Lógica de Primeira Ordem

- necessário: regras para dizer se o mundo continuou o mesmo.
- $\forall a, x, s \; Holding(x, s) \wedge (a \neq Release) \Rightarrow Holding(x, Result(a, s))$
- $\forall a, x, s \; \neg Holding(x, s) \wedge (a \neq Grab \vee \neg(Present(x, s) \wedge Portable(x)) \Rightarrow \neg Holding(x, Result(a, s))$
- *axiomas de frame.*
- combinação de axiomas de efeito e de frame:

  verdadeiro posteriormente

  $\Leftrightarrow$ [uma ação fez ser verdadeiro $\vee$ já era verdadeiro antes]

Lógica de Primeira Ordem

- $\forall a, s, x \; Holding(x, Result(a, s)) \Leftrightarrow [(a = Grab \wedge Present(x, s) \wedge Portable(x)) \vee (Holding(x, s) \wedge a \neq Release)]$
- *axioma do estado sucessor.*
- necessário para cada predicado que pode mudar seu valor no decorrer do tempo.

Lógica de Primeira Ordem

- Localização do agente.
  - qual é a sua direção: N, S, O, L? Convenção: 0 degrees anda p/ direita ou esquerda, 90 degrees anda p/ baixo ou p/ cima. Ex: $Orientation(Agent, S_0) = 0$
  - Como as posições são mapeadas?
    * $\forall x, y \; LocationToward([x, y], 0) = [x + 1, y]$
    * $\forall x, y \; LocationToward([x, y], 90) = [x, y + 1]$
    * $\forall x, y \; LocationToward([x, y], 180) = [x - 1, y]$
    * $\forall x, y \; LocationToward([x, y], 270) = [x, y - 1]$
    * do mapa é possível dizer se um quadrado está diretamente a frente do agente:
      · $\forall p, l, s \; At(p, l, s) \Rightarrow LocationAhead(p, s) = LocationToward(l, Orientation(p, s))$
      · adjacência: $\forall l_1, l_2 \; Adj(l_1, l_2) \Leftrightarrow \exists d l_1 = LocationToward(l_2, d)$

Lógica de Primeira Ordem

- O que as ações devem fazer com as posições?
  - $\forall a, d, p, s \; At(p, l, Result(a, s)) = d \Leftrightarrow [(a = Forward \wedge l = LocationAhead(p, s) \wedge \neg Wall(l)) \vee (At(p, l, s) \wedge a \neq Forward)]$
- O que as ações devem fazer com as orientações?
  - $\forall a, d, p, s \; Orientation(p, Result(a, s)) = d \Leftrightarrow [(a = Turn(Right) \wedge d = Mod(Orientation(p, s) - 90, 360)) \vee (a = Turn(Left) \wedge d = Mod(Orientation(p, s) + 90, 360)) \vee (Orientation(p, s) = d \wedge \neg(a = Turn(Right) \vee a = Turn(Left)))]$
  - Além disso deve saber o q fazer quando está com o ouro e se o wumpus está vivo ou morto (descrição de *Shoot*).

Lógica de Primeira Ordem

- Dedução de “propriedades escondidas”.
  - $\forall l, s\ At(Agent, l, s) \wedge\ Brisa(s) \Rightarrow Fresco(l)$
  - $\forall l, s\ At(Agent, l, s) \wedge\ MauCheiro(s) \Rightarrow MauCheiroso(l)$
- Regras *sincrônicas* para relacionar propriedades de um estado ao mesmo estado.
  - *Causais* (sistemas baseados em modelos):
    * $\forall l_1, l_2, s\ At(Wumpus, l_1, s) \wedge Adj(l_1, l_2) \Rightarrow MauCheiroso(l_2)$
    * $\forall l_1, l_2, s\ At(Buraco, l_1, s) \wedge Adj(l_1, l_2) \Rightarrow Fresco(l_2)$
  - *Diagnósticas* (sistemas baseados em diagnósticos):
    * $\forall l, s\ At(Agent, l, s) \wedge\ Brisa(s) \Rightarrow Fresco(l)$
    * $\forall l, s\ At(Agent, l, s) \wedge\ MauCheiro(s) \Rightarrow MauCheiroso(l)$
    * $\forall l_1, s\ MauCheiroso(l_1) \Rightarrow (\exists l_2\ At(Wumpus, l_2, s) \wedge (l_2 = l_1 \vee Adj(l_1, l_2))$

Lógica de Primeira Ordem

- $\forall x, y, g, u, c, s\ Percept([N, N, g, u, c], t) \wedge\ At(Agent, x, s) \wedge Adj(x, y) \Rightarrow OK(y)$ (regra fraca).
- $\forall x, t(\neg\ At(Wumpus, x, t) \wedge \neg\ Buraco(x)) \Leftrightarrow OK(x)$ (regra mais forte).
- Conclusão: se os axiomas descrevem *completamente* e *corretamente* a forma como o mundo opera e a forma como as percepções são produzidas, então o procedimento de infereência vai inferir a melhor descrição possível de estados no mundo dadas as percepções.

Lógica de Primeira Ordem

- Prioridades para as ações: dada uma determinada situação, podemos escolher entre várias ações.
- $\forall a, s Great(a, s) \Rightarrow Action(a, s)$
- $\forall a, s Good(a, s) \wedge (\neg \exists b\ Great(b, s)) \Rightarrow Action(a, s)$
- $\forall a, s Medium(a, s) \wedge (\neg \exists b\ Great(b, s) \vee\ Good(b, s)) \Rightarrow Action(a, s)$
- $\forall a, s Risky(a, s) \wedge (\neg \exists b\ Great(b, s) \vee\ Good(b, s) \vee Medium(b, s)) \Rightarrow Action(a, s)$
- sistemas que usam este tipo de regra: *action-value*.

Lógica de Primeira Ordem

- Descrições anteriores não dizem nada sobre as ações. Apenas classificam as ações.
- ações do tipo Great consistem em pegar o ouro quando este for encontrado e pular para fora da caverna com o ouro.
- ações do tipo Good consistem em mover para uma posição que é OK, mas que não foi ainda visitada.
- ações do tipo Medium consistem em mover para uma posição que está OK, mas que já foi visitada.
- ações do tipo Risky consistem em mover para uma posição em que não se conhece a situação.

Lógica de Primeira Ordem

- Agente baseado em Objetivos: tenta alcançar os objetivos.
- uma vez que o ouro foi encontrado, a política de movimentação na caverna muda radicalmente: devemos voltar à posição inicial o mais rápido possível.
- $\forall s\ Holding(Ouro, s) \Rightarrow GoalLocation([1,1], s)$
- três formas de encontrar uma sequência de passos que levem a algum objetivo:
  - Inferência
  - Busca (pe, best-first search)
  - Planning: sistemas de raciocínio orientados para ações.

Cap. 9: Inferência em Lógica de Primeira Ordem

- Mesmas regras de cálculo de predicados proposicional podem ser aplicadas.
- Precisamos de mais regras para tratamento de variáveis.
- Substituição de variáveis por constantes individuais: $SUBST(\theta, \alpha)$.
- Ex:
  $SUBST(\{x/Sam, y/Pam\}, Likes(x, y)) = Likes(Sam, Pam)$
- três novas regras:

  Elimin Universal $\dfrac{\forall v \alpha}{SUBST(\{v/g\}, \alpha)}$ Elimin Existencial $\dfrac{\exists v \alpha}{SUBST(\{v/k\}, \alpha)}$

  Introd Existencial $\dfrac{\alpha}{\exists v SUBST(\{g/v\}, \alpha)}$
- Importante: eliminação existencial deve fazer substituições por constantes que ainda **não** tenham aparecido no KB!

Cap. 9: Inferência em Lógica de Primeira Ordem

- Exemplo de prova: "A lei americana diz que é crime um americano vender armas para nações hostis. Nono, um país inimigo dos EUA, tem alguns mísseis, e todos estes mísseis foram vendidos pelo Coronel Oeste, que é americano". Provar que o coronel é criminoso.
- ...é um crime um americano vender armas para nações hostis...
- (1) $\forall x, y, z Amer(x) \wedge Arma(y) \wedge Nacao(z) \wedge Hostil(z) \wedge Vende(x, z, y) \Rightarrow Crim(x)$
- ...Nono...tem alguns mísseis...
- (2) $\exists x Dono(Nono, x) \wedge Missil(x)$

Cap. 9: Inferência em Lógica de Primeira Ordem

- ...todos estes mísseis foram vendidos pelo Coronel Oeste...
- (3) $\forall x Dono(Nono, x) \wedge Missil(x) \Rightarrow Vende(Oeste, Nono, x)$
- (4) $\forall x Missil(x) \Rightarrow Arma(x)$
- (5) $\forall x Inimigo(x, EUA) \Rightarrow Hostil(x)$
- (6) $Americano(Oeste)$, (7) $Nacao(Nono)$, (8) $Inimigo(Nono, EUA)$, (9) $Nacao(EUA)$

Cap. 9: Inferência em Lógica de Primeira Ordem

- (10) De (2) e elim exist: $Dono(Nono, M1) \wedge Missil(M1)$
- (11) e (12) De (10) e elim E: $Dono(Nono, M1), Missil(M1)$
- (13) De (4) e elim univ: $Missil(M1) \Rightarrow Arma(M1)$
- (14) De (12), (13) e Modus Ponens: $Arma(M1)$
- (15) De (3) e elim univ:
  $Dono(Nono, M1) \wedge Missil(M1) \Rightarrow Vende(Oeste, Nono, M1)$
- (16) De (15), (10) e Modus Ponens: $Vende(Oeste, Nono, M1)$

Cap. 9: Inferência em Lógica de Primeira Ordem

- (17) De (1) e elim univ (3x):
  $Amer(Oeste) \wedge Arma(M1) \wedge Nacao(Nono) \wedge Hostil(Nono) \wedge Vende(Oeste, Nono, M1) \Rightarrow Crim(Oeste)$
- (18) De (5) e elim univ:
  $Inimigo(Nono, EUA) \Rightarrow Hostil(Nono)$
- (19) De (8) e (18) e Modus Ponens: $Hostil(Nono)$
- (20) De (6), (7), (14), (16), (19) e introd E:
  $Amer(Oeste) \wedge Arma(M1) \wedge Nacao(Nono) \wedge Hostil(Nono) \wedge Vende(Oeste, Nono, M1)$
- (21) De (17), (20) e Modus Ponens: $Crim(Oeste)$

Cap. 9: Inferência em Lógica de Primeira Ordem

- Observações:
  - Esta prova é a solução para um problema de busca, se formulássemos este problema como um problema de busca.
  - O algoritmo deveria ser bastante esperto para não seguir caminhos errados.
- Formulado como um problema de busca:
  - estado inicial: KB, sentenças de 1 a 9.
  - operadores: regras de inferência.
  - estado final: KB contendo $Crim(Oeste)$.

Cap. 9: Inferência em Lógica de Primeira Ordem

- 14 passos de prova.
- fator de ramificação aumenta de acordo com o aumento do KB.
- Elim univ pode ter um fator de ramificação muito grande, porque podemos substituir as variáveis por qualquer termo "ground".
- Tempo gasto em conjunções, instanciação de variáveis e aplicação de Modus Ponens.
- problema com Modus Ponens é que somente faz deduções sobre termos "ground".
- Métodos mais eficientes de prova.

Cap. 9: Inferência em Lógica de Primeira Ordem

- Modus Ponens Generalizado.
- um único passo para intr E, elim univ e Modus Ponens.
- Idéia: inferir de KB = $Missil(M1)$, $Dono(Nono, M1)$, $\forall x Missil(x) \wedge Dono(Nono, x) \Rightarrow Vende(Oeste, Nono, x)$, num único passo: $Vende(Oeste, Nono, M1)$
- se existir uma substituição θ tal que $SUBST(\theta, p_i^{'}) = SUBST(\theta, p_i)$
- $\frac{p_1^{'}, p_2^{'}, \ldots, p_n^{'}, (p_1 \wedge p_2 \wedge \ldots \wedge p_n \Rightarrow q)}{SUBST(\theta, q)}$

  $p_1^{'}$ is $Missil(M1)$ p_1 is $Missil(x)$

  $p_2^{'}$ is $Dono(y, M1)$ p_2 is $Dono(Nono, x)$

  θ is {x/M1,y/Nono} q is $Vende(Oeste, Nono, x)$

  $SUBST(\theta, q)$ is $Vende(Oeste, Nono, M1)$

Cap. 9: Inferência em Lógica de Primeira Ordem

- Modus Ponens Generalizado é uma regra de inferência eficiente por três razões:
  - usa “passos largos”, combinando várias inferências pequenas em apenas uma.
  - usa passos coerentes, usa substituições que são garantidamente úteis, invés de aplicar elim universal de forma aleatória.
  - usa um passo de pre´compilação para converter todas as sentenças do KB em *forma canônica* (forma da regra obriga a isto).

Cap. 9: Inferência em Lógica de Primeira Ordem

- *Forma canônica*: todas as fórmulas no KB são fórmulas atômicas ou uma implicação com uma conjunção de fórmulas atômicas como antecedente e um único átomo/literal como consequente $\rightarrow$ *sentenças de Horn*.
- um KB contendo somente sentenças de Horn: Forma Normal de Horn.
- sentenças são transformadas em sentenças de Horn qdo dão entrada no KB. Ex: $\exists x Dono(Nono, x) \wedge Missil(x)$ é transformada em duas novas sentenças através de elim exist e elim E.
- uma vez que todos os quantif exist são eliminados, podemos dispor do quantif univ. Assume-se que todas as variáveis estão quantif universalmente.
- Nem toda sentença pode ser convertida em sentença de Horn.

Cap. 9: Inferência em Lógica de Primeira Ordem

- *Unificação*: pega duas sentenças atômicas e retorna uma substituição que faz as duas sentenças parecerem a mesma.
- $UNIFY(p, q) = \theta$, com $SUBST(\theta, p) = SUBST(\theta, q)$.
- θ: unificador das duas sentenças.
- Ex: $Conhece(Joao, x) \Rightarrow Odeia(Joao, x)$
- KB: $Conhece(Joao, Jane)$, $Conhece(y, Leonardo)$, $Conhece(y, Mae(y))$, $Conhece(x, Elisabete)$

Cap. 9: Inferência em Lógica de Primeira Ordem

- Aplicando Unificação:
- $UNIFY(Conhece(Joao, x), Conhece(Joao, Jane)) = \{x/Jane\}$
- $UNIFY(Conhece(Joao, x), Conhece(y, Leonardo)) = \{x/Leonardo, y/Joao\}$
- $UNIFY(Conhece(Joao, x), Conhece(y, Mae(y))) = \{y/Joao, x/Mae(Joao)\}$
- $UNIFY(Conhece(Joao, x), Conhece(x, Elisabete)) = falha!$