Lecture 7: Guaranteeing Feasibility and Stability

I. Receding Horizon Control and Model Predictive Control

- **Recall: Infinite Time Constrained Optimal Control (what we would like to solve)**

$$J_0^*(x(0)) = \min \sum_{k=0}^{\infty} q(x_k, u_k)$$

$$\begin{aligned} \text{s.t. } & x_{k+1} = Ax_k + Bu_k,\ k = 0,\ \dots, N-1 \\ & x_k \in \mathcal{X}, u_k \in \mathcal{U},\ k = 0, \dots, N-1 \\ & x_0 = x(0) \end{aligned}$$

stage cost $q(x,u)$ is the cost of being in the state x and applying the input u

Optimizing over a trajectory provides a **tradeoff between short- and long-term benefits of actions** such a control law has many beneficial properties... but **we can't compute it: there are an infinite number of variables**

- **Recall: Receding Horizon Control (what we can sometimes solve)**

$$J_t^*(x(t)) = \min_{U_t} \ p(x_{t+N}) + \sum_{k=0}^{N-1} q(x_{t+k}, u_{t+k})$$

$$\begin{aligned} \text{s.t. } & x_{t+k+1} = Ax_{t+k} + Bu_{t+k},\ k = 0,\ \dots, N-1 \\ & x_k \in \mathcal{X}, u_k \in \mathcal{U},\ k = 0, \dots, N-1 \\ & x_{t+N} \in \mathcal{X}_f \\ & x_0 = x(0) \end{aligned}$$

Where $U_t = \{u_t, \dots, u_{t+N-1}\}$

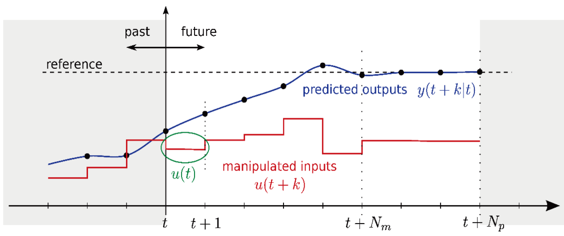
Truncate after a finite horizon:

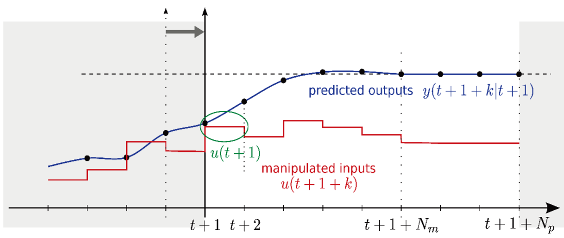
$p(x_{t+N})$ Approximation of the 'tail' of the cost, related to the stability of the controller

$\mathcal{X}_f$ Approximation of the 'tail' of the constraints, related to not violating the constraints

- **Recall: On-Line RHC (MPC)**

1. MEASURE the state $x(t)$ at time instant t
2. OBTAIN $U_t^*(x(t))$ by solving the optimization problem (CFTOC).
3. IF $U_t^*(x(t)) = \emptyset$; THEN 'problem infeasible' STOP
4. APPLY the first element u_t^* of U_t^* to the system
5. WAIT for the new sampling time $t+1$, GOTO 1





II. Motivation

- **Recall: Objective of Constrained Optimal Control**

Given the dynamical system:

$$x^{+}=f(x,u) \qquad (x,u)\in\mathcal{X},\ \mathcal{U}$$

Our objective is to design the control law $u=\kappa(x)$ such that system:

1. Satisfies constraints: $\{x_i\}\subset\mathcal{X},\ \{u_i\}\subset\mathcal{U}$ (feasibility)
2. Is asymptotically stable: $\lim_{i\to\infty} x_i=0$ (stability)
3. Optimizes "performance"
4. Maximizes the set $\{x_0 \mid \text{Conditions } 1-3 \text{ are met}\}$

In previous lectures, we mainly focused on optimal control basics (LQR for unconstrained problems, multiparametric programming for constrained norm minimization problems). Some receding horizon control (predictive control) examples and results were given but not discussed in detail. Starting from this lecture, **we will focus on Model Predictive Control details and demonstrate that the objectives of COC can be met in the MPC framework.**

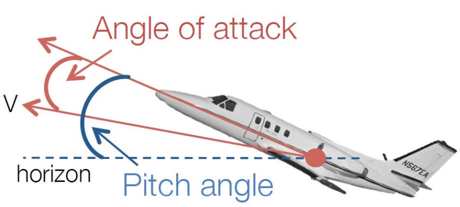
Here we can see that before talking about the performance of the controller, **we first have to guarantee the feasibility and stability of the controller**. If they cannot be guaranteed, in no sense could we talk about the performance.

- **Example: Cessna Citation Aircraft**

Consider the Linearized continuous-time model of the aircraft (at an altitude of 5000m and a speed of 128.2 m/sec)

$$\dot{x}=\begin{bmatrix}-1.2882 & 0 & 0.98 & 0\\ 0 & 0 & 1 & 0\\ -5.4293 & 0. & -1.8366 & 0\\ -128.2 & 128.2 & 0 & 0\end{bmatrix}x+\begin{bmatrix}-0.3\\ 0\\ 17\\ 0\end{bmatrix}u$$

$$y=\begin{bmatrix}0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}x$$



Input: elevator angle. States: x_1: angle of attack, x_2: pitch angle, x_3: pitch rate, x_4: altitude

Output: pitch angle and altitude

Constraints: elevator angle ± 0.262 rad ($\pm 15°$), elevator rate ± 0.524 rad ($\pm 60°$), pitch angle ± 0.349 ($\pm 39°$)

Open-loop response is unstable (open-loop poles: $0, 0, -1.5994 \pm 2.29i$)

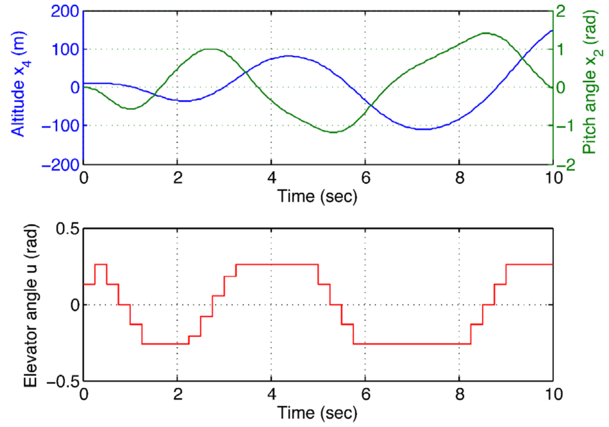
In this case, we will compare the LQR and MPC in almost the same conditions:

Costs: Quadratic cost, Linear system dynamics with weight matrices $Q=Q^{\mathrm{T}}\succeq 0, R=R^{\mathrm{T}}\succ 0$

Constraints: In MPC, additional Linear constraints on inputs and states that can be directly assigned. In LQR, we use the anti-windup method (clipping) to deal with the constraints

Initial Conditions: At time $t = 0$ the plane is flying with a deviation of 10 m of the desired altitude, i.e. $x_0 = [0 \quad 0 \quad 0 \quad 10]^{\mathrm{T}}$

Case 1: LQR with saturation



Settings: Linear quadratic regulator with saturated inputs, i.e. clip input to maximum when exceeding).

Parameters:

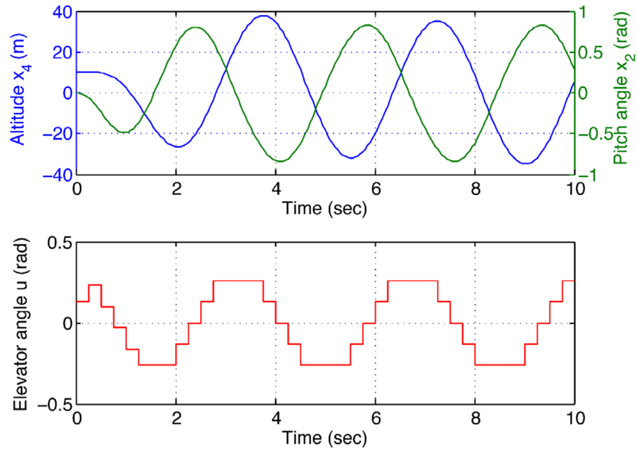
Sampling time 0:25sec, $Q = I, R = 10$

Results:

Closed-loop system is unstable

Applying LQR control and saturating the controller can lead to instability!

Case 2: MPC with Bound Constraints on Inputs



Settings: MPC controller with input constraints $|u_i| \leq 0.262$

Parameters:

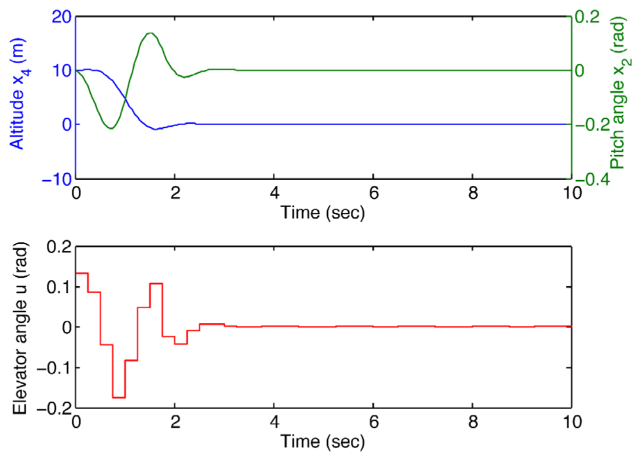
Sampling time 0:25sec, $Q = I, R = 10, N = 10$

Results:

The MPC controller uses the knowledge that the elevator will saturate, but **it does not consider the rate constraints.**

System does not converge to desired steady-state but to a limit cycle

Case 3: MPC with all Input Constraints



Settings: MPC controller with input constraints $|u_i| \leq 0.262$ and rate constraints $|\dot{u}_i| \leq 0.349$ (approximated by $|u_k - u_{k-1}| \leq 0.349 T_s$)

Parameters:

Sampling time 0:25sec, $Q = I, R = 10, N = 10$

Results:

The MPC controller considers all constraints on the actuator. Efficient use of the control authority.

Closed-loop system is stable.

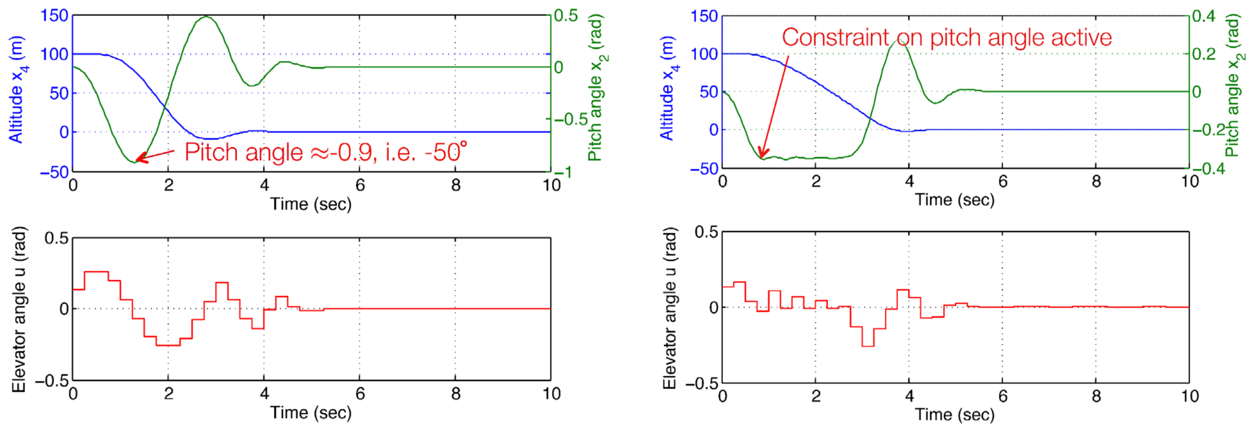
Case 4: MPC with Inclusion of state constraints

Settings: Based on the result of case 3, if at time $t=0$ the plane is flying with a deviation of 100 m of the desired altitude, i.e. $x_0=[0 \quad 0 \quad 0 \quad 100]^{\mathrm{T}}$. The same control will result in too large pitch angle during transient (left below). Therefore, we can add state constraints for passenger comfort: i.e. $|x_2| \leq 0.349$ (right below). **Showing the strength of MPC in dealing with multiple constraints.**

Parameters:

Sampling time 0:25sec, $Q=I, R=10, N=10$

Results:



Special Case: MPC with short horizon

Settings: The same as case 3

Parameters:

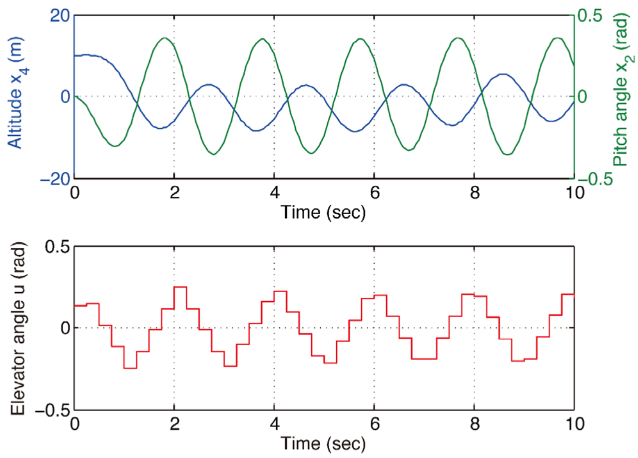
Sampling time 0:25sec, $Q=I, R=10, N=4$

Results:

Closed-loop system is unstable

Decrease in the prediction horizon causes a loss of the stability properties

Note: Finding the proper horizon $N=10$ needs trial and error. A longer horizon does not necessarily mean a stable closed-loop system. Recall the example given in last lecture.



III. Challenges of MPC: Feasibility and Stability

What can go wrong with "standard" MPC? $\Rightarrow$ Two fundamental but important limitations

No **feasibility** guarantee, i.e., the MPC problem may not have a solution

No **stability** guarantee, i.e., trajectories may not converge to the origin

- **Example: Loss of feasibility - Double Integrator**

Consider the double integrator:

$$x(t+1)=\begin{bmatrix}1 & 1\\0 & 1\end{bmatrix}x(t)+\begin{bmatrix}0\\1\end{bmatrix}u(t)$$

$$y(t)=[1\ \ 0]x(t)$$

subject to the input constraints:

$$-0.5\le u(t)\le 0.5$$

and the state constraints:

$$\begin{bmatrix}-5\\-5\end{bmatrix}\le x(t)\le\begin{bmatrix}5\\5\end{bmatrix}$$

Compute a receding horizon controller with quadratic objective with:

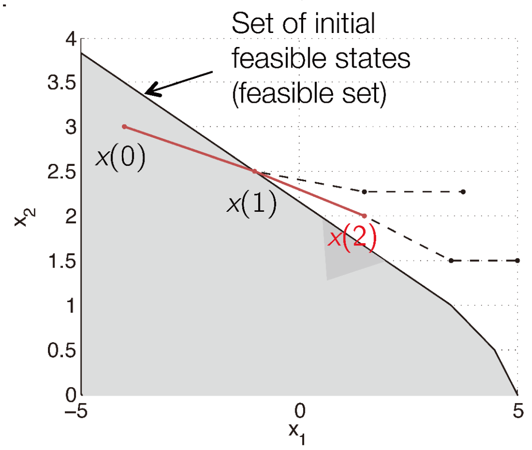
$$N=3, P=Q=\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}, R=10$$

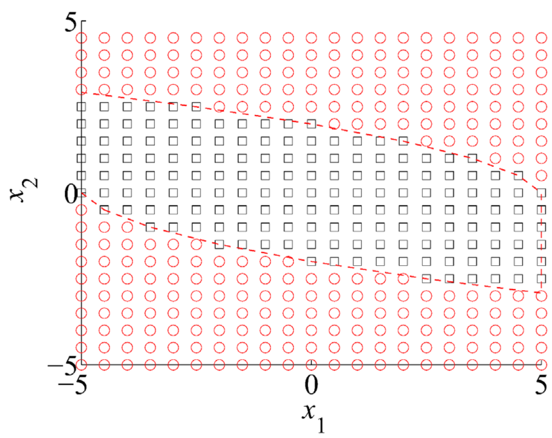
Time step 0: $x_0=[-4\ \ 3]^{\mathrm{T}}, u_0^*(x)=-0.5$

Time step 1: $x_0=[-1\ \ 2.5]^{\mathrm{T}}, u_0^*(x)=-0.5$

Time step 2: $x_0=[1.5\ \ 2]^{\mathrm{T}}$, problem infeasible

Depending on the initial condition, closed-loop trajectory may lead to states for which the optimization problem is infeasible. Boxes (Circles) on the right below are initial points leading (not leading) to feasible closed-loop trajectories.





Note:

Why don't we also add this set of initial feasible states (grey region) into constraints as well?

The set of initial feasible states for a closed loop system (grey region) is calculated by solving the true constrained infinite horizon optimal control problem (in other words, calculating MPC with horizon $N=\infty$), which requires a global perspective and a lot of computing resources. It is generally too hard to be prespecified in most of the problems.

Also see the example and summary below

- **Example: Feasibility and stability are function of tuning**

Given an unstable system:

$$x(t+1)=\begin{bmatrix}2 & 1\\ 0 & 0.5\end{bmatrix}x(t)+\begin{bmatrix}1\\ 0\end{bmatrix}u(t)$$

With input constraints: $-1\le u(t)\le 1$

State constraints: $\begin{bmatrix}-10\\ -10\end{bmatrix}\le x(t)\le\begin{bmatrix}10\\ 10\end{bmatrix}$, parameters $Q=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$

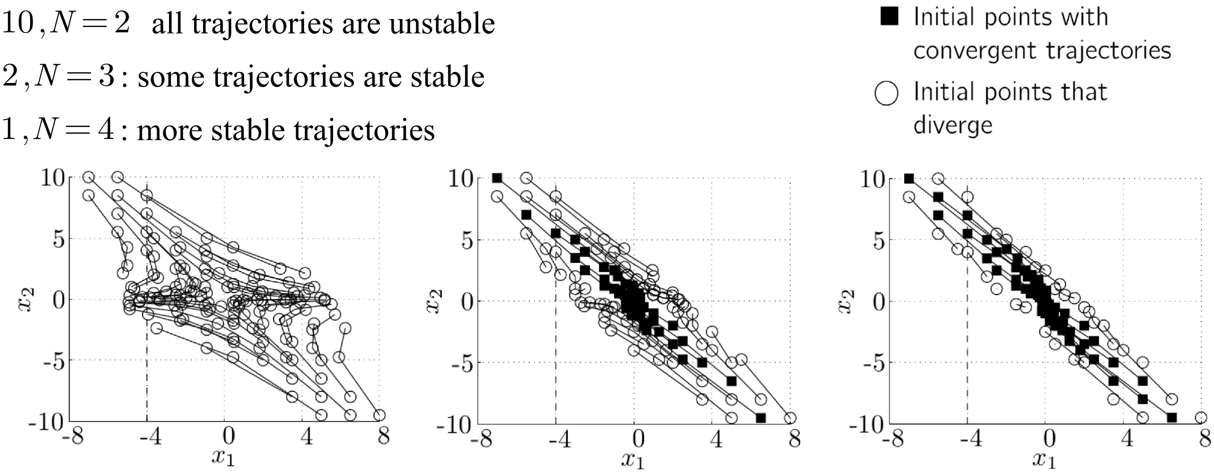
Investigate the stability properties for different horizons N and weights R by solving the finite-horizon MPC problem in a receding horizon fashion

The results are shown below:

$R=10, N=2$ all trajectories are unstable

$R=2, N=3$: some trajectories are stable

$R=1, N=4$: more stable trajectories



Case 1 Case 2 Case 3

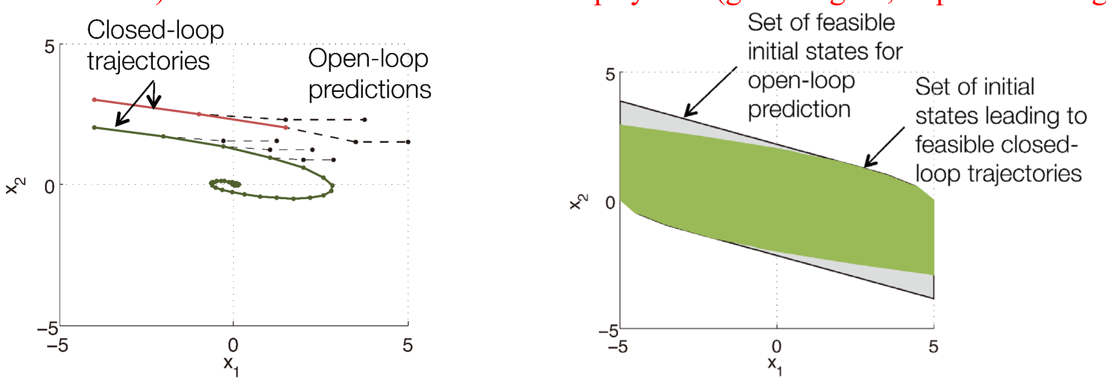
Feasible initial points depend on the horizon N but not on the cost R

⇒ In fact, parameters have complex effects on trajectories.

- **MPC: Where Do Feasibility and Stability issues come from?**

Problems **originate from the use of a 'short-sighted' strategy (the cut-off on the horizon)**

⇒ The finite horizon causes deviation between the feasible set of open-loop prediction (grey region, MPC calculation) and the feasible set of closed-loop system (green region, requires solving CIHOC)



Ideally, we should solve the MPC problem with infinite horizon, but that is computationally intractable.

⇒ Therefore, we need to design a finite horizon problem such that it approximates the infinite horizon.

- **Summary: Feasibility and Stability**

Infinite-Horizon

If we solve the RHC problem for $N=\infty$ (as done for LQR), then the open loop trajectories are the same as the closed-loop trajectories. Hence:

1. If the problem is feasible, the closed-loop trajectories will always be feasible
2. If the cost is finite, then states and inputs will converge asymptotically to the origin

But always remember that the infinite horizon constrained problem is computationally intractable!

Finite-Horizon in Receding Horizon fashion

RHC (MPC) is a strategy approximating infinite horizon controller, But in a "short-sighted" way, hence:

Feasibility. After some steps, the finite horizon optimal control problem may become infeasible. (Infeasibility occurs even without disturbances and model mismatch!)

Stability. The generated control inputs may not lead to trajectories that converge to the origin.

IV. Guarantee Feasibility and Stability: Theorems

- **Feasibility and stability in MPC – Solution and Proof Sketch**

Main idea: Introduce terminal cost and constraints to explicitly ensure feasibility and stability

$$
\begin{aligned}
J_0^*(x(t)) = \min_{U_0}\ & p(x_N) + \sum_{k=0}^{N-1} q(x_k,u_k) \\
\text{s.t. } & x_{k+1} = Ax_k + Bu_k,\ k=0,\dots,N-1 \\
& x_k \in \mathcal{X}, u_k \in \mathcal{U},\ k=0,\dots,N-1 \\
& x_N \in \mathcal{X}_f \\
& x_0 = x(t)
\end{aligned}
$$

$p(\cdot)$, $\mathcal{X}_f$ are chosen to mimic infinite horizon, corresponding to stability and feasibility, respectively.

We would prove the stability and feasibility with terminal cost and constraints, through two main steps:

- First, prove recursive feasibility by showing the existence of a feasible control sequence at all time instants when starting from a feasible initial point
- Then prove stability by showing that the optimal cost function is a Lyapunov function

We would consider two cases:

1. Terminal constraint at zero: $x_N = 0$
2. Terminal constraint in some general (convex polytopic) set: $x_N \in \mathcal{X}_f$

General notation:

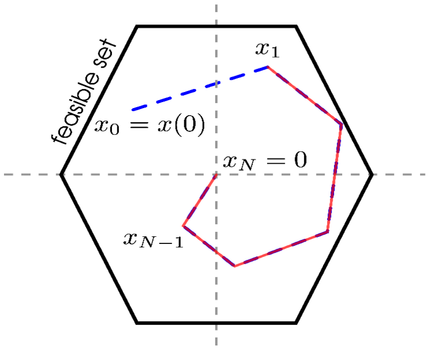
$$J_0^*(x(t)) = \min_{U_0} \underbrace{p(x_N)}_{\text{Terminal cost}} + \sum_{k=0}^{N-1} \underbrace{q(x_k,u_k)}_{\text{stage cost}}$$

Cost is quadratic cost:

$$q(x_i,u_i) = x_i^{\mathrm{T}} Q x_i + u_i^{\mathrm{T}} R u_i,\ p(x_N) = x_N^{\mathrm{T}} P x_N$$

- **Proof for Terminal constraint at zero:** $x_N = \mathcal{X}_f = 0$

Feasibility (Recursive Feasibility):



1. **Assume feasibility** of x_0 and let $\{u_0^*, u_0^*, \ldots, u_{N-1}^*\}$ be the optimal control sequence computed at x_0 and $\{x(0), x_1, \ldots, x_N\}$ be the corresponding state trajectory
2. Apply u_0^* and let the system evolve to $x(1) = Ax_0 + Bu_0^*$
3. At $x(1)$, only need to let the control sequence be $\{u_1^*, u_2^*, \ldots, u_{N-1}^*, 0\}$, (i.e. apply old sequence and at $t = N$, use 0 control input and we can ensure $\Rightarrow x_N = x_{N+1} = 0$)

Therefore, we prove the recursive feasibility

Stability: Goal is to show that $J_0^*(x_1) < J_0^*(x_0),\ \forall x_0 \neq 0$

$$J_0^*(x_1) \leq \tilde{J}_0(x_1) = \sum_{i=1}^{N} q(x_i, u_i^*)$$

$$J_0^*(x_0) = \underbrace{p(x_N)}_{=0} + \sum_{i=0}^{N-1} q(x_i, u_i^*) \quad \Rightarrow \quad = \sum_{i=0}^{N-1} q(x_i, u_i^*) - q(x_0, u_0^*) + q(x_N, u_N^*)$$

$$= J_0^*(x_0) - \underbrace{q(x_0, u_0^*)}_{>0} + \underbrace{q(0,0)}_{=0,\ \because x_N = 0, u_N = 0}$$

$\Rightarrow J_0^*(x_1) \leq \tilde{J}_0(x_1) = J_0^*(x_0) - q(x_0, u_0^*) < J_0^*(x_0) \Rightarrow J_0^*$ is a Lyapunov function $\Rightarrow$ stability ✔

Note:

1. $\tilde{J}_0(x_1)$ here means we do not do optimization and directly use sub-optimal $\{u_1^*, u_2^*, \ldots, u_{N-1}^*, 0\}$
2. All the above proof sketch is based on the assumption that we have the **feasibility of** x_0 **given a horizon** N, i.e. initial feasibility. If from the very start, say, a very short horizon is given or only a very limited control can be applied so that the problem is infeasible, then even if we follow the same induction process, **it would mean nothing**! Because it may not even let your state reach the final state $x_N = 0$, or in other words, the precondition we build on would be totally wrong. See the example below.
3. We show that the sub-optimal control input $\{u_1^*, u_2^*, \ldots, u_{N-1}^*, 0\}$ is feasible. It also implies that the new true optimal is feasible. It is because the next feasible set is always a subset of the previous feasible set. Recall the feasible set relation we mentioned in lecture 6.

Example: Impact of Horizon with Zero Terminal Constraint

$$x_{k+1} = \begin{bmatrix} 1.2 & 1 \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} u_k$$

Constraints:

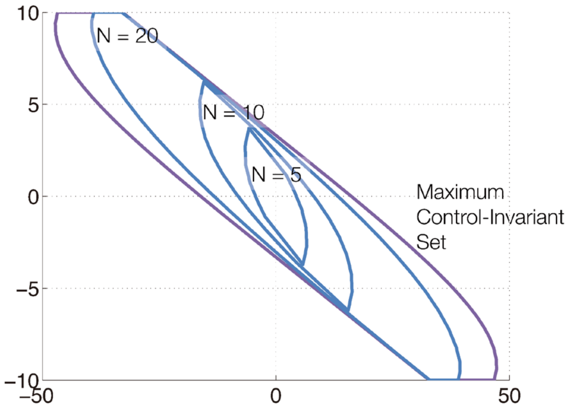
$$\mathcal{X} = \{x \mid -50 \leq x_1 \leq 50,\ -10 \leq x_2 \leq 10\} = \{x \mid A_x x \leq b_x\}$$

$$\mathcal{U} = \{u \mid \|u\|_\infty \leq 1\} = \{u \mid A_u u \leq b_u\}$$

Stage Cost:

$$q(x,u) = x^{\mathrm{T}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x + u^{\mathrm{T}} u$$

Note: We can see that with the increase of the horizon length N, the region of attraction is also enlarging and would become closer to the theoretical maximum control invariant set because we are more confident in getting initial feasibility with a longer horizon.



Summary for Terminal constraint at zero:

If we choose the terminal constraint: $x_N = \mathcal{X}_f = 0$, then:

1. The set of feasible initial states $\mathcal{X}_0$ is also the **set of initial states which are persistently feasible (feasible at all** $t \geq 0$**)** for the system in closed-loop with the designed MPC.
2. The equilibrium point $(0, 0)$ is asymptotically stable according to Lyapunov.
3. $J_0^*(x_0)$ is a Lyapunov function for the closed-loop system (system + MPC) defined over $\mathcal{X}_0$ and $\mathcal{X}_0$ **is the region of attraction** of the equilibrium point.

Note:

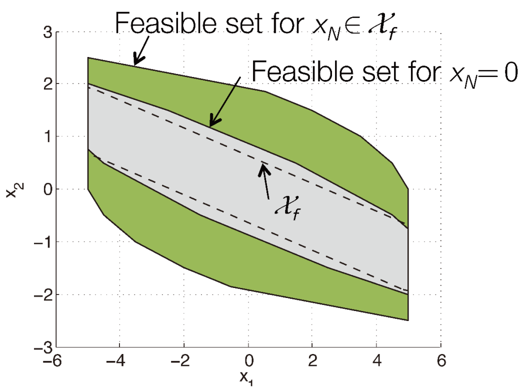
In the proof sketch, we can find that it is not limited to the form of quadratic cost, i.e. any positive definite cost would work, and we also do not assume that the form of the Lyapunov function has to be quadratic. This actually shows another very important characteristic of MPC compared to traditional control design methodology. Typically, people would do analysis, for instance, assuming a specific form of a Lyapunov function, then try to design the controller, which sometimes is a bit tricky and would suffer due to the lack of intuition when the problem scales to a higher dimension. However, in MPC, we do synthesis instead of analysis. In other words, MPC design actually implicitly contains the idea of Lyapunov stability in the whole process.

- **General Terminal (Convex) Set:** $x_N \in \mathcal{X}_f$ **: Motivation and Concepts**

Problem: The terminal constraint $x_N = \mathcal{X}_f = 0$ reduces the size of the feasible set (or, it requires a longer horizon N to accurately hit a single point, i.e. origin)

Goal: Use a convex set $\mathcal{X}_f$ to increase the region of attraction. In other words, we want to generalize proof to the constraint $x_N \in \mathcal{X}_f$

Example: Still consider the same double integrator, the numerical result with the same horizon N is shown here. If we can increase the scope of the terminal set, the corresponding feasible set could also become larger.

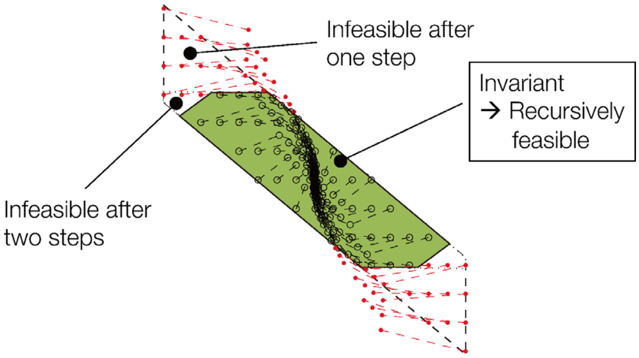


Definition (Invariant sets):

A set $\mathcal{O}$ is called positively invariant for system $x(t+1)=f_d(x(t))$, if:

$$x(0)\in\mathcal{O}\Rightarrow x(t)\in\mathcal{O},\ \forall t\in\mathbb{N}_+$$

The positively invariant set that contains every closed positively invariant set is called the maximal positively invariant set $\mathcal{O}_\infty$



Note: In our concepts, positively invariant implies recursively feasible.

Summary of important sets:

Consider the constrained system:

$$x(t+1)=g(x(t),u(t))$$
$$x(t)\in\mathcal{X},u(t)\in\mathcal{U}$$

and the MPC controller: $u(t)=\mathrm{MPC}(x(t),\ g,\ Q,\ R,\ P,\ N,\mathcal{X},\ \mathcal{U},\ \mathcal{X}_f)$

compactly rewritten as $u(t)=\mathrm{MPC}(x(t),\mathrm{par})$

below are the important set we will be considering in the future:

$\mathcal{X}$: State constraints: we want the system state to be in $\mathcal{X}$ at all time instants

$\mathcal{X}_0$: Set of $\bar{x}$ such that $\mathrm{MPC}(\bar{x},\mathrm{par})$ is feasible. A control input U_0 can only be found if $x(0)\in\mathcal{X}_0$. The set $\mathcal{X}_0$ depends on $\mathcal{X}$ and $\mathcal{U}$, on the controller horizon N and on the controller terminal set $\mathcal{X}_f$. It does not depend on the objective function.

$\mathcal{O}_\infty$: The maximum positive invariant set for the system in closed-loop with $u(t)=\mathrm{MPC}(x(t),\mathrm{par})$. It depends on the MPC controller and as such on all parameters affecting the controller, i.e. $N,\mathcal{X},\ \mathcal{U},\ \mathcal{X}_f$ and the objective function with its parameters P,Q and R. Clearly $\mathcal{O}_\infty\subseteq\mathcal{X}_0$ because if it were not there would be points in $\mathcal{O}_\infty$ for which the control problem is not feasible. Because of invariance, the closed-loop is persistently feasible for all states $x(0)\in\mathcal{O}_\infty$.

$\mathcal{C}_\infty$: The maximal control invariant set $\mathcal{C}_\infty$ for the system. It is only affected by the sets $\mathcal{X}$ and $\mathcal{U}$, the constraints on states and inputs. It is the largest set over which we can expect some controllers to work. $\mathcal{X}_0$ has generally no relation with $\mathcal{C}_\infty$ (it can be larger, smaller, etc), and clearly, $\mathcal{O}_\infty\subseteq\mathcal{C}_\infty$.

- **Proof for General Terminal (Convex Polytopic) Set:** $x_N=\mathcal{X}_f$

Assumptions:

1. Stage cost is positive definite, i.e. it is strictly positive and only zero at the origin
2. The terminal set is **invariant** under the local control law $v(x_k)$:

$$x_{k+1}=Ax_k+Bv(x_k)\in\mathcal{X}_f \text{ for all } x_k\in\mathcal{X}_f$$

All **state and input constraints** are satisfied in $\mathcal{X}_f$:

$$\mathcal{X}_f\subseteq\mathcal{X},\ v(x_k)\in\mathcal{U} \text{ for all } x_k\in\mathcal{X}_f$$

3. Terminal cost is a continuous Lyapunov function in the terminal set $\mathcal{X}_f$ and satisfies:

$$p(x_{k+1}) - p(x_k) \le -q(x_k, v(x_k)) \quad \text{for all} \quad x_k \in \mathcal{X}_f$$

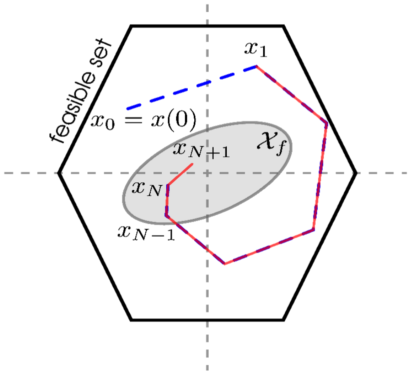
Note: the third assumption is not obvious, and it gives a criterion for selecting the terminal cost!

Theorem (Stability of MPC for General Terminal Set):

The closed-loop system under the MPC control law $u_0^*(x)$ is asymptotically stable, and the set $\mathcal{X}_f$ is positive invariant for the system: $x_{k+1} = Ax_k + Bu_0^*(k)$ if the above three assumptions hold.

Feasibility:

1. **Assume feasibility** of x_0 and let $\{u_0^*, u_0^*, \dots, u_{N-1}^*\}$ be the optimal control sequence computed at x_0 and $\{x(0), x_1, \dots, x_N\}$ be the corresponding state trajectory that lead the state to the **terminal region**
2. At $x(1)$, the control sequence $\{u_1^*, u_2^*, \dots, u_{N-1}^*, v(x_N)\}$, would be feasible (apply until u_{N-1}^* would bring the state to $x_N \in \mathcal{X}_f$, and based on our assumption, local control law $v(x_N)$ is feasible and would keep the state in $\mathcal{X}_f$)



Therefore, terminal constraint provides the recursive feasibility and we prove it.

Stability: Show that $J_0^*(x_1) < J_0^*(x_0),\ \forall x_0 \ne 0$. Still, consider:

$$J_0^*(x_0) = p(x_N) + \sum_{i=0}^{N-1} q(x_i, u_i^*)$$

As is mentioned above and similar to the previous proof sketch, we apply the feasible, sub-optimal sequence $\{u_1^*, u_2^*, \dots, u_{N-1}^*, v(x_N)\}$ and hence we have:

$$\begin{aligned} J_0^*(x_1) \le \tilde{J}_0(x_1) &= \sum_{i=1}^{N} q(x_i, u_i^*) + p(Ax_N + Bv(x_N)) \\ &= \sum_{i=0}^{N-1} q(x_i, u_i^*) + p(x_N) - q(x_0, u_0^*) - p(x_N) + q(x_N, v(x_N)) + p(Ax_N + Bv(x_N)) \\ &= J_0^*(x_0) - \underbrace{q(x_0, u_0^*)}_{>0} + \underbrace{[p(Ax_N + Bv(x_N)) - p(x_N) + q(x_N, v(x_N))]}_{\le 0 \ \because\ p(x_{k+1}) - p(x_k) \le -q(x_k, v(x_k))} \end{aligned}$$

$J_0^*(x_1) \le \tilde{J}_0(x_1) = J_0^*(x_0) - q(x_0, u_0^*) + p,\ q(x_0, u_0^*) > 0, p < 0 \Rightarrow J_0^*(x_1) < J_0^*(x_0)$

$J_0^*(x)$ is a Lyapunov function decreasing along the closed-loop trajectories

$\Rightarrow$ the closed-loop system under the MPC control law is asymptotically stable.

Summary:

If we choose the terminal constraint: $\mathcal{X}_f$, to be an invariant set (Assumption 2) and the terminal cost $p(x)$ to be a Lyapunov function with the described decreasing rate (Assumption 3), then:

1. The set of feasible initial states $\mathcal{X}_0$ is also the **set of initial states which are persistently feasible (feasible for all** $t \ge 0$**)** for the system in closed-loop with the designed MPC.

2. The equilibrium point $(0, 0)$ is asymptotically stable according to Lyapunov.
3. $J_0^*(x_0)$ is a Lyapunov function for the closed-loop system (system + MPC) defined over $\mathcal{X}_0$ and $\mathcal{X}_0$ **is the region of attraction** of the equilibrium point.

Note: Proof works for any nonlinear system and positive definite and continuous cost.

V. Guarantee Feasibility and Stability: Choices and Examples

In the previous section, we show the theorems and proofs for guaranteeing the feasibility and stability of the MPC. Then in practice, how should we use it?

- **Choice of Terminal Sets and Cost - Linear System, Quadratic Cost**

We need to design the terminal cost and the terminal set, and we want them to satisfy the three assumptions mentioned above (1. positive definite cost, 2. local control that gives invariant terminal set with no active constraints, and 3. Lyapunov function in the terminal set).

$\Rightarrow$ Core idea: make use of the LQR controller!

Design pipeline:

1. Design the unconstrained LQR control law:

$$F_\infty = -(B^\mathrm{T} P_\infty B + R)^{-1} B^\mathrm{T} P_\infty A$$

Where P_∞ is the solution to the discrete-time algebraic Riccati equation:

$$P_\infty = A^\mathrm{T} P_\infty A + Q - A^\mathrm{T} P_\infty B (B^\mathrm{T} P_\infty B + R)^{-1} B^\mathrm{T} P_\infty A$$

2. Choose the terminal weight $P = P_\infty$
3. Choose the terminal set $\mathcal{X}_f$ to be the maximum invariant set for the closed-loop system:

$$x_{k+1} = Ax_k + BF_\infty(x_k) \in \mathcal{X}_f \quad \text{for all} \quad x_k \in \mathcal{X}_f$$

Also, make sure that all state and input constraints are satisfied in this chosen terminal set:

$$\mathcal{X}_f \subseteq \mathcal{X},\ F_\infty x_k \in \mathcal{U} \quad \text{for all} \quad x_k \in \mathcal{X}_f$$

Note: maximum invariant set with constraints satisfaction would ensure all the future constraints satisfaction, i.e. recursively feasibility as mentioned and proved above. We do not talk about how to get the invariant set for now. The techniques of getting the invariant set would be discussed in the following lecture!

- **Example: Unstable Linear Systems**

Given system dynamics:

$$x_{k+1} = \begin{bmatrix} 1.2 & 1 \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} u_k$$

Constraints:

$$\mathcal{X} := \{x \mid -50 \le x_1 \le 50,\ -10 \le x_2 \le 10\} = \{x \mid A_x x \le b\}$$
$$\mathcal{U} := \{u \mid \|u\|_\infty \le 1\} = \{u \mid A_u u \le b_u\}$$

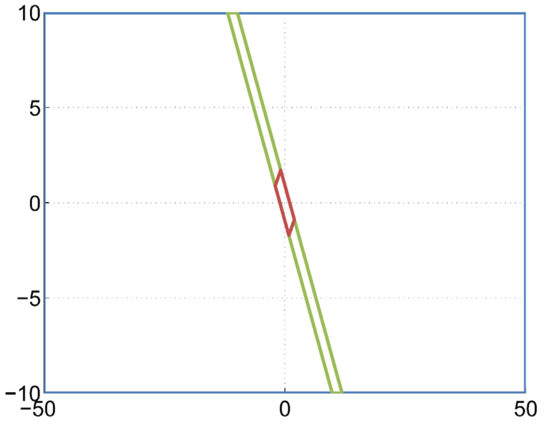
Stage cost:

$$q(x,u)=x^{\mathrm{T}}\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}x+u^{\mathrm{T}}u$$

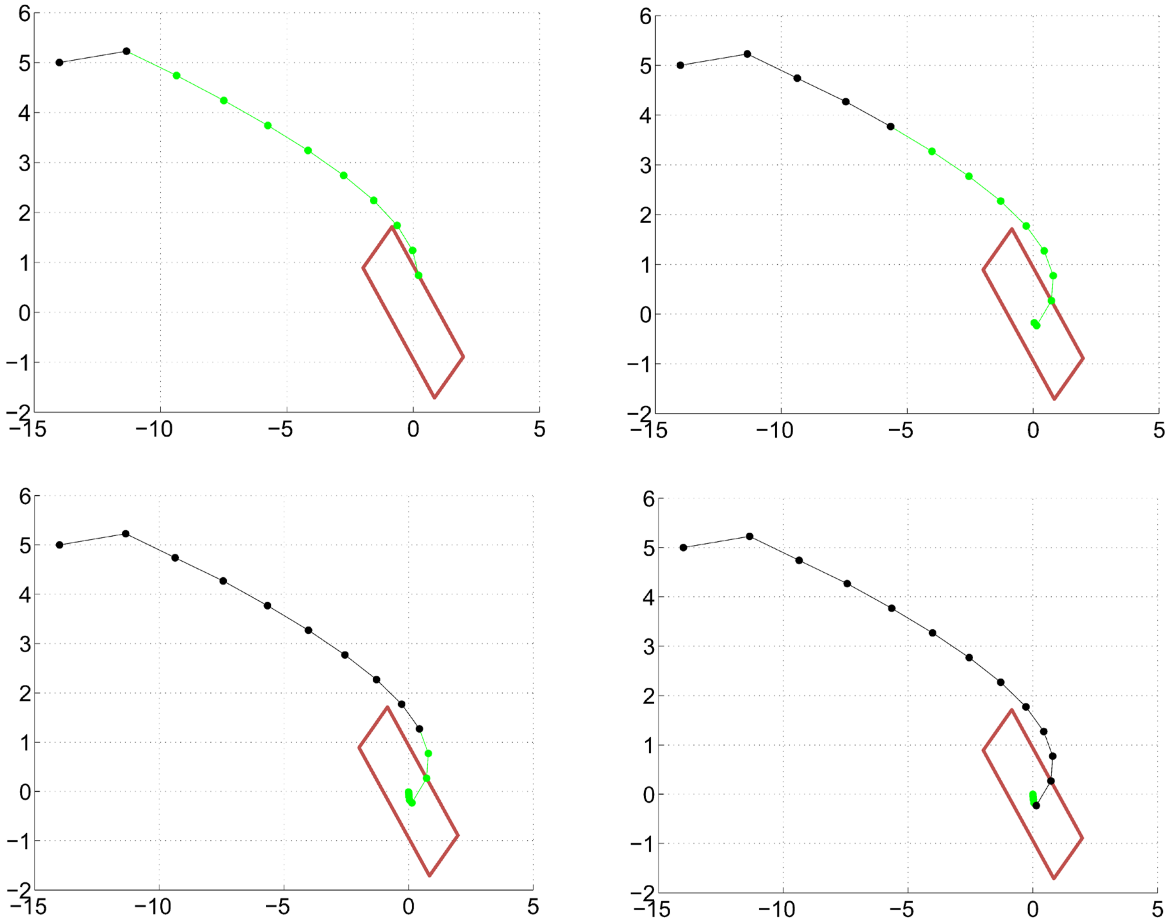
Horizon $N=10$

1. Compute the optimal LQR controller and cost matrices: F_∞, P_∞
2. Compute the maximal invariant set $\mathcal{X}_f$ for the closed-loop linear system $x_{k+1}=(A+BF_\infty)x_k$ subject to the constraints, as is shown below.

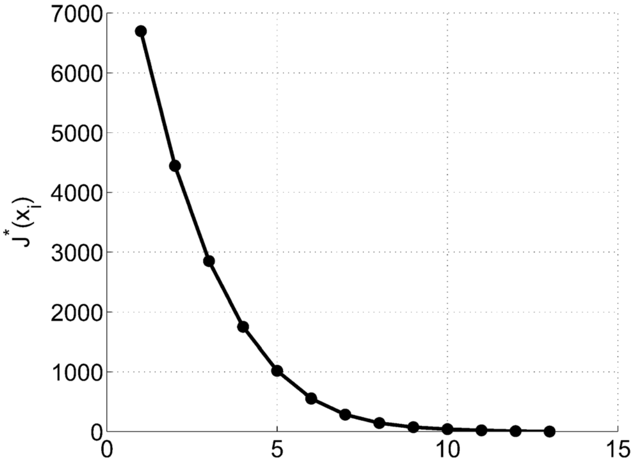
$$\mathcal{X}_{\mathrm{cl}}\left\{x \,\middle|\, \begin{bmatrix} A_x \\ A_u F_\infty \end{bmatrix} x \le \begin{bmatrix} b_x \\ b_u \end{bmatrix}\right\}$$



Note: The blue region is the box constraint for the states, i.e. $A_x x \le b_x$, the green region is the input constraint represented by $A_u F_\infty x \le b_u$, the red region is invariant set, i.e. everything starting from the red region would end up in the red region. The terminal region should be chosen as the red region because even if you start from the green region, it might still go out of the green region later!

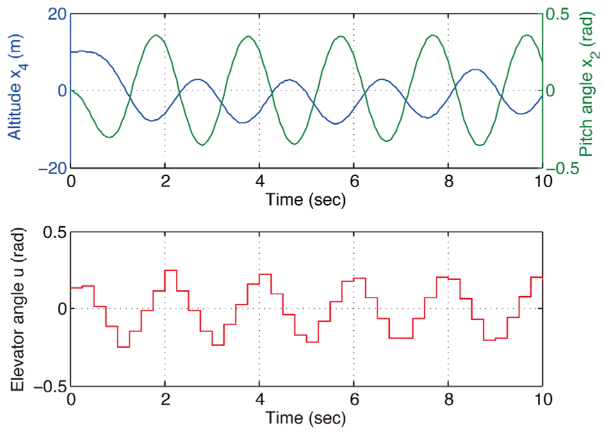


We can see that with horizon $N=10$, we can reach the terminal set **at the very first beginning** (i.e. initial feasibility). Since we have the initial feasibility, by recursive feasibility and stability for the general terminal set shown and proved above, we can guarantee the MPC controller is feasible and stable in the end, as the following figure shows.



- **Example: Short horizon with Terminal set**

Recall the aircraft example at the beginning of this lecture:



Parameters: MPC controller with input constraints $|u_i| \leq 0.262$ and rate constraints $|\dot{u}_i| \leq 0.349$ (approximated by $|u_k - u_{k-1}| \leq 0.349T_s$)

Sampling time 0:25sec, $Q=I, R=10, N=4$

Results:

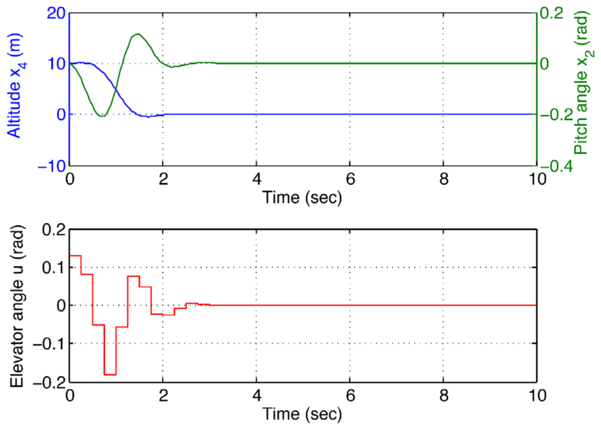
Closed-loop system is unstable

Decrease in the prediction horizon causes loss of the stability properties

Improvement using terminal set:

Add the LQR terminal cost and terminal set constraints, still keeping the same short horizon

Results: Inclusion of terminal cost and constraint provides stability



Note: Always remember that we use the terminal cost and terminal set to "mimic" the infinite horizon, Also, we need initial feasibility and have the three assumptions for the terminal cost and terminal set.

- **Stability of MPC – Practical Remarks and Summary**

Remarks and Summary:

1. The terminal set $\mathcal{X}_f$ and the terminal cost ensure recursive feasibility and stability of the closed-loop system. But, the terminal constraint reduces the region of attraction.

   Solution:

   We extend the concept of the terminal set from a single point (origin) to general convex polytopes

   We can extend the horizon to a sufficiently large value to increase the region

   Note: That is to say, the terminal set and horizon somewhat play against each other in the complexity of the problem and deciding the region of attraction. Terminal set with only origin ($\mathcal{X}_f = x_N = 0$) is the easiest to do, but it may require a long horizon to first ensure feasibility at the very beginning (recall that our proof for recursive feasibility all starts with initial feasibility), so the region of attraction is small. General convex terminal set $\mathcal{X}_f$ (polytopes) gives more flexibility. We can use a shorter horizon to reach the terminal set (since reaching a goal of a single point is much stricter than reaching a goal of a set) and guarantee feasibility and stability. However, we need to calculate the invariant terminal set, which will be shown in the next lecture and definitely is not easy.

2. Are general terminal sets $\mathcal{X}_f$ used in practice? Generally not because
   a. Not well understood by practitioners.
   b. Requires advanced tools to compute invariant set (polyhedral computation or LMI)
   c. Reduces region of attraction while a 'real' controller must provide some input in every case.
   d. Often unnecessary or overkill because if the system is stable and if we take a long horizon, the MPC controller will be stable and feasible in a large neighborhood of the origin

VI. Extension to Nonlinear MPC

Again, summarize the core idea of this lecture:

1. An infinite-horizon provides stability and invariance.
2. We 'fake' infinite-horizon by forcing the final state to be in an invariant set for which there exists an invariance-inducing controller, whose infinite-horizon cost can be expressed in closed-form.

It is not hard to see that these ideas extend to nonlinear systems, but the sets are difficult to compute:

$$J_0^*(x(t)) = \min_{U_0} \; p(x_N) + \sum_{k=0}^{N-1} q(x_k, u_k)$$

$$\begin{aligned} \text{s.t. } & x_{k+1} = g(x_k, u_k),\; k = 0, \ldots, N-1 \\ & x_k \in \mathcal{X}, u_k \in \mathcal{U},\; k = 0, \ldots, N-1 \\ & x_N \in \mathcal{X}_f \\ & x_0 = x(t) \end{aligned}$$

Also, note that in our proof sketch for stability and feasibility:

1. Presented assumptions on the terminal set and cost did not rely on linearity
2. Lyapunov stability is a general framework to analyze the stability of nonlinear dynamic systems

Therefore, the results can be directly extended to nonlinear systems. However, computing the sets $\mathcal{X}_f$ and function $p(x_N)$ can be very difficult!