** (precondition axioms): تعیین این که اجرای هر عمل نیازمند آن است که پیش شرط‌های آن فراهم باشد.
- **اصول انحصار عمل** (action exclusion axioms): برای جلوگیری از درست فرض کردن اجرای دو عمل متفاوت در یک گام زمانی.

پایان