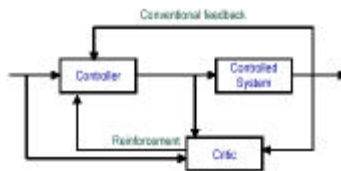


Aprendizagem em Robótica

Aprendizagem em controlo robótico: Aprendizagem pelo Reforço
Numérica, Indutiva, Contínua, não-supervisada

Base: “aplicando uma recompensa imediatamente a seguir à ocorrência de uma Resposta, aumenta a sua probabilidade de tornar a ocorrer, enquanto que se aplicar uma punição a seguir à resposta, decresce essa probabilidade”

Como se calculam os valores de recompensa/punição aplicados pelo Crítico?



2003 / LEIC

Eugénio Oliveira/FEUP

Robótica

Aprendizagem em Robótica

Aprendizagem Q:

α taxa de aprendizagem

r recompensa/punição

γ factor de desconto ($0 < \gamma < 1$)

$E(y)$ utilidade do estado y

resultante da acção.

$E(y) = \max_a (Q(y,a))$ para todas as acções a

$$Q(x,a) \leftarrow Q(x,a) + \alpha(r + \gamma E(y) - Q(x,a))$$

Propagar as recompensas através dos estados de forma a que estados semelhantes também aprendam.

Estados semelhantes através da “Distância pesada de Hamming”

2003 / LEIC

Eugénio Oliveira/FEUP

Robótica

Aprendizagem em Robótica

Modificação das respostas dos robôs usando Aprendizagem Q

Inicialize todos os $Q(x,a)$ a 0

Fazer sempre

Determine estado corrente s pelos sensores

Em 90% dos casos escolha a acção a que maximiza $Q(x,a)$

Nos outros casos escolha uma acção aleatória

Execute a

Determine recompensa r

Modifique $Q(x,a)$ de acordo com a fórmula

Modifique $Q(x',a)$ para todos estados x' semelhantes a x

Fim Fazer

Aprendizagem em Robótica

Exemplo de uso de Aprendizagem Q: Ensinar a empurrar caixas:

Robô:

8 sensores sonar (4 frente, 2 direita, 2 esquerda):

distinguem "perto" (entre 20 e 35 cm) e "longe"

sensor infra-vermelho à frente:

resposta binária de presença de obstáculo a cerca de 5cm

sensor de corrente do motor:

se superior a um limite o robô está bloqueado

Informação sensorial em vector de 18 bits (16 dist, bater, preso)

Comandos para o Motor:

mover para-frente

virar esquerda 22 graus

virar direita 22 graus

virar esquerda 45 graus

virar direita 45 graus

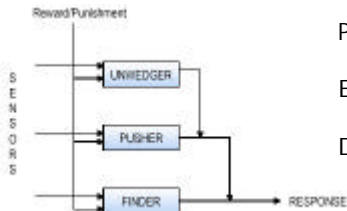
Aprendizagem em Robótica

Exemplo de uso de Aprendizagem Q: Ensinar a empurrar caixas:

Problema a aprender:

para cada um dos cerca de 250.000 estados perceptuais
quais as acções que melhor permitem encontrar e empurrar caixas numa
sala sem ficar preso.

Controlador dos Comportamentos:



Procurar: move o robô para próximo das caixas

Empurrar: Aplicado depois de encontrada caixa (I V)

Desistir: Retira o robô quando a caixa não é mais deslocável

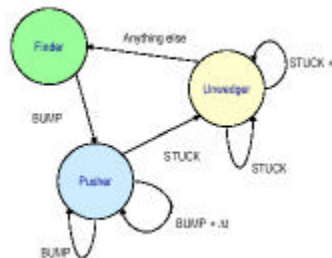
2003 / LEIC

Eugénio Oliveira/FEUP

Robótica

Aprendizagem em Robótica

Diagrama de Transições entre Comportamentos



Recompensas/Punições:

Procurar: recompensa +1 quando o movimento é "para frente" e vector das percepções inclui bits de "perto"
punição -1 quando bits "perto" se alteram para "longe"

Empurrar: recompensa +1 se comando "para frente" e continua sensor IV activo
punição -3 se deixa de estar activo sensor IV

Desistir: Recompensa +1 Se corrente excede limite
punição -3 se não continua no estado "preso"

2003 / LEIC

Eugénio Oliveira/FEUP

Robótica

Aprendizagem em Robótica

Distância de Hamming:
Número de bits em que dois estados diferem

Por exemplo:
Estados podem ser considerados semelhantes se $DH < 3$

Robô Obelix melhora a sua capacidade de empurrar caixas

Aprendizagem em Robótica

Aprendizagem de Coordenação de Comportamentos usando:
Aprendizagem pelo Reforço "formado" (modelado?) (Maja Mataric)

Espaço de estados com:
comportamentos e condições
Funções de recompensa heterogéneas e estimadores de progresso

"formado" pelo domínio pois conhecimento sobre o domínio permitirá
o feedback intermitente em vez de sinal contínuo de erro.

Dois tipos:

Funções de recompensa heterogéneas
(dependendo das percepções e do estado interno)
Estimadores de progresso
(metrica de avaliação durante a execução do comportamento)

Aprendizagem em Robótica

Recolha de esferas por robôs com aprendizagem



2003 / LEIC

Eugénio Oliveira/FEUP

Robótica

Aprendizagem em Robótica

Recolha de esferas por robôs com aprendizagem

Comportamentos:

vaguear
dispersar
parar
recolher ao silo

Condições

tem esfera?
no-silo?
próximo-de-intruso?
tempo de repouso?

Reforço R:

Para cada par condição c -comportamento b , no instante t :

Dimensão: $2^4 * 4 = 64$

Condições

comportamentos

$$A(c, b) = \sum_{t=1}^T R(c, t)$$

2003 / LEIC

Eugénio Oliveira/FEUP

Robótica

Aprendizagem em Robótica

Funções de Reforço heterogéneas:

Recompensas (reforço positivo imediato)

E_p : esfera apanhada

E_{gd} : largar esfera no silo

E_{gw} : sair do silo

Punição (reforço negativo imediato)

E_{bd} : largar esfera fora do silo

E_{bw} : longe do silo

$$R_E(c) = \begin{cases} p & \text{se } E_p > 0 \\ gd & \text{se } E_{gd} > 0 \\ bd & \text{se } E_{bd} < 0 \\ gw & \text{se } E_{gw} > 0 \\ bw & \text{se } E_{bw} < 0 \\ 0 & \text{nos outros casos} \end{cases}$$

Para $p, gd, gw > 0$,
 $bd, bw < 0$

Aprendizagem em Robótica

Estimadores de progresso:

Permitem acções que não levam a uma recompensa imediata mas mais tardia

Função de progresso no evitar intruso:

$$R_I = R_I(c, t) = \begin{cases} i & \text{distância ao intruso aumenta} \\ d & \text{nos outros casos} \end{cases}$$

Função de progresso de regresso ao silo:

$$R_S = R_S(c, t) = \begin{cases} pr & \text{próximo do silo} \\ lo & \text{longe do silo} \\ 0 & \text{nos outros casos} \end{cases}$$

Cálculo do Reforço total: $R(c, t) = uR_E(c, t) + vR_I(c, t) + wR_S(c, t)$

$$u, v, w \geq 0 \text{ e } (u+v+w) = 1$$