6.231 DYNAMIC PROGRAMMING

LECTURE 16

LECTURE OUTLINE

- More on rollout algorithms
- Simulation-based methods
- Approximations of rollout algorithms
- Rolling horizon approximations
- Discretization issues
- Other suboptimal approaches

ROLLOUT ALGORITHMS

- *Rollout policy*: At each k and state x_k, use the control $\overline{\mu}_k(x_k)$ that

$$\min_{u_k \in U_k(x_k)} Q_k(x_k, u_k),$$

where

$$Q_k(x_k, u_k) = E\Big\{ g_k(x_k, u_k, w_k) + H_{k+1}\Big(f_k(x_k, u_k, w_k)\Big)\Big\}$$

and $H_{k+1}(x_{k+1})$ is the cost-to-go of the heuristic.

- $Q_k(x_k, u_k)$ is called the $Q\text{-}factor$ of (x_k, u_k), and for a stochastic problem, its computation may involve Monte Carlo simulation.

- Potential difficulty: To minimize over u_k the Q-factor, we must form Q-factor differences $Q_k(x_k, u) - Q_k(x_k, \overline{u})$. This differencing often amplifies the simulation error in the calculation of the Q-factors.

- Potential remedy: Compare any two controls u and $\overline{u}$ by simulating $Q_k(x_k, u) - Q_k(x_k, \overline{u})$ directly.

Q-FACTOR APPROXIMATION

- Here, instead of simulating the Q-factors, we approximate the costs-to-go $H_{k+1}(x_{k+1})$.
- Certainty equivalence approach: Given x_k, fix future disturbances at "typical" values $\overline{w}_{k+1}, \ldots, \overline{w}_{N-1}$ and approximate the Q-factors with

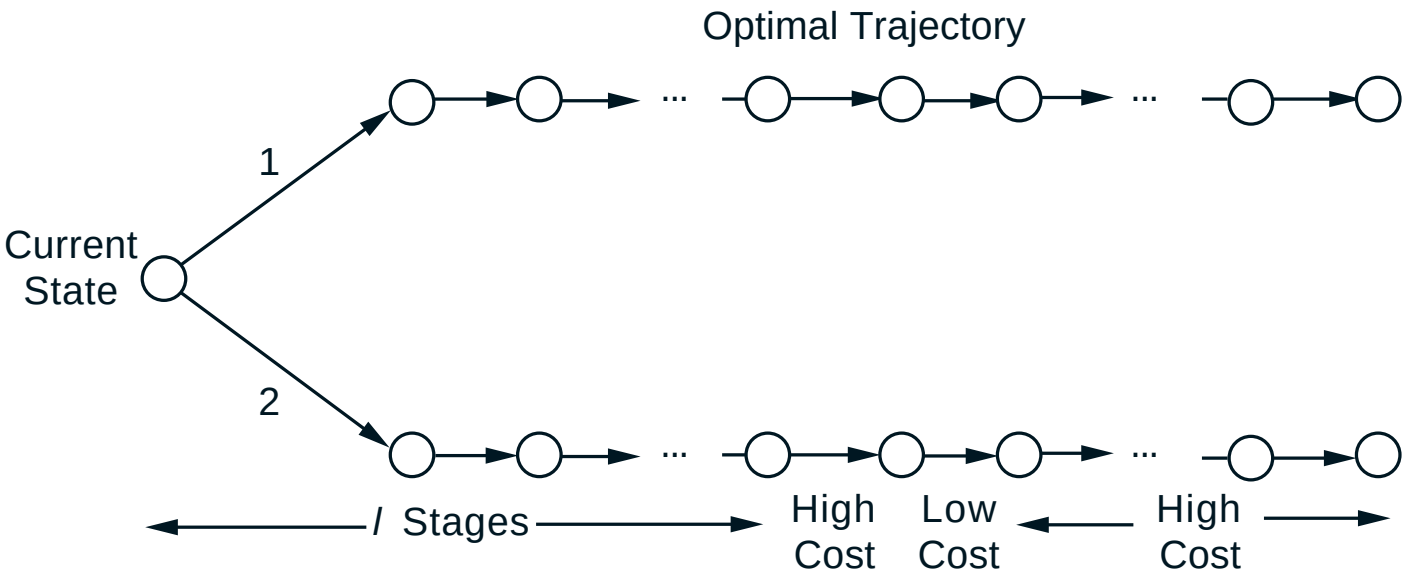
$$\tilde{Q}_k(x_k, u_k) = E\Big\{ g_k(x_k, u_k, w_k) + \tilde{H}_{k+1}\Big(f_k(x_k, u_k, w_k)\Big)\Big\}$$

where $\tilde{H}_{k+1}\Big(f_k(x_k, u_k, w_k)\Big)$ is the cost of the heuristic with the disturbances fixed at the typical values.

- This is an approximation of $H_{k+1}\Big(f_k(x_k, u_k, w_k)\Big)$ by using a "single sample simulation."
- Variant of the certainty equivalence approach: Approximate $H_{k+1}\Big(f_k(x_k, u_k, w_k)\Big)$ by simulation using a small number of "representative samples" (scenarios).
- Alternative: Calculate (exact or approximate) values for the cost-to-go of the base policy at a limited set of state-time pairs, and then approximate H_{k+1} using an approximation architecture and a "least-squares fit."

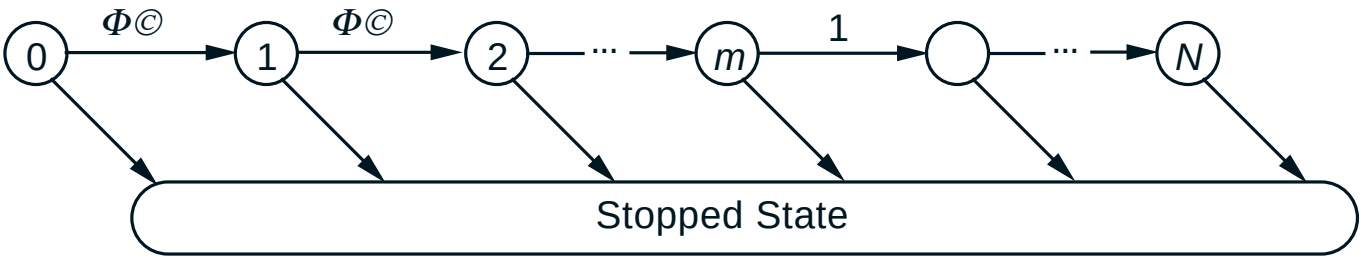
ROLLING HORIZON APPROACH

- This is an l-step lookahead policy where the cost-to-go approximation is just 0.
- Alternatively, the cost-to-go approximation is the terminal cost function g_N.
- A short rolling horizon saves computation.
- "Paradox": It is not true that a longer rolling horizon always improves performance.
- Example: At the initial state, there are two controls available (1 and 2). At every other state, there is only one control.



ROLLING HORIZON COMBINED WITH ROLLOUT

- We can use a rolling horizon approximation in calculating the cost-to-go of the base heuristic.
- Because the heuristic is suboptimal, the rationale for a long rolling horizon becomes weaker.
- Example: N-stage stopping problem where the stopping cost is 0, the continuation cost is either $-\epsilon$ or 1, where $0 < \epsilon < 1/N$, and the first state with continuation cost equal to 1 is state m. Then the optimal policy is to stop at state m, and the optimal cost is $-m\epsilon$.



- Consider the heuristic that continues at every state, and the rollout policy that is based on this heuristic, with a rolling horizon of $l \leq m$ steps.
- It will continue up to the first $m - l + 1$ stages, thus compiling a cost of $-(m-l+1)\epsilon$. The rollout performance improves as l becomes shorter!

DISCRETIZATION

- If the state space and/or control space is continuous/infinite, it must be replaced by a finite discretization.

- Need for consistency, i.e., as the discretization becomes finer, the cost-to-go functions of the discretized problem converge to those of the continuous problem.

- Pitfalls with discretizing continuous time.

- The control constraint set changes a lot as we pass to the discrete-time approximation.

- Example:

$$\dot{x}_1(t) = u_1(t), \qquad \dot{x}_2(t) = u_2(t),$$

with the control constraint $u_i(t) \in \{-1, 1\}$ for $i = 1, 2$. Compare with the discretized version

$$x_1(t+\Delta t) = x_1(t)+\Delta t u_1(t),\ x_2(t+\Delta t) = x_2(t)+\Delta t u_2(t),$$

with $u_i(t) \in \{-1, 1\}$.

- "Convexification effect" of continuous time.

GENERAL APPROACH FOR DISCRETIZATION I

- Given a discrete-time system with state space S, consider a finite subset $\overline{S}$; for example $\overline{S}$ could be a finite grid within a continuous state space S. Assume stationarity for convenience, i.e., that the system equation and cost per stage are the same for all times.

- We define an approximation to the original problem, with state space $\overline{S}$, as follows:

- Express each $x \in S$ as a convex combination of states in $\overline{S}$, i.e.,

$$x = \sum_{x_i \in \overline{S}} \gamma_i(x) x_i \quad \text{where } \gamma_i(x) \geq 0, \ \sum_i \gamma_i(x) = 1$$

- Define a "reduced" dynamic system with state space $\overline{S}$, whereby from each $x_i \in \overline{S}$ we move to $\overline{x} = f(x_i, u, w)$ according to the system equation of the original problem, and then move to $x_j \in \overline{S}$ with probabilities $\gamma_j(\overline{x})$.

- Define similarly the corresponding cost per stage of the transitions of the reduced system.

GENERAL APPROACH FOR DISCRETIZATION II

- Let $\overline{J}_k(x_i)$ be the optimal cost-to-go of the "reduced" problem from each state $x_i \in \overline{S}$ and time k onward.

- Approximate the optimal cost-to-go of any $x \in S$ for the original problem by

$$\tilde{J}_k(x) = \sum_{x_i \in \overline{S}} \gamma_i(x) \overline{J}_k(x_i),$$

and use one-step-lookahead based on $\tilde{J}_k$.

- The choice of coefficients $\gamma_i(x)$ is in principle arbitrary, but should aim at consistency, i.e., as the number of states in $\overline{S}$ increases, $\tilde{J}_k(x)$ should converge to the optimal cost-to-go of the original problem.

- Interesting observation: While the original problem may be deterministic, the reduced problem is always stochastic.

- Generalization: The set $\overline{S}$ may be any finite set (not a subset of $\overline{S}$) as long as the coefficients $\gamma_i(x)$ admit a meaningful interpretation that quantifies the degree of association of x with x_i.

OTHER SUBOPTIMAL CONTROL APPROACHES

- **Minimize the DP equation error:** Approximate the optimal cost-to-go functions $J_k(x_k)$ with functions $\tilde{J}_k(x_k, r_k)$, where r_k is a vector of unknown parameters, chosen to minimize some form of error in the DP equations.

- **Approximate directly control policies:** For a subset of states x^i, $i = 1, \ldots, m$, find

$$\hat{\mu}_k(x^i) = \arg \min_{u_k \in U_k(x^i)} E\Big\{ g(x^i, u_k, w_k) + \tilde{J}_{k+1}\big(f_k(x^i, u_k, w_k), r_{k+1}\big) \Big\}.$$

Then find $\tilde{\mu}_k(x_k, s_k)$, where s_k is a vector of parameters obtained by solving the problem

$$\min_s \sum_{i=1}^{m} \|\hat{\mu}_k(x^i) - \tilde{\mu}_k(x^i, s)\|^2.$$

- **Approximation in policy space:** Do not bother with cost-to-go approximations. Parametrize the policies as $\tilde{\mu}_k(x_k, s_k)$, and minimize the cost function of the problem over the parameters s_k.