

بسمه تعالی

هوش مصنوعی عاملین منطقی - ۴ نیمسال اول ۱۴۰۳-۱۴۰۲

دکتر مازیار پالهنک
آزمایشگاه هوش مصنوعی
دانشکده مهندسی برق و کامپیوتر
دانشگاه صنعتی اصفهان

یادآوری

- عامل دانش - مبنا
- منطق، ایجاب کردن
- دنیای دیو، اکتشاف در دنیای دیو
- مدلهای، استنتاج
- منطق گزاره ای
- استنتاج با جدول درستی
- معتبر و قابل ارضا بودن
- قوانین استنتاج
- قانون انتزاع، حذف و، هم ارزیها
- تحلیل (resolution)
- تبدیل به شکل اصلی عطفی
- پایگاه دانش به شکل کلاوز معین

- استنتاج با کلاوزهای هرن را می توان بصورت زنجیربندی به جلو یا زنجیربندی به عقب انجام داد.
- تعیین ایجاب کردن از روی کلاوزهای هرن در زمانی بصورت خطی نسبت به اندازه پایگاه دانش قابل انجام است.

زنجیر بندی به جلو و عقب

- در شکل هر ن پایگاه دانش عطف کلاوزهای هر ن است.
- نماد گزاره ای، یا
- عطف نمادها \Leftarrow نماد
- E.g., $C \wedge (B \Rightarrow A) \wedge (C \wedge D \Rightarrow B)$
- قانون انتزاع (برای شکل هر ن) کامل است.

$$\alpha_1, \dots, \alpha_n, \quad \alpha_1 \wedge \dots \wedge \alpha_n \Rightarrow \beta$$

$$\beta$$

- می تواند با زنجیر بندی به جلو و عقب پیاده سازی شود.
- در زمان خطی با اندازه پایگاه دانش اجرا می شوند

زنجیر بندی به جلو

- هر قانونی که مؤلفه های مقدمش در KB بدست آمده آتش می شود،
- تالی آن به KB اضافه شده و اگر به جواب رسیده ایم توقف می کنیم.

$$P \Rightarrow Q$$

$$L \wedge M \Rightarrow P$$

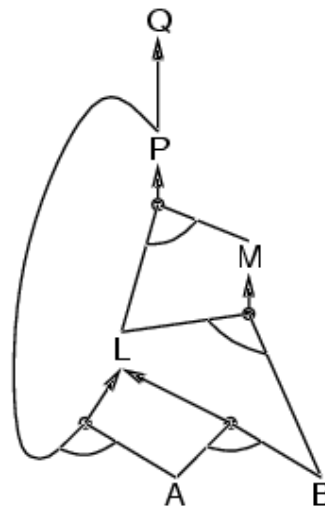
$$B \wedge L \Rightarrow M$$

$$A \wedge P \Rightarrow L$$

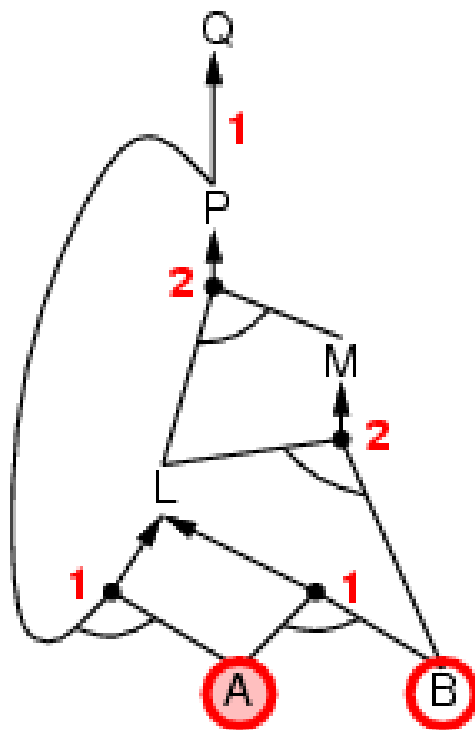
$$A \wedge B \Rightarrow L$$

A

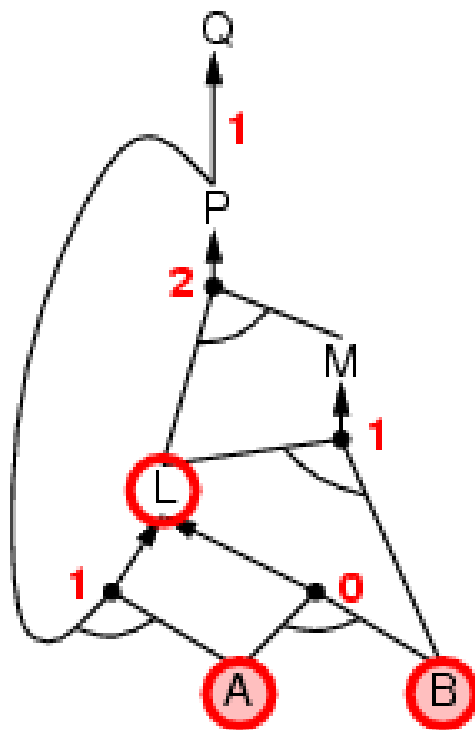
B



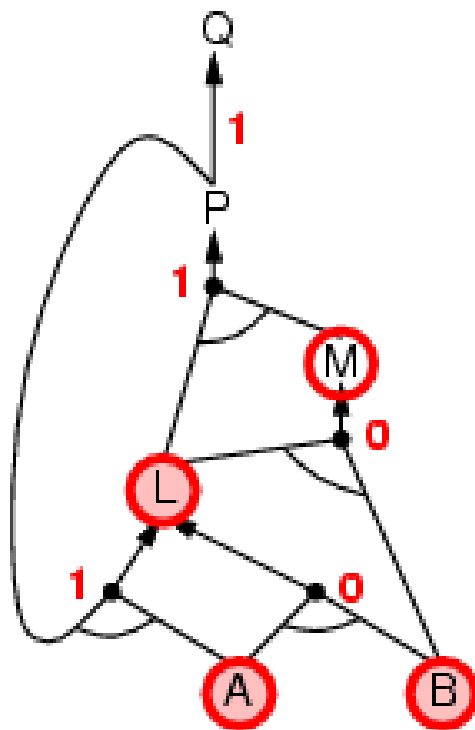
مثال زنجیر بندی به جلو



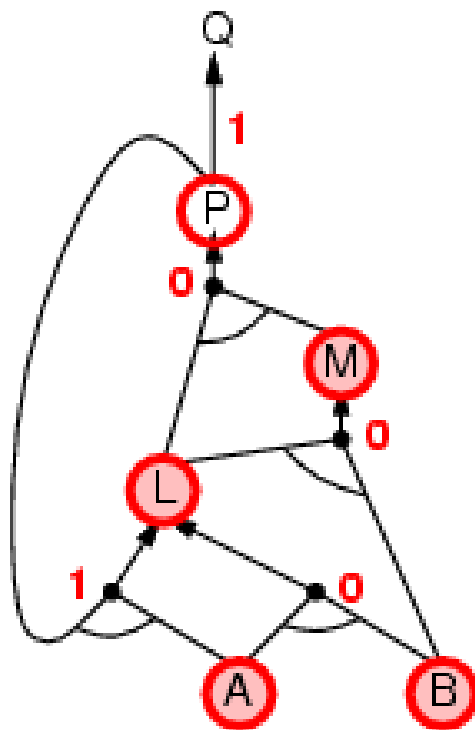
مثال زنجیر بندی به جلو



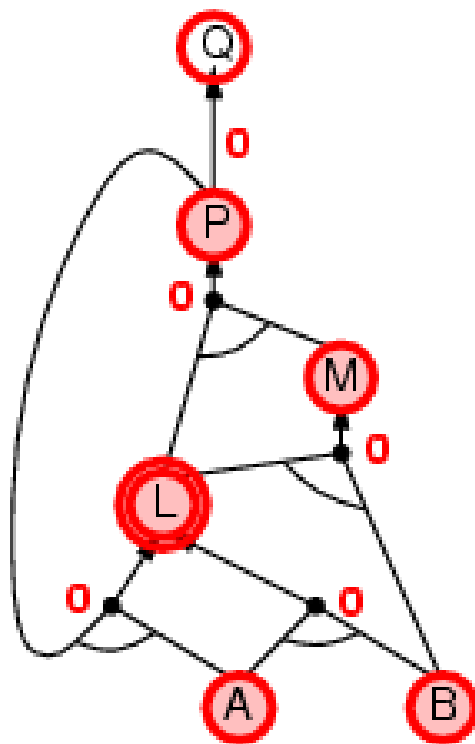
مثال زنجیر بندی به جلو



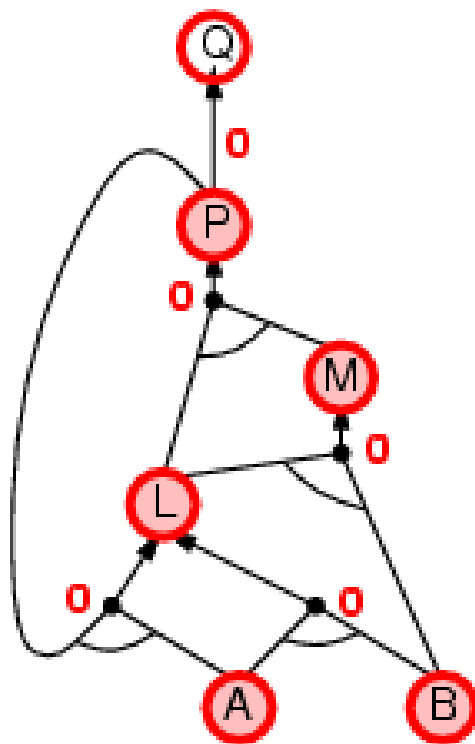
مثال زنجیر بندی به جلو



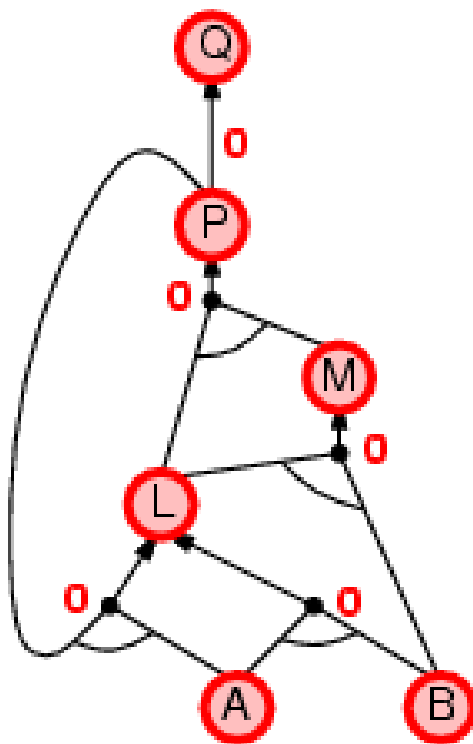
مثال زنجیر بندی به جلو



مثال زنجیر بندی به جلو



مثال زنجیر بندی به جلو



■ زنجیر بندی به جلو موثق و کامل برای کلاوزهای هرن می باشد.

اثبات کامل بودن

- ز.ج. چون اعمال قانون انتزاع است، بنابر این یک روال استنتاج موثق است.
- کامل بودن آن را باید بررسی کنیم.

اثبات کامل بودن

- ز.ج. همه اتمهائی که از KB ایجاب می شوند را تولید می کند.
- 1. ز.ج. به نقطه ثابتی می رسد که دیگر اتمی قابل بدست آمدن نیست.
(وضعیت نهائی جدول Inferred)
- 2. حالت نهائی را به عنوان مدل m که مقادیر درست/نادرست به نمادها انتساب شده در نظر بگیرید.
- 3. هر کلاوز معینی در KB اولیه در m درست است.
- 4. اگر اینگونه نباشد کلاوزی مثل $a_1 \wedge \dots \wedge a_k \Rightarrow b$ وجود دارد که نادرست است.
- 5. یعنی مقدم درست و تالی نادرست است
- 6. این در تناقض است که الگوریتم به نقطه ثابتی رسیده است.
- 7. بنابر این m یک مدل KB است.
- 8. اگر $KB \models q$ ، باید در هر مدل KB درست باشد.
- 9. بنابر این در m نیز درست است.
- 01. بنابر این هر جمله ایجاب شونده q توسط الگوریتم استنتاج می شود.

■ ز.ج. در یک عامل می تواند در هنگام ورود یک درک استفاده شود.

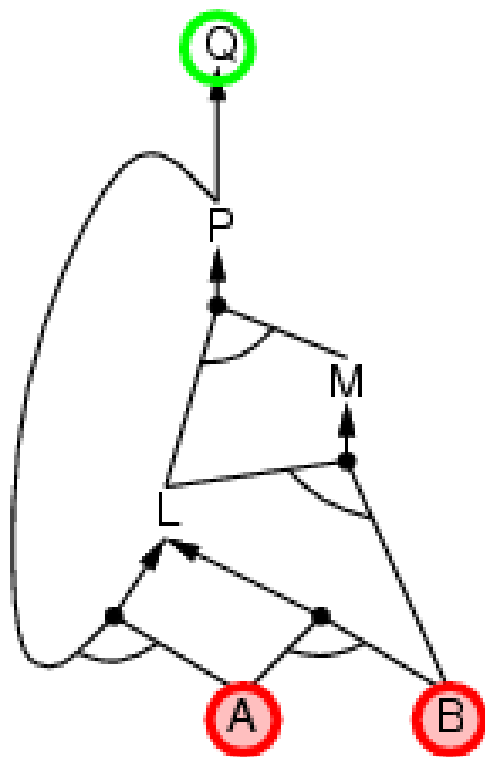
■ در Tell

■ در انسان تحت کنترل است

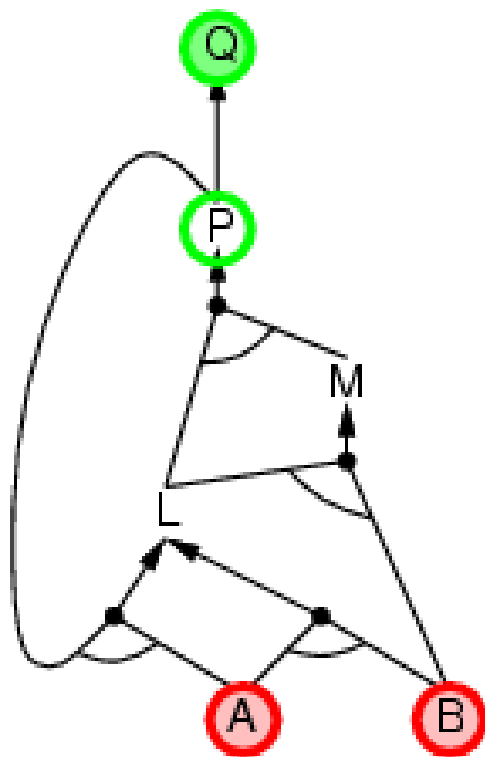
زنجیربندی به عقب

- حرکت به سوی عقب از پرسش q
- بررسی این که q هم اکنون درست است یا نه، یا
- بوسیله ز.ع. همه مؤلفه های مقدم قانونی که q را به عنوان تالی دارد بررسی می کنیم.

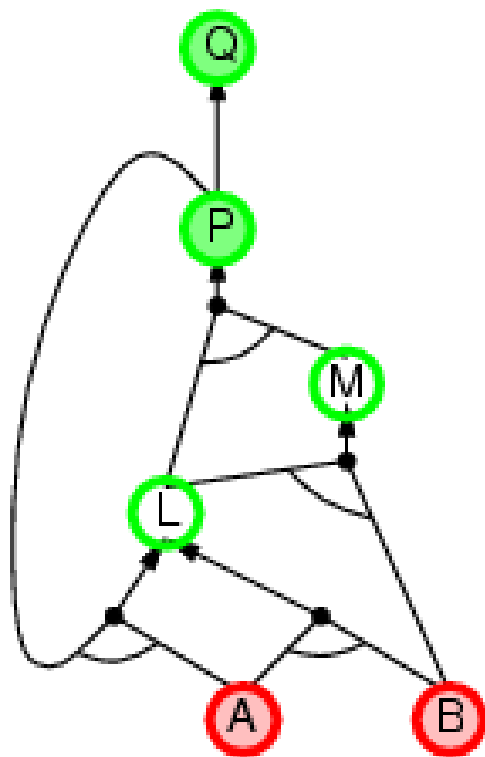
مثال زنجیربندی به عقب



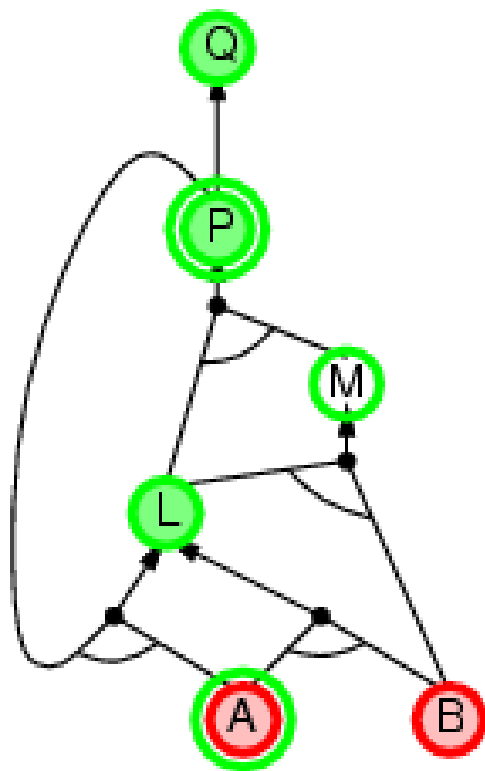
مثال زنجیربندی به عقب



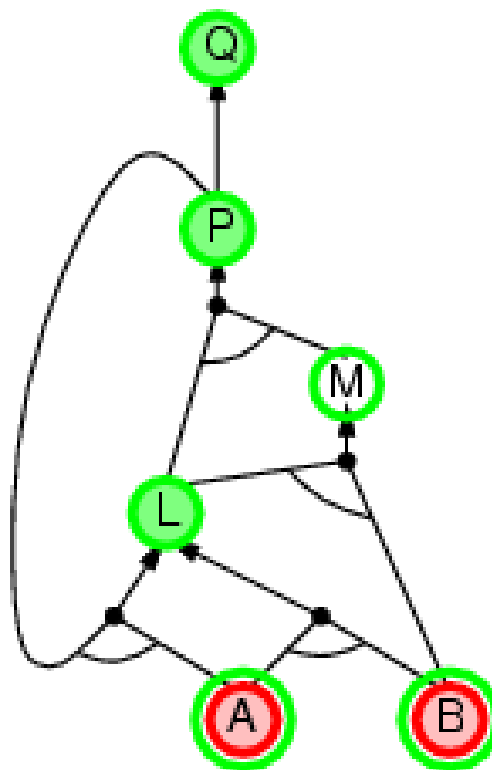
مثال زنجیربندی به عقب



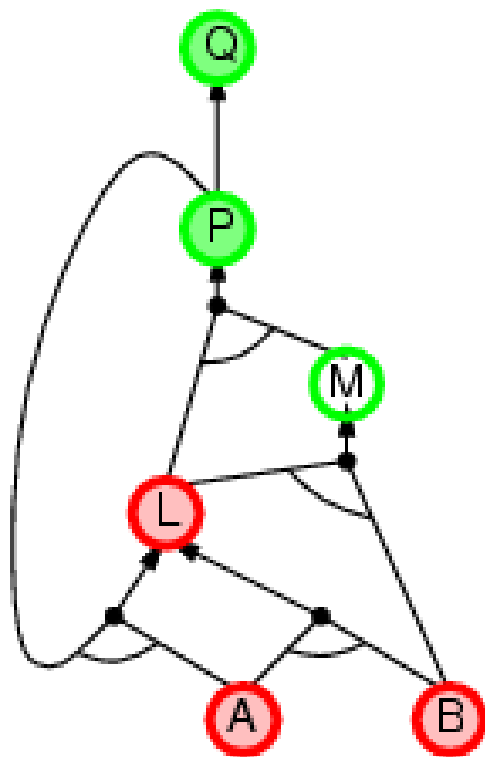
مثال زنجیربندی به عقب



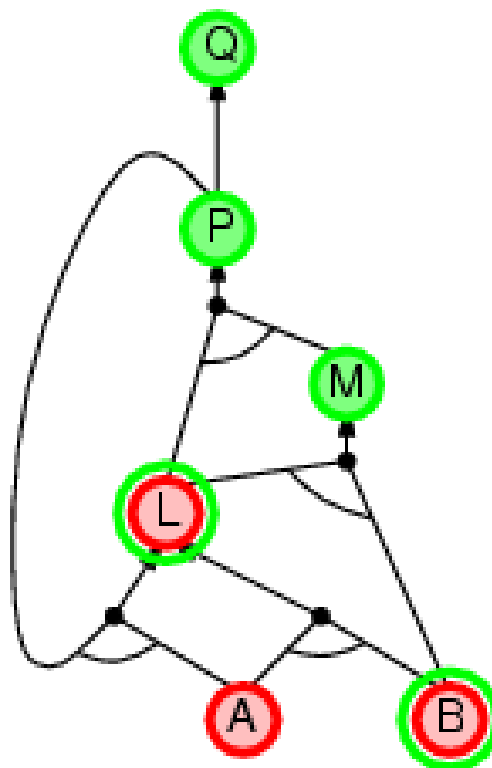
مثال زنجیربندی به عقب



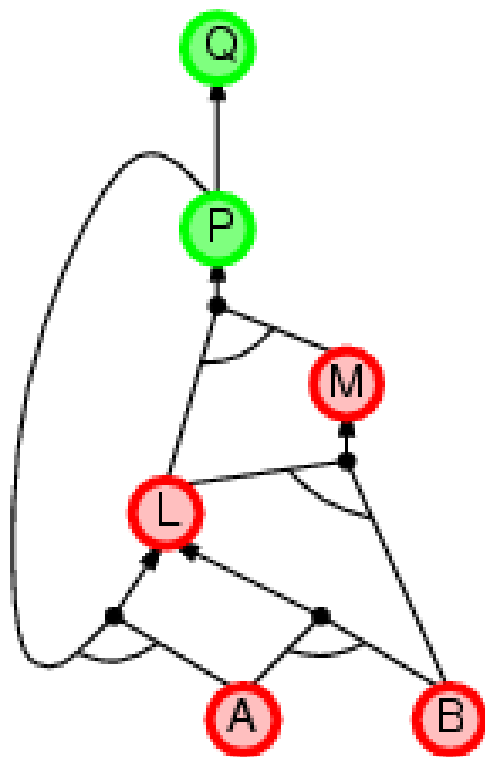
مثال زنجیربندی به عقب



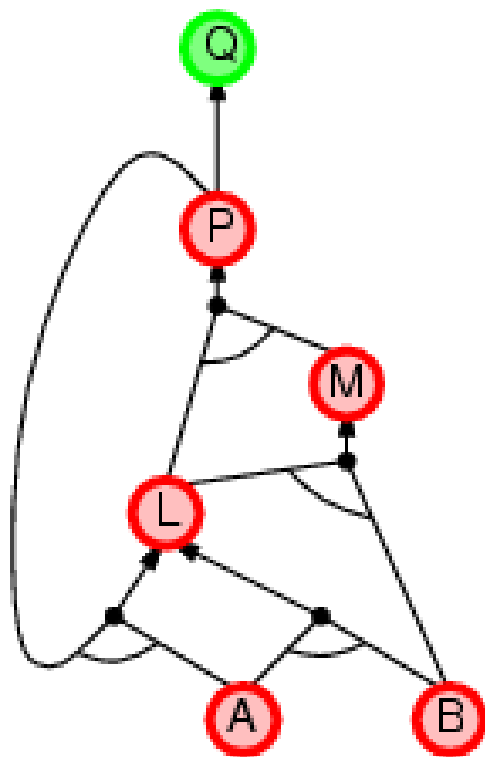
مثال زنجیربندی به عقب



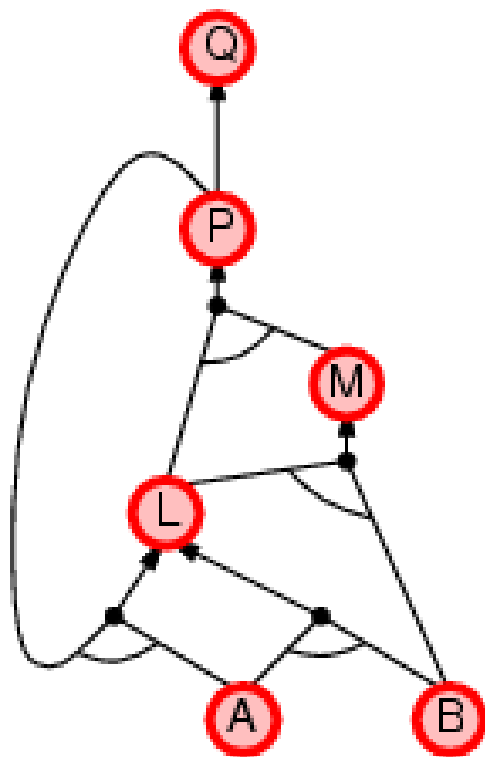
مثال زنجیربندی به عقب



مثال زنجیربندی به عقب



مثال زنجیربندی به عقب



استنتاج کارآ در منطق گزاره ای

- چک مدل در منطق گزاره ای همانند مسائل ارضاء محدودیتها
- جستجوی عقبگرد
- الگوریتم DPLL (Davis, Putnam, Logemann, Loveland)
- جستجوی محلی
- الگوریتم WalkSat

الگوریتم DPLL

- استفاده از چند مکاشفه در الگوریتم عقبگرد برای بهبود چک کامل مدلها در فهرست کردن جدول درستی
- بهبودها:

1. خاتمه زودهنگام

یک کلاوز درست است اگر لیترالی از آن درست باشد
یک جمله نادرست است اگر کلاوزی از آن نادرست باشد

2. مکاشفه نماد خالص

نماد خالص همیشه با یک علامت در کلاوزها ظاهر می شود
مثال: در $(C \vee A)$ ، $(\neg B \vee \neg C)$ ، $(A \vee \neg B)$ ، و B خالص بوده ولی C نه
لیترال متناظر با نماد خالص را درست کن
(چون اگر جمله مدلی داشته باشد با درست کردن آن نماد خالص کلاوز نادرست نمی شود.)

3. مکاشفه کلاوز تک

کلاوز تک فقط شامل یک لیترال است
آن تنها لیترال را درست کن

استفاده از جستجوی محلی

- شروع از یک انتساب به همه متغیرها،
- سعی در بهبود آن،
- با احتمال p انجام یک گام تصادفی،
- در غیر این صورت، سعی در تغییر مقدار متغیری که تعداد کلاوزهای ارضاء شده را بیشینه کند.

گزاره ها در دنیای دیو

■ دانش اولیه

$$\neg P_{1,1}$$
$$\neg W_{1,1}$$

■ و برای هر خانه

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

■ و اینکه فقط یک دیو وجود دارد:

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

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

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

■ تعداد زیادی متغیر گزاره ای

گزاره ها در دنیای دیو

- اگر حال Stench و قبلاً \neg Stench
- احتیاج به معین کردن زمان
- مثلاً Stench^4 و Stench^3 \neg
- و برای هر چیزی که با زمان تغییر کند.
- متغیرهایی که جنبه های دائم محیط را نشان داده و به زمان وابسته نیستند **متغیرهای بدون زمان** $\text{atemporal variables}$ نامیده می شوند.
- می توان نسیم و بوی بد را پس از تجربه از طریق متغیرهای زماندار به مکان آنها نسبت داد:
$$L_{x,y}^t \Rightarrow (Breeze^t \Leftrightarrow B_{x,y})$$
$$L_{x,y}^t \Rightarrow (Stench^t \Leftrightarrow S_{x,y}) .$$

گزاره ها در دنیای دیو

- برای نمایش اثر اعمال به اصول اثری effect axioms نیاز است.

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

- برای هر واحد زمانی، هر خانه، و هر یک از ۴ جهت و دیگر اعمال چنین اصولی باید نوشته شود.

- حال می توان سؤال کرد: $ASK(KB, L_{2,1}^1) = true.$

- چه اتفاقی می افتد اگر سؤال شود:

$$ASK(KB, HaveArrow^1)$$

گزاره ها در دنیای دیو

- پاسخ false خواهد بود.
- نیاز است که اعلام کنیم پس از انجام برخی از اعمال، برخی امور تغییر نمی کنند.
- نیاز به اصول قاب frame axioms

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

گزاره ها در دنیای دیو

- یا راه حل استفاده از اصول حالت تالی successors-state :axioms

$$F^{t+1} \Leftrightarrow ActionCausesF^t \vee (F^t \wedge \neg ActionCausesNotF^t)$$

- مثال:

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

- چون فرض شده دوباره تیر بدست نمی آورد.

■ اصل حالت تالی برای مکان عامل:

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

■ نیاز به دانستن امن بودن خانه ها:

$$OK_{x,y}^t \Leftrightarrow \neg P_{x,y} \wedge \neg (W_{x,y} \wedge WumpusAlive^t)$$

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

گزاره ها در دنیای دیو

■ برای وضعیت:

$\neg Stench^0 \wedge \neg Breeze^0 \wedge \neg Glitter^0 \wedge \neg Bump^0 \wedge \neg Scream^0 ; Forward^0$
 $\neg Stench^1 \wedge Breeze^1 \wedge \neg Glitter^1 \wedge \neg Bump^1 \wedge \neg Scream^1 ; TurnRight^1$
 $\neg Stench^2 \wedge Breeze^2 \wedge \neg Glitter^2 \wedge \neg Bump^2 \wedge \neg Scream^2 ; TurnRight^2$
 $\neg Stench^3 \wedge Breeze^3 \wedge \neg Glitter^3 \wedge \neg Bump^3 \wedge \neg Scream^3 ; Forward^3$
 $\neg Stench^4 \wedge \neg Breeze^4 \wedge \neg Glitter^4 \wedge \neg Bump^4 \wedge \neg Scream^4 ; TurnRight^4$
 $\neg Stench^5 \wedge \neg Breeze^5 \wedge \neg Glitter^5 \wedge \neg Bump^5 \wedge \neg Scream^5 ; Forward^5$
 $Stench^6 \wedge \neg Breeze^6 \wedge \neg Glitter^6 \wedge \neg Bump^6 \wedge \neg Scream^6$

■ حال داریم.

$ASK(KB, L_{1,2}^6) = true, \quad ASK(KB, W_{1,3}) = true \quad ASK(KB, P_{3,1}) = true,$

- زنجیربندی به جلو و عقب
- چک مدل همانند مسئله ارضاء محدودیتها
- الگوریتم عقبگرد DPLL
- الگوریتم محلی WalkSat
- گزاره ها در دنیای دیو



دانشگاه صنعتی اصفهان - مجموعه تالارها

مازیار پالهنک

هوش مصنوعی - نیمسال اول ۱۴۰۲-۰۳

39

- دقت نمائید که پاورپوینت ابزاری جهت کمک به یک ارائه شفاهی می باشد و به هیچ وجه یک جزوه درسی نیست و شما را از خواندن مراجع درس بی نیاز نمی کند.
- لذا حتماً مراجع اصلی درس را مطالعه نمائید.
- در تهیه اسلایدها از سایت کتاب استفاده شده است.