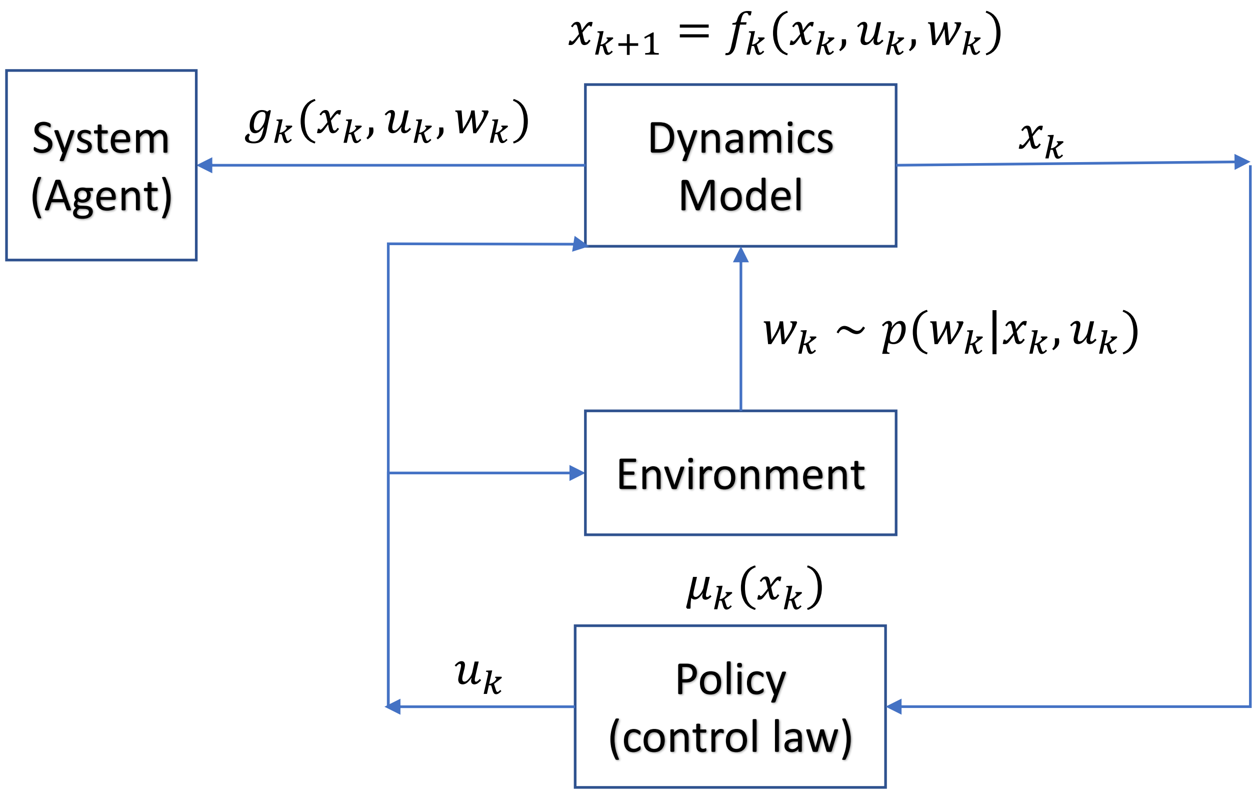
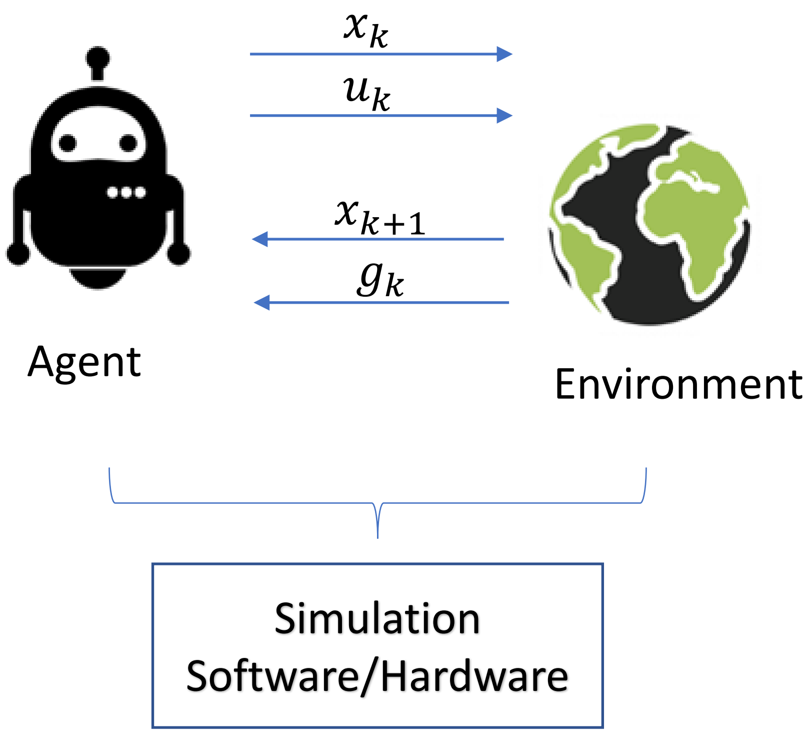
Model Predictive Control

Model-based X Model-free



Model-based Case



Model-free Case

Stochastic Dynamic Programming

- Let's recall our general DP formulation that is (ideally) solved by the backward DP recursion:

$$J_N(x_n) = g_N(x_N)$$

$$J_i(x_i) = \min_{u_i \in U_i(x_i)} \left\{ \mathbb{E}_{w_i} \left[g_i(x_i, u_i, w_i) + J_{i+1}(f_i(x_i, u_i, w_i)) \right] \right\}, \forall i \in \{0, ..., N-1\}$$

- Now, we will let:

$$x_i \in \mathbb{R}^n, \qquad u_i \in \mathbb{R}^m, \qquad w_i \in \mathbb{R}^d \longrightarrow$$

Continuous Vectors

$$U_i(x_i) \subset \mathbb{R}^m \longrightarrow$$

Sets:
Convex Sets
Polyhedrons
etc

Recap: Linear Quadratic Regulator (LQR)

- We actually saw one type of problem like this before, which was the LQR problem, with linear dynamics:

$$x_{k+1} = A_k x_k + B_k u_k + w_k, \forall k \in \{0, 1, ..., N-1\}$$

- And the cost is quadratic:

$$\mathbb{E}_w \left[x_N^\top Q_N x_N + \sum_{k=0}^{N-1} (x_k^\top Q_k x_k + u_k^\top R_k u_k) \right]$$

- Where the matrices $Q_k's$ are symmetric p.s.d. and the matrices are $R_k's$ are symmetric p.d.. Furthermore

$$\mathbb{E}[w_k] = 0, \, \mathbb{E}[w_k^2] < \infty, \, \forall k \in \{0, 1, ..., N-1\}$$

Recap: Linear Quadratic Regulator (LQR)

- The (backwards) DP recursion for the LQR problem is given by:

$$J_N(x_N) = x_N^\top Q_N x_N$$

$$J_k(x_k) = \min_{u_k} \left\{ \mathbb{E}_{w_k} \left[x_k^\top Q_k x_k + u_k^\top R_k u_k + J_{k+1}(A_k x_k + B u_k + w_k) \right] \right\}$$

- And the solution is the Riccati Recursion:

$$P_N = Q_N$$

$$P_k = A_k^\top (P_{k+1} - P_{k+1} B_k (B_k^\top P_{k+1} B_k + R_k)^{-1} B_k^\top P_{k+1}) A_k + Q_k, \; \forall k \in \{0, 1, ..., N-1\}$$

$$K_k = -(R_k + B_k^\top P_{k+1} B_k)^{-1} B_k^\top P_{k+1} A_k$$

Recap: Linear Quadratic Regulator (LQR)

- The key feature of the LQR problem, is that the optimal policy is **linear** on the state:

$$\mu_k^*(x_k) = K_k x_k$$

- where the matrix K_k (called the *control gain matrix*).

- And the optimal Value Function is **quadratic** on the states:

$$J_0^*(x_0) = x_0^\top P_0 x_0 + \sum_{k=0}^{N-1} \mathbb{E}_{w_k}\left[w_k^\top P_{k+1} w_k\right]$$

Recap: Linear Quadratic Regulator (LQR)

- And we saw that for Infinite-Horizon problem, given the conditions of controllability and observability, then the optimal **stationary policy** is linear:

$$\mu^*(x) = Kx, \qquad \begin{cases} K = -(B^\top KB + R)^{-1}B^\top PA \\ P = A^\top (P - PB(B^\top PB + R)^{-1}B^\top P)A + Q \end{cases}$$

- And the closed-loop system:

$$x_{k+1} = Ax_k + Bu_k = (A + BK)x_k, \quad \forall k \in \{0, 1, 2, ...\}$$

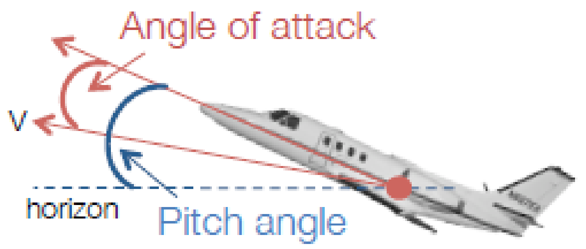
- Is stable (has spectral radius less than unity).

Example: Controlling an Airplane

- We illustrate the use of LQR with Airplane control (example taken from Borelli and Morari):

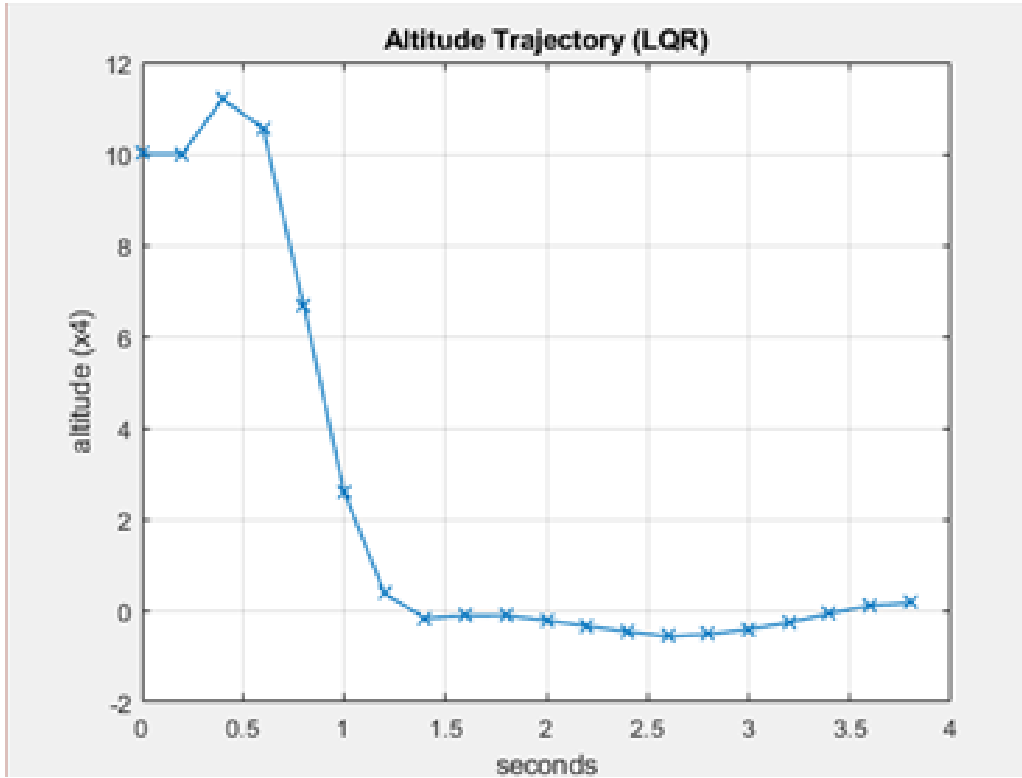
$$A = \begin{bmatrix} 0.74 & 0 & 1.96 & 0 \\ 0 & 1 & 0.2 & 0 \\ -1.09 & 0 & 0.63 & 0 \\ -25.64 & 25.64 & 0 & 1.0 \end{bmatrix} \qquad B = \begin{bmatrix} -0.06 \\ 0 \\ -3.4 \\ 0 \end{bmatrix} \qquad Q = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad R = 10$$

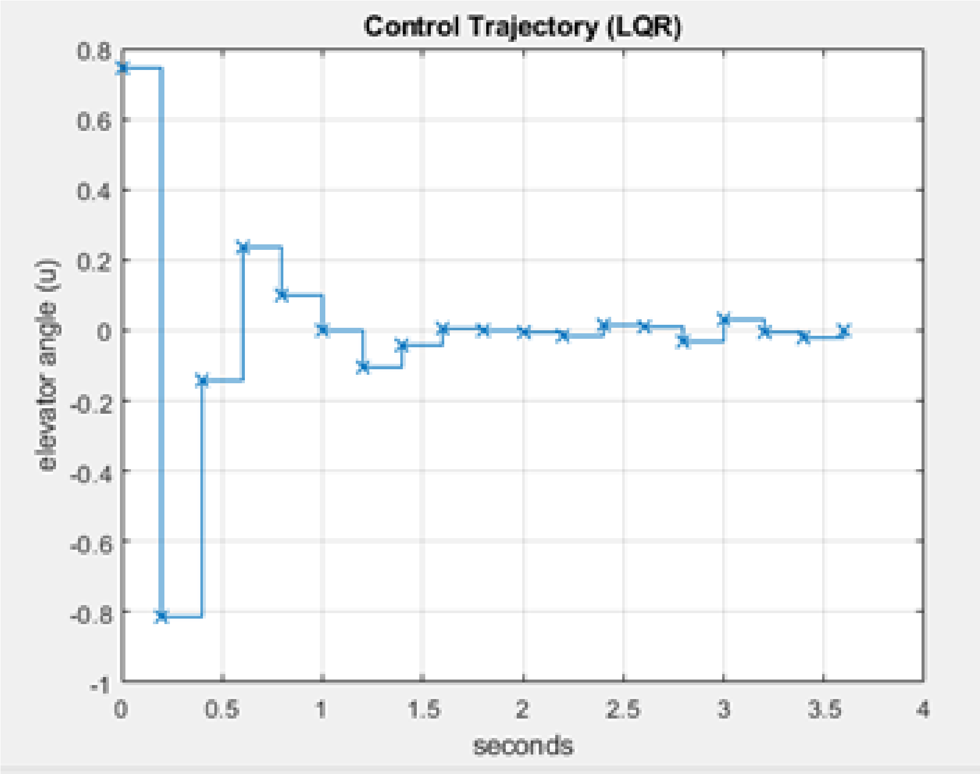
- Where:
  - x_1 angle attack; $x_2$is the pitch angle; x_3 pitch rate; x_4 altitude.
  - u is the elevator angle.
  - No constraints.



Example: Controlling an Airplane

- Starting from an altitude deviation of 10m, so $x_0 = [0,0,0,10]$.
- We can compute the LQR controller using the Riccati Recursion and the result is as follows:





Example: Controlling an Airplane

- If the horizon is infinity, the closed-loop system is **unstable**

- We can solve the Riccati Equation, obtaining:

$$K = \begin{bmatrix} -17.06 & 23.03 & 1.05 & 0.32 \end{bmatrix}$$

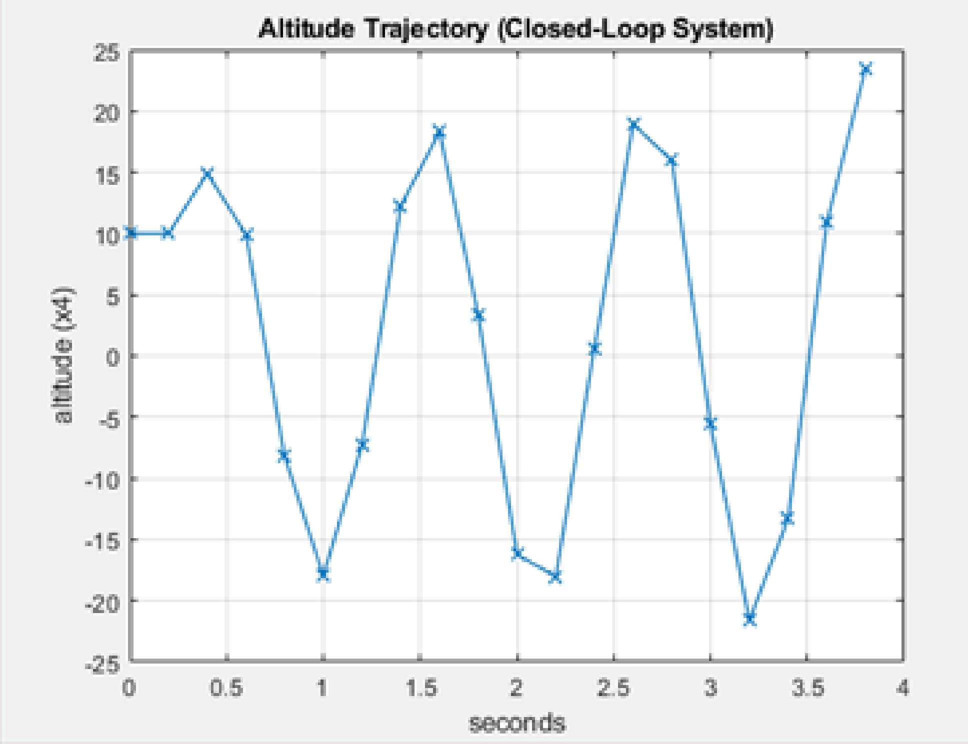
- And compute the closed-loop system:

$$A + BK = \begin{bmatrix} 1.77 & -1.38 & 0.1329 & -0.019 \\ 0 & 1 & 0.2 & 0 \\ 58.9 & -78.3 & -2.95 & -1.08 \\ -25.64 & 25.64 & 0 & 1.0 \end{bmatrix}$$

- Which has spectral radius bigger than unity.

Example: Controlling an Airplane

- The instability is verified by simulation:



Model Predictive Control (MPC)

- The example highlights of one the issues with LQR:
  - It cannot handle constraints
  - It only handles linear dynamics

- Let's write the DP problem again **with no** disturbances:

$$
\begin{aligned}
J^*(\bar{x}_0) = \min_{X,U} \quad & \sum_{i=0}^{\infty} x_i^\top Q x_i + u_i^\top R u_i \\
\text{s.t.} \quad & x_{i+1} = A x_i + B u_i, \quad \forall i \in \{0, 1, 2, ...\} \\
& x_i \in \mathcal{X}, \quad \forall i \in \{0, 1, 2, ...\} \\
& u_i \in \mathcal{U}, \quad \forall i \in \{0, 1, 2, ...\} \\
& x_0 = \bar{x}_0
\end{aligned}
$$

Model Predictive Control (MPC)

- Solving this problem is hard as it has infinite horizon and constraints. The goal of MPC is to solve, instead, the following N-step lookahead problem:

$$
\begin{aligned}
J_0(\bar{x}_0) = \min_{X,U} \;& \hat{J}_N(x_N) + \sum_{i=0}^{N-1} x_i^\top Q x_i + u_i^\top R u_i \\
\text{s.t. } & x_{i+1} = A x_i + B u_i, \quad \forall i \in \{0,1,2,\dots\} \\
& x_i \in \mathcal{X}, \quad \forall i \in \{0,1,2,\dots\} \\
& u_i \in \mathcal{U}, \quad \forall i \in \{0,1,2,\dots\} \\
& x_0 = \bar{x}_0 \\
& x_N \in \mathcal{X}_f
\end{aligned}
$$

Terminal Cost Approximation

Terminal set constraint

Model Predictive Control (MPC)

- The MPC algorithm essentially solves the previous optimization and obtain the open sequence of controls $(u_0^*, \dots, u_{N-1}^*)$.

- It then applies **just** the first control. Hence, the closed-loop policy is:

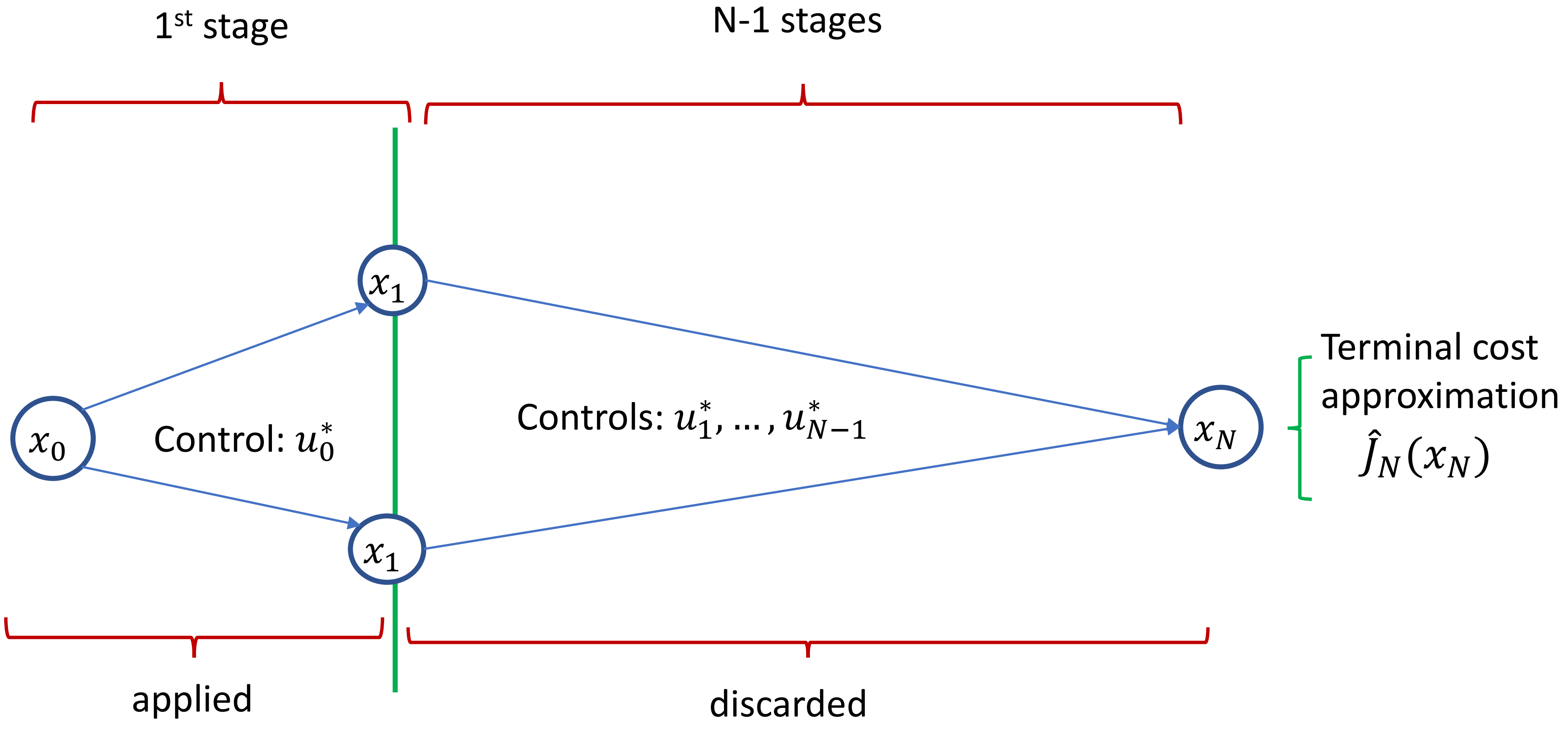
$$\mu_{\mathrm{MPC}}(\bar{x}_0) = u_0^*$$

- Then we move the horizon forward one time step and resolve the problem
  - Now from stages 1 to N+1
  - Starting from the new state $x_1 = Ax_0 + Bu_0^*$

- And then we repeat.

Model Predictive Control (MPC)

- This procedure is exactly the Rollout Algorithm where we perform N-step lookahead minimization:



Model Predictive Control (MPC)

- So using our definitions of the Rollout Algorithm:

- The solution $(u_0^*, \ldots, u_{N-1}^*)$ plays the role of the *Base Policy:*
  - It creates an open-loop trajectory:

$$(x_0, u_0^*, x_1, u_1^*, ..., x_{N-1}, u_{N-1}^*, x_N)$$

- The MPC policy is the associated **Rollout Policy:**

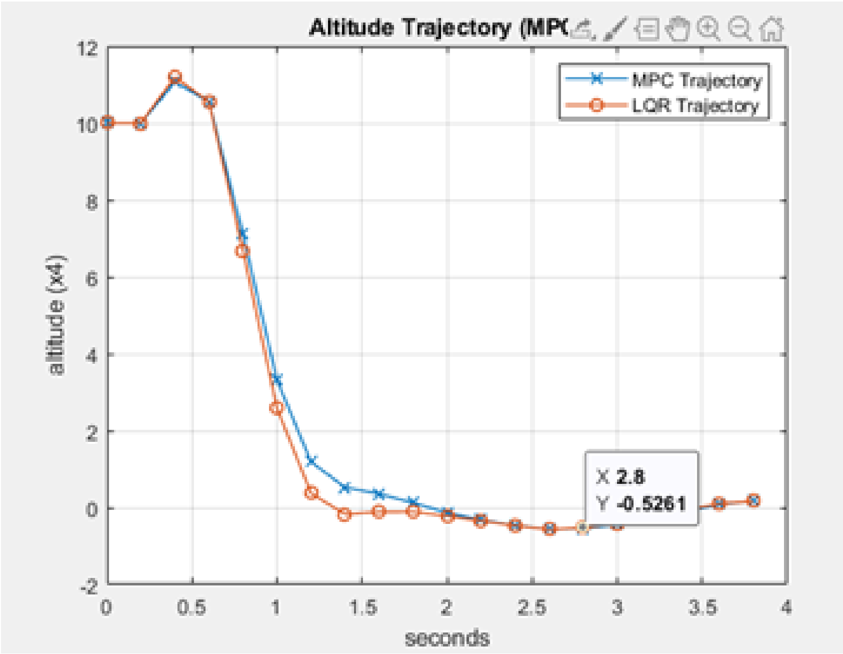
$$\mu_{\mathrm{MPC}}(x_0) \in \arg \min_{u_0 \in \mathcal{U}} \left\{ x_0^\top Q x_0 + u_0^\top R u_0 + \tilde{J}_1(Ax_0 + Bu_0) \right\}$$

- Where $\tilde{J}_1(\cdot)$ is the cost-to-go of the open loop trajectory:

$$\tilde{J}_1(x_1) = \hat{J}_N(x_N) + \sum_{k=1}^{N-1} x_k^\top Q x_k + u_k^{*\top} R u_k^*$$

Example: Applying MPC to Airplane Control

- We apply the MPC Algorithm to our previous Airplane Model.
- The MPC Algorithm correctly solves the issues presented by the LQR algorithm.
- We compute the controls in the receding horizon fashion of the Rollout Algorithm:



Model Predictive Control (MPC)

- Usually on MPC, the goal is to stabilize the system, that is design a policy that is closed-loop stable:
  - So we drive the system to the origin.

- Without disturbances, if we *reach* the origin, then we stay there at zero cost
  - This is evident by our choice of quadratic p.s.d. cost matrices.

- So in an Infinite-Horizon problem, if we solve the MPC and the terminal state is the origin 0, then we will remain there indefinetly.

- Hence the “sub-optimality” of MPC lies in the *Transient Period* of the (deterministic) DP problem.

Model Predictive Control (MPC)

- Let's state again the MPC problem, now starting at some state x_k:

$$\begin{aligned} J_k(\bar{x}_k) = \min_{X,U} \quad & \hat{J}_{k+N}(x_{k+N}) + \sum_{i=k}^{N+k-1} x_i^\top Q x_i + u_i^\top R u_i \\ \text{s.t.} \quad & x_{i+1} = Ax_i + Bu_i, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\ & x_i \in \mathcal{X}, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\ & u_i \in \mathcal{U}, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\ & x_k = \bar{x}_k \\ & x_{k+N} \in \mathcal{X}_f \end{aligned}$$

- We can ask the following questions:
  - (1) Is it always feasible?
  - (2) Does it improve the cost? (that is does it have the Policy Improvement?)

Example: scalar system

- Consider this problem:

$$
\begin{aligned}
J_0(\bar{x}_k) = \min_{X,U} \quad & x_{k+N}^2 + \sum_{i=k}^{N+k-1} x_i^2 \\
\text{s.t.} \; & x_{i+1} = 2x_i + u_i, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& -\beta \leq x_i \leq \beta, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& -1 \leq u_i \leq 1, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& x_k = \bar{x}_k
\end{aligned}
$$

- Where β is some positive scalar. Let's see what happens to the MPC algorithm as change β.

- Suppose we start at $0 \leq x_0 < 1$.

Example: scalar system

- Since there are no cost on the controls, we can set $u_0 = -1$ and obtain:

$$x_1 = 2x_0 - 1 < x_0$$

- Which is closer to zero than x_0. And we keep applying $u_k = -1$ until we reach a state $x_{k'}$ where:

$$0 \leq x_{k'} \leq \frac{1}{2}$$

- Once this happens, the feasible control $u_{k'} = -2x_{k'}$ will drive the state to 0.

- A similar behavior happens if we start from $-1 < x_0 \leq 0$.

Example: scalar system

- So if $\beta \leq 1$, we are fine, and the MPC is feasible for a horizon length N (that is dependent on β.
  - For example if $\beta < \frac{1}{2}$ we can let N=1.

- Now suppose $\beta \geq 1$. And we start from $x_0 \in [1, \beta]$, t hen it is impossible to bring the system to the origin.

- Moreover the system is unstable and will diverge to infinity:

$$x_k \to \infty, \text{ as } \quad k \to \infty$$

- And this happens no matter how big the horizon length N is.

Example: Double-Integrator

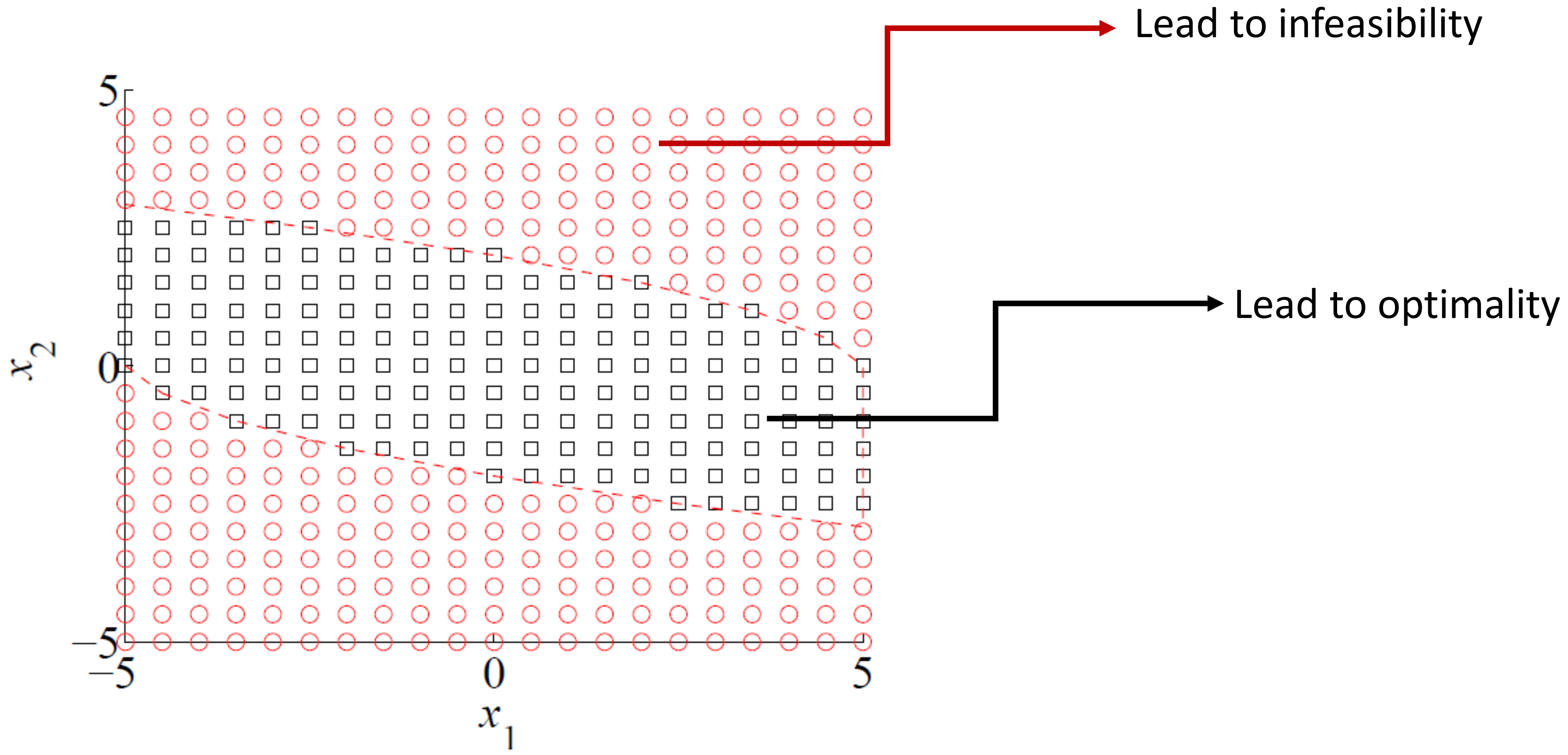
- Consider this problem:

$$
\begin{aligned}
J_0(\bar{x}_k) = \min_{X,\,U} \quad & x_{k+N}^\top \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_{k+N} + \sum_{i=k}^{N+k-1} x_i^\top \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_i + u_i^2 \\
\text{s.t.} \quad & x_{i+1} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x_i + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& \begin{bmatrix} -5 \\ -5 \end{bmatrix} \leq x_i \leq \begin{bmatrix} 5 \\ 5 \end{bmatrix}, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& -0.5 \leq u_i \leq 0.5, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& x_k = \bar{x}_k
\end{aligned}
$$

- Where N = 3.

Example: Double-Integrator

- If we apply the MPC Algorithm this problem may become infeasible depending on where we start:



(figure taken form the Book “Predictive Control, F. Borrelli, A. Bemporad, M. Morari”)

Model Predictive Control (MPC)

- The two questions raised and illustrated by the examples can be boiled down to two desirable properties:
  - <u>**(1) Recursive Feasibility:**</u> As we apply the MPC Algorithm, we want the Optimal Control problem to be feasible at every stage.
  - <u>**(2) Asymptotic Stability:**</u> As we apply the MPC Algorithm, we want to get closer and closer to the origin at the system evolves.
- As the examples show, the initial state x_0 play a key role in determining these properties
- In addition, the horizon length (that is the “lookahead window”) N is also important.
- And it turns out that both the terminal cost approximation and terminal set constraint also play a role in this.

Model Predictive Control (MPC)

- Let's start simple. Suppose:

$$\mathcal{X}_f = 0$$

- So if we start from some state x_k then we enforce that we reach the origin in N steps. The Optimal Control becomes:

$$J_k(\bar{x}_k) = \min_{X,U} \quad \cancelto{0}{\hat{J}_{k+N}(x_{k+N})} + \sum_{i=k}^{N+k-1} x_i^\top Q x_i + u_i^\top R u_i$$

$$\begin{aligned}
\text{s.t. } & x_{i+1} = Ax_i + Bu_i, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& x_i \in \mathcal{X}, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& u_i \in \mathcal{U}, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& x_k = \bar{x}_k \\
& x_{k+N} = 0
\end{aligned}$$

Model Predictive Control (MPC)

- Suppose N is fixed. Consider the following set:

$$\mathcal{X}_0 = \left\{ x \in \mathcal{X} : \exists (u_0, u_1, ..., u_{N-1}) \text{ s.t.: } \begin{cases} x_{k+1} = Ax_k + Bu_k, \forall k \in \{0, ..., N-1\} \\ x_k \in \mathcal{X},\ u_k \in \mathcal{U}, \forall k \in \{0, ..., N-1\} \\ x_0 = x, x_N = 0 \end{cases} \right\}$$

- In words: “this is the set of all initial states x_0 such that the Optimal Control Problem is feasible.

- Assume for the moment we are able to compute this set and we select some $x_0 \in \mathcal{X}_0$ as our starting stage.

Model Predictive Control (MPC)

- Since $x_0 \in \mathcal{X}_0$, the MPC Algorithm is feasible for the very first step.

- Let $(u_0^*, u_1^*, .. u_{N-1}^*)$ be the optimal solution (the optimal open-loop sequence for the first step).

- The MPC algorithm applies the 1st control and discards the rest:

$$\mu_{\mathrm{MPC}}(x_0) = u_0^*$$

- The system then evolves:

$$x_1 = Ax_0 + Bu_0^*$$

Model Predictive Control (MPC)

- At x_1, the following control sequence is **<u>feasible:</u>**

$$(u_1^*, u_2^*, ..., u_{N-1}^*, 0)$$

- Note that it may not be optimal for the new problem, but that is fine as we are concerned with feasibility here.

- By induction we repeat the same argument and show that for all stages $k \geq 0$ the Optimal Control problem is feasible.

- Hence, if we start feasible ⇒ stay feasible: **<u>Recursive Feasibility</u>**

Model Predictive Control (MPC)

- Now let's focus on the question of stability. We want to show that by applying the MPC Algorithm we eventually reach the origin.
  - Note that this is essentially asking to prove that the MPC Algorithm has the Policy Improvement property of the Rollout Algorithm.

- Let $J_0(x_0)$ be the total cost-to-go when we start from x_0 and solve the Optimal Control Problem.

- We want to show that:

$$J_1(x_1) \leq J_0(x_0)$$

- In words: "we want to show that the cost-to-go decreases as the MPC Algorithm progresses".

Model Predictive Control (MPC)

- This can be verified by just writing down the costs:

$$J_0(x_0) + \sum_{i=0}^{N-1} x_i^\top Q x_i + u_i^{*\top} R u_i^*$$

$$J_1(x_1) = \sum_{i=1}^{N-1} x_i^\top Q x_i + u_i^{*\top} R u_i^* = \sum_{i=0}^{N-1} x_i^\top Q x_i + u_i^{*\top} R u_i^* - (x_0^\top Q x_0 + u_0^{*\top} R u_0^*)$$

$$J_1(x_1) = J_0(x_0) - (x_0^\top Q x_0 + u_0^{*\top} R u_0^*) \leq J_0(x_0)$$

- By arguing inductively we have:

$$J_{k+1}(x_{k+1}) \leq J_k(x_k), \quad \forall k \geq 0$$

Model Predictive Control (MPC)

- Now to obtain asymptotic stability, note that if we sum over K periods

$$J_k(x_k) + \sum_{i=0}^{k-1} (x_i^\top Q x_i + u_i^\top R u_i) \leq J_0(x_0)$$

- This is true for all k. And since all the costs are non negative it follows that:

$$\sum_{i=0}^{k-1} (x_i^\top Q x_i + u_i^\top R u_i) \leq J_0(x_0)$$

- Now if let $k \to \infty$ we obtain that:

$$\sum_{i=0}^{\infty} (x_i^\top Q x_i + u_i^\top R u_i) < \infty$$ ⇒ The infinite sum is finite: **Asymptotic Stability**

Model Predictive Control (MPC)

- So for $\mathcal{X}_f = 0$, we can draw the following conclusions:

- The set of feasible initial states $x_0 \in \mathcal{X}_0$ guarantee that all Optimal Control problems in the future stages will be feasible.

- The MPC Algorithm has the Policy Improvement Property: the Closed-Loop system is asymptotically stable.

- The challenge here is computing this set:

$$\mathcal{X}_0 = \left\{ x \in \mathcal{X} : \exists (u_0, u_1, ..., u_{N-1}) \text{ s.t.: } \begin{cases} x_{k+1} = Ax_k + Bu_k, \forall k \in \{0, ..., N-1\} \\ x_k \in \mathcal{X},\, u_k \in \mathcal{U}, \forall k \in \{0, ..., N-1\} \\ x_0 = x, x_N = 0 \end{cases} \right\}$$

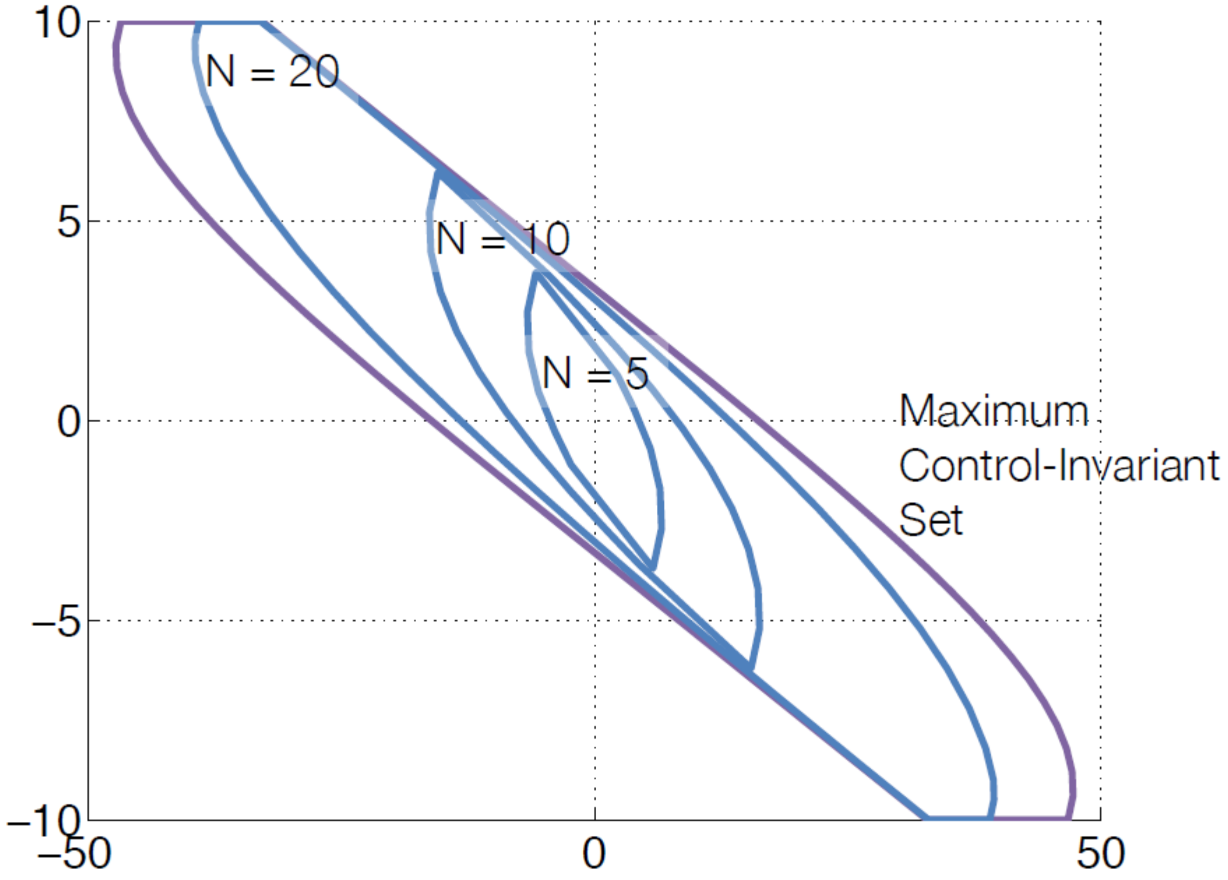
Example: Impact of horizon length

- Consider this example:

$$
\begin{aligned}
J_0(\bar{x}_k) = \min_{X,U} \quad & x_{k+N}^\top \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_{k+N} + \sum_{i=k}^{N+k-1} x_i^\top \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_i + 1u_i^2 \\
\text{s.t. } & x_{i+1} = \begin{bmatrix} 1.2 & 1 \\ 0 & 1 \end{bmatrix} x_i + \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} u_i, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& \begin{bmatrix} -50 \\ -10 \end{bmatrix} \leq x_i \leq \begin{bmatrix} 50 \\ 10 \end{bmatrix}, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& -1 \leq u_i \leq 1, \quad \forall i \in \{k, k+1, ..., N+k-1\} \\
& x_k = \bar{x}_k
\end{aligned}
$$

Example: Impact of horizon length

- We can plot the how the initial feasible set changes as the Horizon changes:



(figure taken form the Book “Predictive Control, F. Borrelli, A. Bemporad, M. Morari”)