

# AA203

# Optimal and Learning-based Control

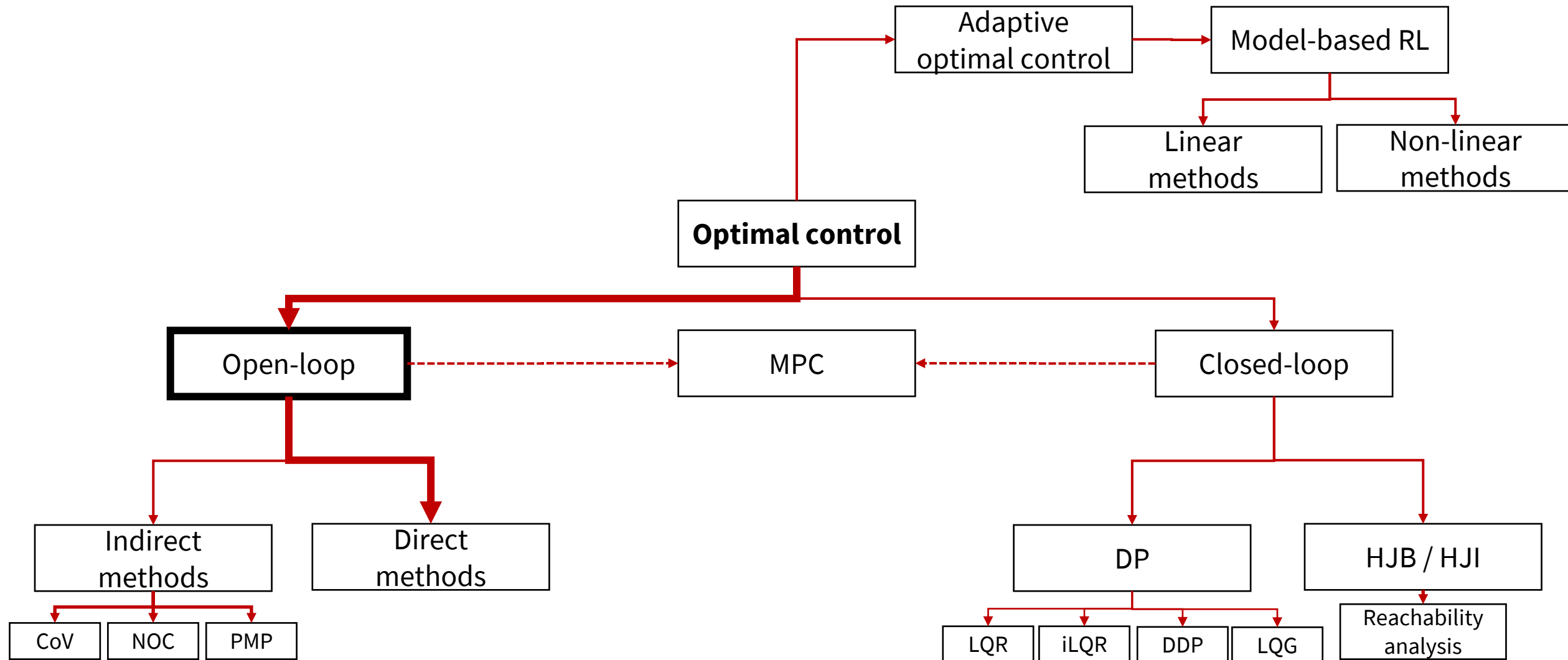
Direct methods for optimal control: fundamental concepts\*



\*Slides co-prepared by Riccardo Bonalli,  
[rbonalli@stanford.edu](mailto:rbonalli@stanford.edu)



# Roadmap



# Optimal control problem

$$\min \int_0^{t_f} g(\mathbf{x}(t), \mathbf{u}(t), t) dt$$

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t), \quad t \in [0, t_f]$$

**(OCP)**

$$\mathbf{x}(0) = \mathbf{x}_0, \quad \mathbf{x}(t_f) \in M_f$$

$$\mathbf{u}(t) \in U \subseteq \mathbb{R}^m, \quad t \in [0, t_f]$$

For simplicity:

- We assume the terminal cost  $h$  is equal to 0
- We assume  $t_0 = 0$

- Indirect Methods:

1. Apply necessary conditions for optimality to **(OCP)**
2. Solve a two-point boundary value problem

- Direct Methods:

1. Transcribe **(OCP)** into a nonlinear, constrained optimization problem
2. Solve the optimization problem via nonlinear programming

# Transcription into nonlinear programming

Forward Euler time discretization

$$\min \int_0^{t_f} g(\mathbf{x}(t), \mathbf{u}(t), t) dt$$

**(OCP)**

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t), \quad t \in [0, t_f]$$

$$\mathbf{x}(0) = \mathbf{x}_0, \quad \mathbf{x}(t_f) \in M_f$$

$$\mathbf{u}(t) \in U \subseteq \mathbb{R}^m, \quad t \in [0, t_f]$$

1. Select a discretization  $0 = t_0 < t_1 < \dots < t_N = t_f$  for the interval  $[0, t_f]$  and, for every  $i = 0, \dots, N - 1$ , define  $\mathbf{x}_{i+1} \sim \mathbf{x}(t)$ ,  $\mathbf{u}_i \sim \mathbf{u}(t)$ ,  $t \in (t_i, t_{i+1}]$  and  $\mathbf{x}_0 \sim \mathbf{x}(0)$
2. By denoting  $h_i = t_{i+1} - t_i$ , **(OCP)** is transcribed into the following nonlinear, constrained optimization problem

$$\min_{(\mathbf{x}_i, \mathbf{u}_i)} \sum_{i=0}^{N-1} h_i g(\mathbf{x}_i, \mathbf{u}_i, t_i)$$

**(NLOP)**

$$\mathbf{x}_{i+1} = \mathbf{x}_i + h_i \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i, t_i), \quad i = 0, \dots, N - 1$$

$$\mathbf{u}_i \in U, \quad i = 0, \dots, N - 1, \quad F(\mathbf{x}_N) = 0$$

# Transcription into nonlinear programming

## Consistency of Time Discretization

**Is this approximation consistent with the original formulation?**

**Yes!**

Indeed, the KKT conditions for **(NLOP)** converge to the necessary optimality conditions for **(OCP)**, that are given by the Pontryagin's Minimum Principle, when  $h_i \rightarrow 0$

## Forward Euler time discretization

1. Select a discretization  $0 = t_0 < t_1 < \dots < t_N = t_f$  for the interval  $[0, t_f]$  and, for every  $i = 0, \dots, N - 1$ , define  $\mathbf{x}_{i+1} \sim \mathbf{x}(t)$ ,  $\mathbf{u}_i \sim \mathbf{u}(t)$ ,  $t \in (t_i, t_{i+1}]$  and  $\mathbf{x}_0 \sim \mathbf{x}(0)$
2. By denoting  $h_i = t_{i+1} - t_i$ , **(OCP)** is transcribed into the following nonlinear, constrained optimization problem

$$\min_{(\mathbf{x}_i, \mathbf{u}_i)} \sum_{i=0}^{N-1} h_i g(\mathbf{x}_i, \mathbf{u}_i, t_i)$$

**(NLOP)**

$$\mathbf{x}_{i+1} = \mathbf{x}_i + h_i \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i, t_i), \quad i = 0, \dots, N - 1$$

$$\mathbf{u}_i \in U, i = 0, \dots, N - 1, \quad F(\mathbf{x}_N) = 0$$

# Consistency of time discretization

## Simplified Formulation

$$\min \int_0^{t_f} g(\mathbf{x}(t), \mathbf{u}(t)) dt$$

**(OCP)**  $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), t \in [0, t_f]$

$$\mathbf{x}(0) = \mathbf{x}_0$$

## Pontryagin's Minimum Principle (PMP)

Recall that the necessary optimality conditions for (OCP) are given by the following expressions

- Co-state equation:

$$\dot{\mathbf{p}}(t) = -\frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}(t), \mathbf{u}(t))' \mathbf{p}(t) - \frac{\partial g}{\partial \mathbf{x}}(\mathbf{x}(t), \mathbf{u}(t))$$

- Control equation:

$$\frac{\partial \mathbf{f}}{\partial \mathbf{u}}(\mathbf{x}(t), \mathbf{u}(t))' \mathbf{p}(t) + \frac{\partial g}{\partial \mathbf{u}}(\mathbf{x}(t), \mathbf{u}(t)) = \mathbf{0}$$

# Consistency of time discretization

Simplified Formulation

Related non-linear program (NLOP)

After discretization in time:

$$\min \int_0^{t_f} g(\mathbf{x}(t), \mathbf{u}(t)) \, dt$$

**(OCP)**

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \quad t \in [0, t_f]$$
$$\mathbf{x}(0) = \mathbf{x}_0$$

$$\min_{(\mathbf{x}_i, \mathbf{u}_i)} \sum_{i=0}^{N-1} h_i g(\mathbf{x}_i, \mathbf{u}_i) \quad \textbf{(NLOP)}$$

$$\mathbf{x}_i + h_i \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i) - \mathbf{x}_{i+1} = \mathbf{0}, \quad i = 0, \dots, N-1$$

# Consistency of time discretization

KKT Related to (NLOP)

Related non-linear program (NLOP)

Denote the Lagrangian related to **(NLOP)** as

After discretization in time:

$$\mathcal{L} = \sum_{i=0}^{N-1} h_i g(\mathbf{x}_i, \mathbf{u}_i) + \sum_{i=0}^{N-1} \lambda'_i (\mathbf{x}_i + h_i \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i) - \mathbf{x}_{i+1})$$

$$\min_{(\mathbf{x}_i, \mathbf{u}_i)} \sum_{i=0}^{N-1} h_i g(\mathbf{x}_i, \mathbf{u}_i) \quad \textbf{(NLOP)}$$

Then, the KKT conditions related to **(NLOP)** read as:

- Derivative w.r.t.  $\mathbf{x}_i$  :

$$h_i \frac{\partial g}{\partial \mathbf{x}_i}(\mathbf{x}_i, \mathbf{u}_i) - \lambda_i + \lambda_{i-1} - \lambda'_i h_i \frac{\partial \mathbf{f}}{\partial \mathbf{x}_i}(\mathbf{x}_i, \mathbf{u}_i) = \mathbf{0}$$

- Derivative w.r.t.  $\mathbf{u}_i$  :

$$h_i \frac{\partial g}{\partial \mathbf{u}_i}(\mathbf{x}_i, \mathbf{u}_i) - \lambda'_i h_i \frac{\partial \mathbf{f}}{\partial \mathbf{u}_i}(\mathbf{x}_i, \mathbf{u}_i) = \mathbf{0}$$

$$\mathbf{x}_i + h_i \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i) - \mathbf{x}_{i+1} = \mathbf{0}, \quad i = 0, \dots, N-1$$



# Consistency of time discretization

## KKT Related to (NLOP)

Denote the Lagrangian related to **(NLOP)** as

$$\mathcal{L} = \sum_{i=0}^{N-1} h_i g(\mathbf{x}_i, \mathbf{u}_i) + \sum_{i=0}^{N-1} \lambda_i' (\mathbf{x}_i + h_i \mathbf{f}(\mathbf{x}_i, \mathbf{u}_i) - \mathbf{x}_{i+1})$$

Then, the KKT conditions related to **(NLOP)** read as:

- Derivative w.r.t.  $\mathbf{x}_i$  :

$$h_i \frac{\partial g}{\partial \mathbf{x}_i}(\mathbf{x}_i, \mathbf{u}_i) + \lambda_i - \lambda_{i-1} + h_i \frac{\partial \mathbf{f}}{\partial \mathbf{x}_i}(\mathbf{x}_i, \mathbf{u}_i)' \lambda_i = \mathbf{0}$$

- Derivative w.r.t.  $\mathbf{u}_i$  :

$$h_i \frac{\partial g}{\partial \mathbf{u}_i}(\mathbf{x}_i, \mathbf{u}_i) + h_i \frac{\partial \mathbf{f}}{\partial \mathbf{u}_i}(\mathbf{x}_i, \mathbf{u}_i)' \lambda_i = \mathbf{0}$$

## Consistency with the PMP

We finally obtain:

$$\begin{aligned} \frac{\lambda_i - \lambda_{i-1}}{h_i} &= - \frac{\partial \mathbf{f}}{\partial \mathbf{x}_i}(\mathbf{x}_i, \mathbf{u}_i)' \lambda_i - \frac{\partial g}{\partial \mathbf{x}_i}(\mathbf{x}_i, \mathbf{u}_i) \\ \frac{\partial \mathbf{f}}{\partial \mathbf{u}_i}(\mathbf{x}_i, \mathbf{u}_i)' \lambda_i + \frac{\partial g}{\partial \mathbf{u}_i}(\mathbf{x}_i, \mathbf{u}_i) &= \mathbf{0} \end{aligned}$$

Let  $\mathbf{p}(t) = \lambda_i$  for  $t \in [t_i, t_{i+1}]$ ,  $i = 0, \dots, N-1$  and  $\mathbf{p}(0) = \lambda_0$ . Then, the equations above are the discretized version of the necessary conditions for **(OCP)**:

$$\begin{aligned} \dot{\mathbf{p}}(t) &= - \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}(t), \mathbf{u}(t))' \mathbf{p}(t) - \frac{\partial g}{\partial \mathbf{x}}(\mathbf{x}(t), \mathbf{u}(t)) \\ \frac{\partial \mathbf{f}}{\partial \mathbf{u}}(\mathbf{x}(t), \mathbf{u}(t))' \mathbf{p}(t) + \frac{\partial g}{\partial \mathbf{u}}(\mathbf{x}(t), \mathbf{u}(t)) &= \mathbf{0} \end{aligned}$$

# Solution approaches:

1. state and control parameterization methods
2. control parameterization methods

# Example: Zermelo's Problem

- Designing direct methods in Matlab: transcribe optimal control problem into a non-linear program, and solve it via fmincon

Modified Zermelo's Problem

State and control parameterization method

**(OCP)**

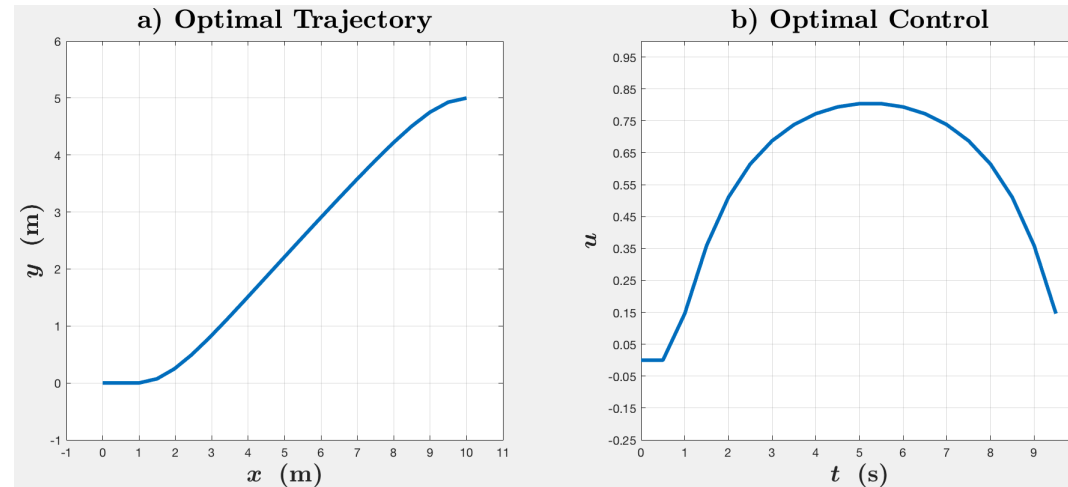
$$\begin{aligned} \min \quad & \int_0^{t_f} u(t)^2 dt \\ \dot{x}(t) = & v \cos(u(t)) + \text{flow}(y(t)), t \in [0, t_f] \\ \dot{y}(t) = & v \sin(u(t)), t \in [0, t_f] \\ (x, y)(0) = & 0, (x, y)(t_f) = (M, \ell) \\ |u(t)| \leq & 1, t \in [0, t_f] \end{aligned}$$



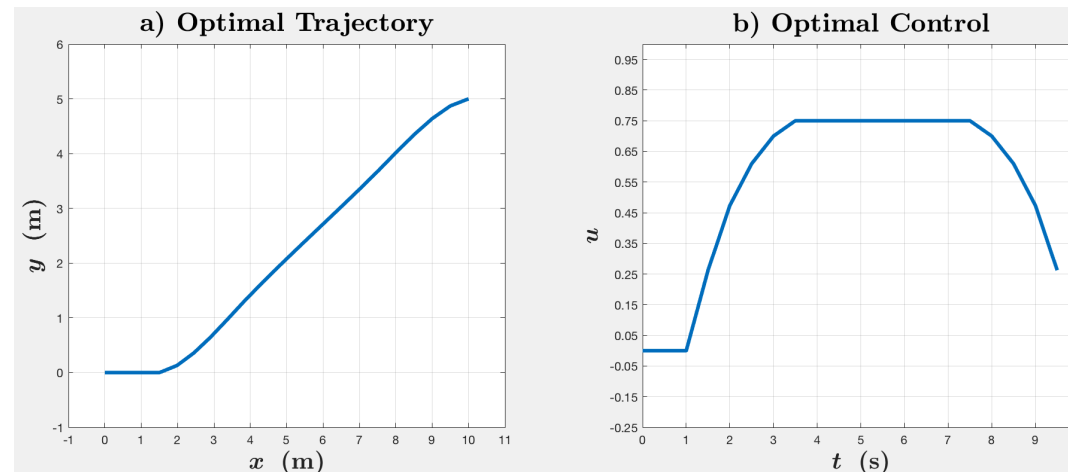
**(NLOP)**

$$\begin{aligned} \min_{(x_i, u_i)} \quad & \sum_{i=0}^{N-1} u_i^2 \\ x_{i+1} = & x_i + h(v \cos(u_i) + \text{flow}(y_i)) \\ y_{i+1} = & y_i + h v \sin(u_i), |u_i| \leq u_{\max} \\ (x_0, y_0) = & 0, (x_N, y_N) = (M, \ell) \end{aligned}$$

# Results



$|u(t)| \leq 1$   
 $N = 20$   
28 iterations



$|u(t)| \leq 0.75$   
 $N = 20$   
23 iterations

# Transcription into nonlinear programming (control parametrization method)

## Time and control discretization

$$\min \int_0^{t_f} g(\mathbf{x}(t), \mathbf{u}(t), t) dt$$

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t), \quad t \in [0, t_f]$$

**(OCP)**

$$\mathbf{x}(0) = \mathbf{x}_0, \quad \mathbf{x}(t_f) \in M_f$$

$$\mathbf{u}(t) \in U \subseteq \mathbb{R}^m, \quad t \in [0, t_f]$$

1. Select a discretization  $0 = t_0 < t_1 < \dots < t_N = t_f$  for the interval  $[0, t_f]$  and, for every  $i = 0, \dots, N - 1$ , define

$$\mathbf{u}_i \sim \mathbf{u}(t), \quad t \in (t_i, t_{i+1}]$$

2. By denoting  $h_i = t_{i+1} - t_i$ , **(OCP)** is transcribed into the following nonlinear, constrained optimization problem

$$\begin{aligned} & \min_{\mathbf{u}_i} \sum_{i=0}^{N-1} h_i g(\mathbf{x}(t_i), \mathbf{u}_i, t_i) \\ & \mathbf{u}_i \in U, i = 0, \dots, N - 1, \quad F(\mathbf{x}(t_N)) = 0 \end{aligned}$$

**(NLOP-C)**

where each  $\mathbf{x}(t_i)$  is recursively computed via  
 $\mathbf{x}(t_{i+1}) = \mathbf{x}(t_i) + h_i \mathbf{f}(\mathbf{x}(t_i), \mathbf{u}_i, t_i), i = 0, \dots, N - 1$

# Example: Zermelo's Problem

Modified Zermelo's Problem

**(OCP)**

$$\begin{aligned} \min \quad & \int_0^{t_f} u(t)^2 dt \\ \dot{x}(t) = & v \cos(u(t)) + \text{flow}(y(t)), t \in [0, t_f] \\ \dot{y}(t) = & v \sin(u(t)), t \in [0, t_f] \\ (x, y)(0) = & 0, (x, y)(t_f) = (M, \ell) \\ |u(t)| \leq & 1, t \in [0, t_f] \end{aligned}$$



Control parameterization method

$$\min_{u_i} \sum_{i=0}^{N-1} u_i^2 \quad \textbf{(NLOP-C)}$$

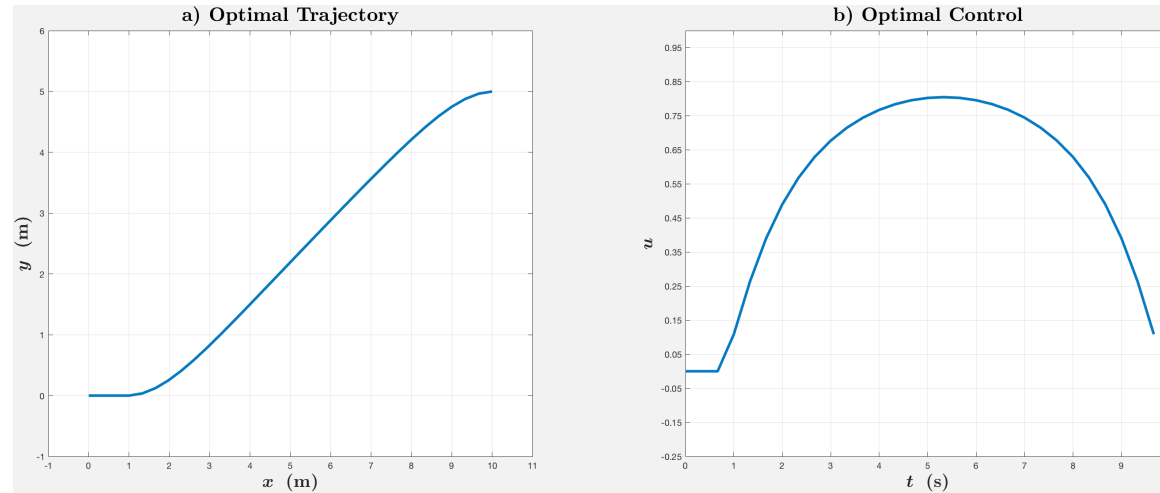
$$(x, y)(t_N) = (M, \ell), \quad |u_i| \leq u_{max}$$

where, recursively:

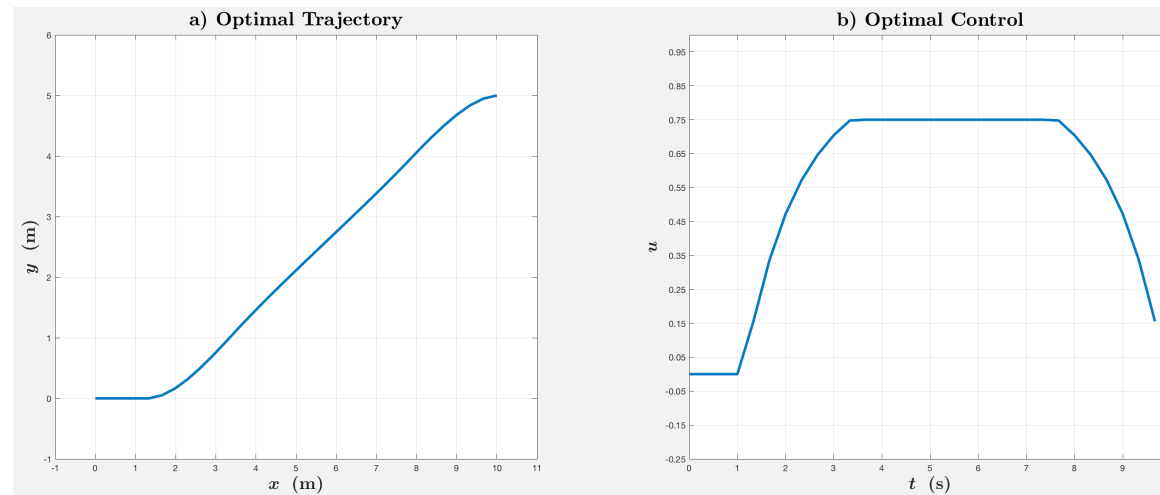
$$x(t_N) = x_0 + h_i \sum_{i=0}^{N-1} (v \cos(u_i) + \text{flow}(y(t_i)))$$

$$y(t_i) = y_0 + h_i \sum_{j=0}^i v \sin(u_j)$$

# Results



$|u(t)| \leq 1$   
 $N = 30$   
50 iterations



$|u(t)| \leq 0.75$   
 $N = 30$   
16 iterations

# Next time

- Direct collocation and SCP