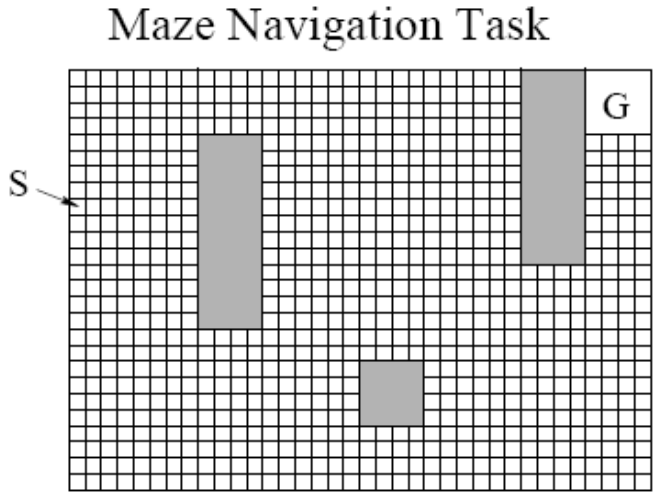
Model Based Learning in continuous state spaces

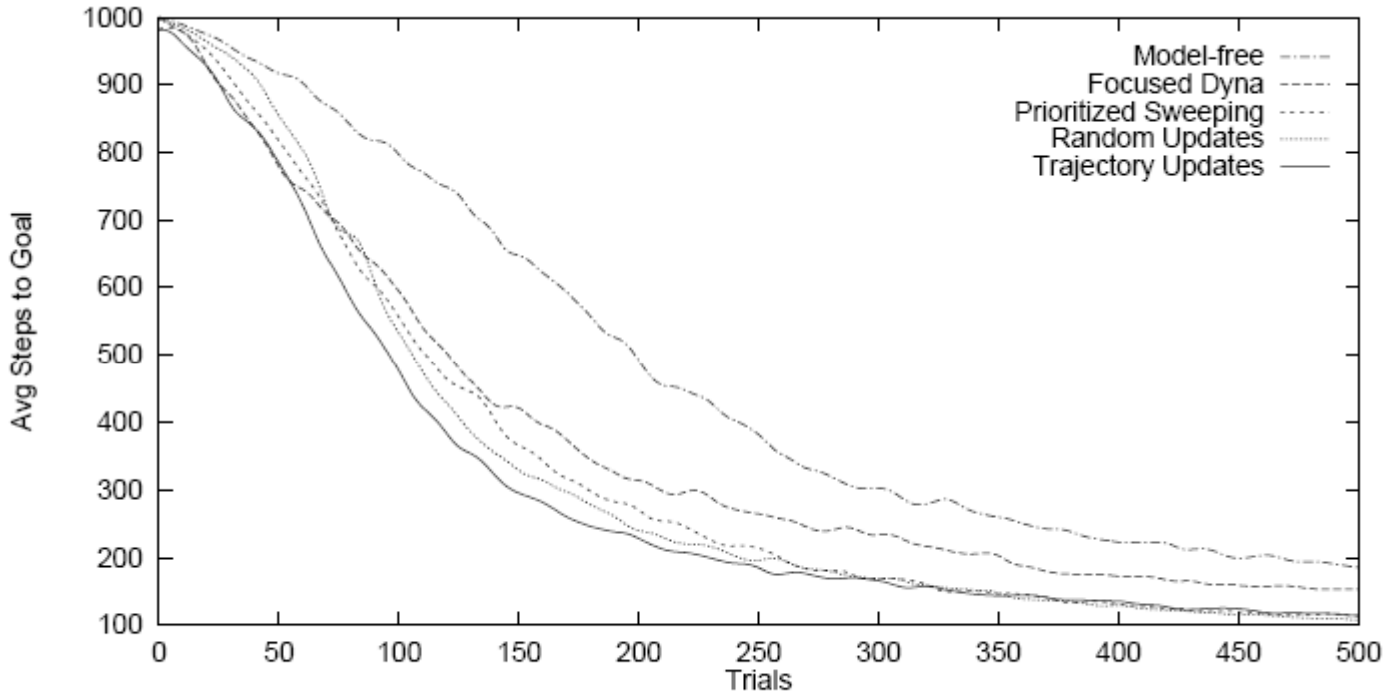
Doran Chakraborty

A comparative study

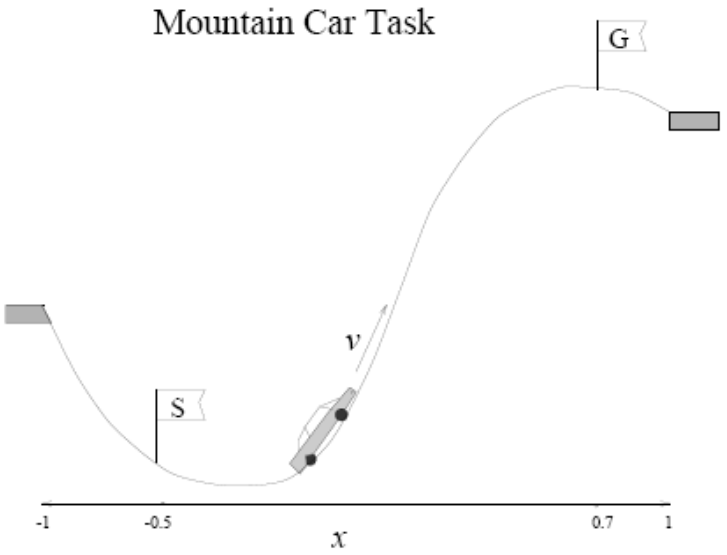
- Random Updates
- Prioritized sweeping
- Trajectory sampling



empirical results



Mountain car problem



$$-1 \leq x_t \leq 1$$

$$-2 \leq v_t \leq 2.$$

$$q_t = \begin{cases} 2 \cdot x + 1 & \text{if } x < 0 \\ \frac{1}{(1+5x_t^2)^{3/2}} & \text{if } x \geq 0 \end{cases}$$

$$a_t = \frac{f_t}{m \cdot \sqrt{1+q_t^2}} - \frac{g \cdot q_t}{1+q_t^2}$$

$$x_{t+1} = x_t + v_t \cdot \Delta t + \frac{a \cdot \Delta t^2}{2}$$

$$v_{t+1} = v_t + a \cdot \Delta t$$

Model Based RL in continuous state spaces

1. Initially: $w(t) := 0$, $\forall t \in Tiles$; $s_{sim} = s_0$; $a_{sim} = policy(s_{sim})$
2. Start of Trial: $s := s_0$, $a := policy(s)$
3. Take action a; observe reward r and next state s'
4. $a' := policy(s)$
5. Learn:
   $\epsilon := r + \sum_{t' \in Tiles(s',a')} w(t') \quad - \quad \sum_{t \in Tiles(s,a)} w(t)$
   $w(t) := w(t) + \frac{\alpha}{L} \cdot \epsilon, \quad \forall t \in Tiles(s,a)$
6. Update Model: Add a new observation s' to a list of past observations kept in the hash table entry $m(s,a)$. If s' is already in the table then increment the number of times s' has been observed by 1
7. Sample Model:
   Repeat K times
   take action a_{sim};
   use model to compute the predicted next state, s'_{sim}, and reward, r';
   if s'_{sim} is the terminal state
   set $s_{sim} := s_0, a_{sim} := policy(s_{sim})$
   go to the beginning of the repeat loop

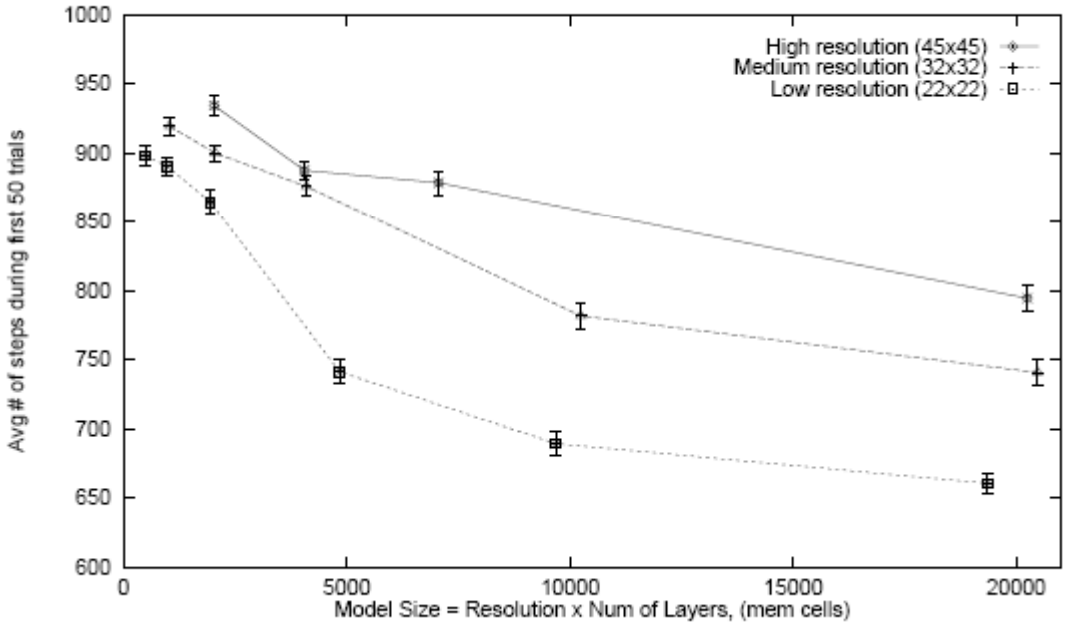
   $a'_{sim} := policy(s_{sim})$;
   learn: $\epsilon := r + \sum_{t' \in Tiles(s'_{sim}, a'_{sim})} w(t') \quad - \quad \sum_{t \in Tiles(s_{sim}, a_{sim})} w(t)$
   $w(t) := w(t) + \frac{\alpha}{L} \cdot \epsilon, \forall t \in Tiles(s_{sim}, a_{sim})$
8. Loop: $a := a'$; $s := s'$; if s' is the terminal state, go to 2, else go to 3

Input Representation

- Need to represent the T function
- f(x,v) -> x f(x,v)-> v for all actions
- CMACS used to learn the function
- Model Size = Resolution X Num of Layers

What makes a good model?

Choosing a model



Model Early method