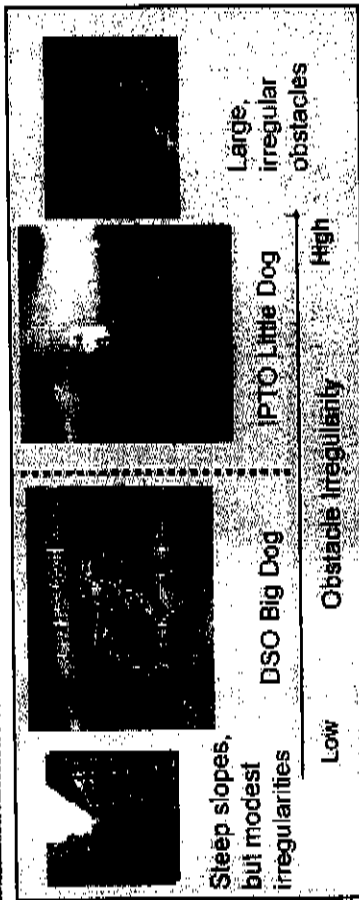


Little Dog Overview
 [LARRY JACKEL, DARPA ITO, LEARNING LOCOMOTION]
 LINK ON WEDPAGE

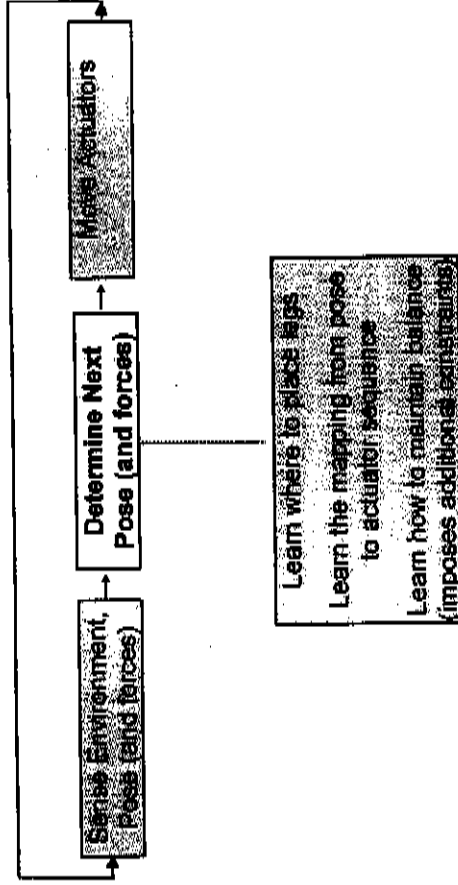
Goal: Learn how to traverse large, irregular obstacles with a high DOF robot



- Rough, irregular terrain
 - Too complex for "high-compliance", reactive algorithms
 - Requires deliberately planned, coordinated movements
 - Too tedious for handcrafting – automatic learning is needed

- Initial assumption: Perfect perception of environment

Locomotion Process



From: Sutton + Barto, Reinforcement Learning
 LINK TO ONLINE VERSION ON WEAPRE

The second approach to avoiding the infinite number of episodes nominally required for policy evaluation is to forgo trying to complete policy evaluation before returning to policy improvement. On each evaluation step we move the value function toward Q^{π} , but we do not expect to actually get close except over many steps. We used this idea when we first introduced the idea of GPI in Section 4.6. One extreme form of the idea is value iteration, in which only one iteration of iterative policy evaluation is performed between each step of policy improvement. The in-place version of value iteration is even more extreme; there we alternate between improvement and evaluation steps for single states.

For Monte Carlo policy evaluation it is natural to alternate between evaluation and improvement on an episode-by-episode basis. After each episode, the observed returns are used for policy evaluation, and then the policy is improved at all the states visited in the episode. A complete simple algorithm along these lines is given in Figure 5.4. We call this algorithm *Monte Carlo ES*, for Monte Carlo with Exploring Starts.

In Monte Carlo ES, all the returns for each state-action pair are accumulated and averaged, irrespective of what policy was in force when they were observed. It is easy to see that Monte Carlo ES cannot converge to any suboptimal policy. If it did, then the value function would eventually converge to the value function for that policy, and that in turn would cause the policy to change. Stability is achieved only

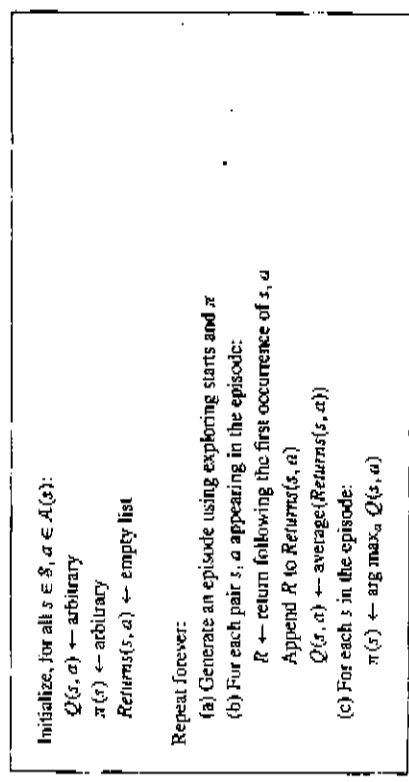


Figure 5.4 Monte Carlo ES: A Monte Carlo control algorithm assuming exploring starts.

5.3 Monte Carlo Control

when both the policy and the value function are optimal. Convergence to this optimal fixed point seems inevitable as the changes to the action-value function decrease over time, but has not yet been formally proved. In our opinion, this is one of the most fundamental open questions in reinforcement learning.

Example 5.3: Solving Blackjack It is straightforward to apply Monte Carlo ES to blackjack. Since the episodes are all simulated games, it is easy to arrange for exploring starts that include all possibilities. In this case one simply picks the dealer's cards, the player's sum, and whether or not the player has a usable ace, all at random with equal probability. As the initial policy we use the policy evaluated in the previous blackjack example, that which sticks only on 20 or 21. The initial action-value function can be zero for all state-action pairs. Figure 5.5 shows the optimal policy for blackjack found by Monte Carlo ES. This policy is the same as the "basic" strategy of Thorp (1966) with the sole exception of the leftmost notch in the policy for a usable ace, which is not present in Thorp's strategy. We are uncertain of the reason for this discrepancy, but confident that what is shown here is indeed the optimal policy for the version of blackjack we have described.

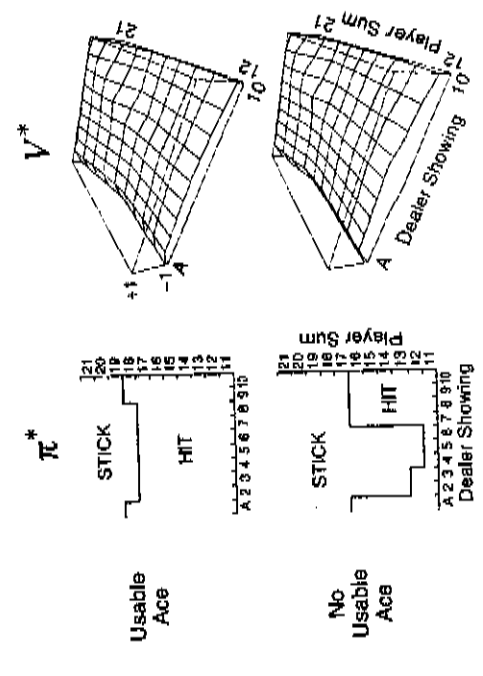


Figure 5.5 The optimal policy and state-value function for blackjack, found by Monte Carlo ES (Figure 5.4). The state-value function shown was computed from the action-value function found by Monte Carlo ES.