Oral presentation | Data science and AI

Data science and AI-II

Thu. Jul 18, 2024 10:45 AM - 12:45 PM Room C

[10-C-01] Data-driven optimal control of self-propelled undulatory swimmers

*Karl Maroun[1], Michel Bergmann[2], Philippe Traoré[1] （1. Université de Poitiers, 2. INRIA Bordeaux Sud-Ouest）

Keywords: Computational Fluid Dynamics, System Identification, Optimal Control, Model Predictive Control, Undulatory swimming

Data-driven trajectory optimization of self-propelled undulatory swimming

Karl Maroun
Supervisors:
Michel Bergmann (Inria Bordeaux)
Philippe Traoré (Institut P’)

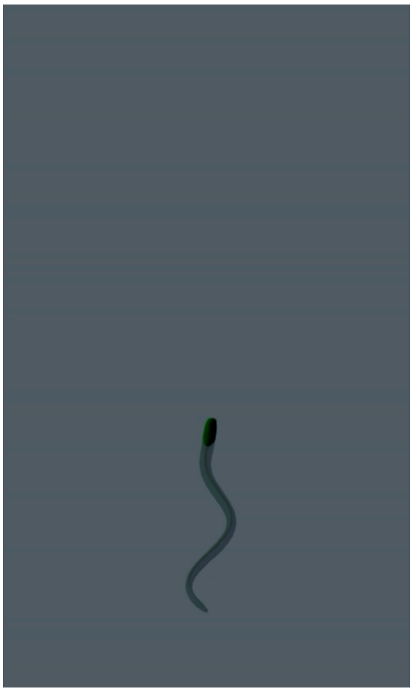
Outline

- Context & Objectives
- Numerical modelling
- Optimal control
- System identification
- Velocity tracking
- Cost of transport minimization
- Future work & challenges

Self-propelled undulatory swimmers

- Aquatic animals that propel themselves through the water by deforming their spines and propagating deformation waves through the body

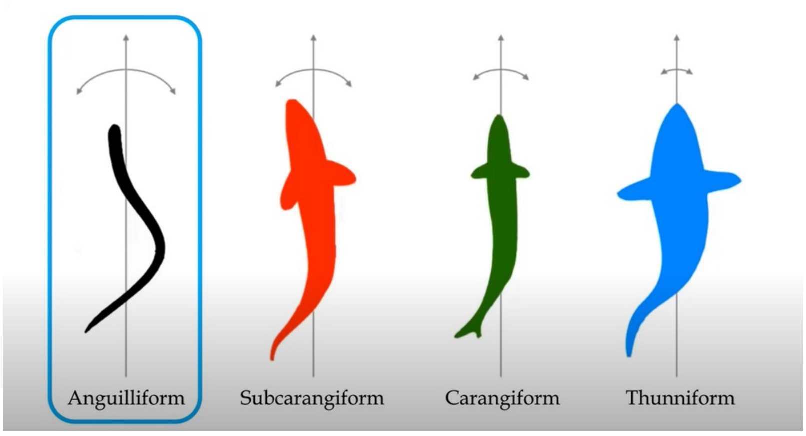
- 3D simulation of snake swimming



Undulatory swimming modes

Anguilliform:

- Entire body participates in the waveform
- Amplitude relatively large along the entire length
- e.g: Eels, snakes, tadpoles



Connaboy, Chris, Simon Coleman, and Ross H. Sanders. 2009.

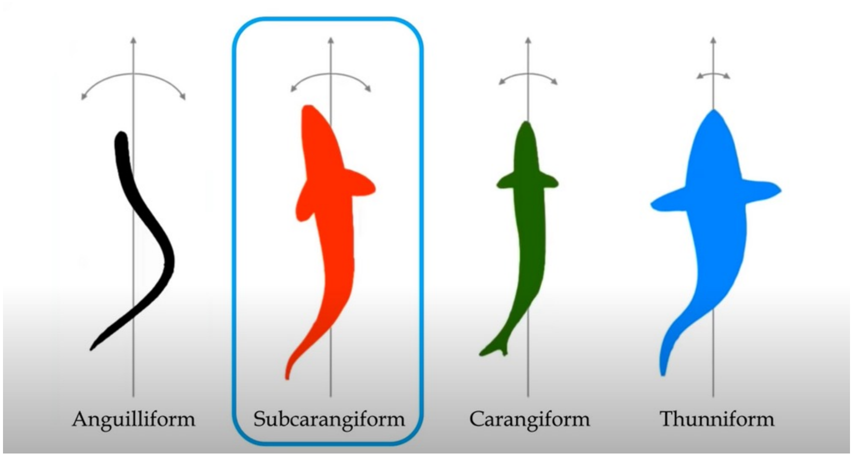
Undulatory swimming modes

Carangiform:
- e.g: Salmon

Subcarangiform:
- e.g: Trout

Thunniform:
- e.g: Tuna

Undulation confined in the posterior regions of the body



Connaboy, Chris, Simon Coleman, and Ross H. Sanders. 2009.

Objectives

- Present the numerical methods used to simulate undulatory swimmers
- Present a simple framework to solve trajectory optimization problems applied to undulatory swimming
- **Trajectory optimization:** process of designing a **trajectory** that **minimizes** (or maximizes) some **measure of performance** while satisfying a set of constraints.
- E.g: find the swimming kinematics that minimizes the energy spent moving from point A to point B

Undulatory swimming kinematics

- For all swimming modes the 2D kinematics of the backbone can be approximated by a backward travelling wave:

$$y(x,t) = a(x)\sin(kx - \omega t)$$

Kinematic parameters:

- Wavenumber: $k = \frac{2\pi}{\lambda}$ (wavelength λ)
- Angular frequency: $\omega = 2\pi f$ (frequency f)
- Curve envelope : $a(x) = c_0 + c_1 x + c_2 x^2$ (depends on the swimming mode)

Carangiform

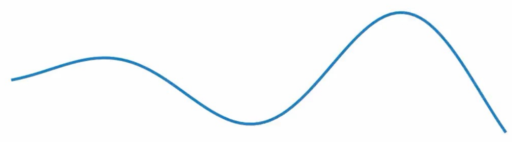
$$f = 2\ Hz$$

$$a(x) = 0.02 - 0.12x + 0.2x^2$$

Anguilliform

$$f = 2\ Hz$$

$$a(x) = x + 0.1$$



Swimming kinematics for control

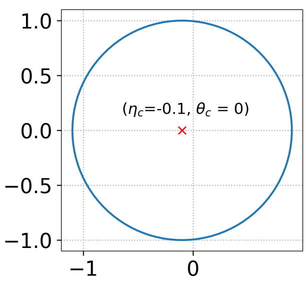
- 2 control parameters added to swimming law:

$$y(x,t) = b(t)a(x)\sin[kx - \phi(t)]$$

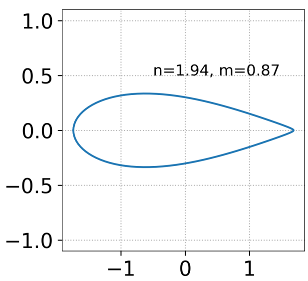
- Amplitude gain, $b(t)$: to allow the swimmer to modify its amplitude

- Instantaneous phase, $\phi(t) = 2\pi \int_{t_0}^{t} f(t)\, dt$: to allow frequency modulation

2D swimmer shape

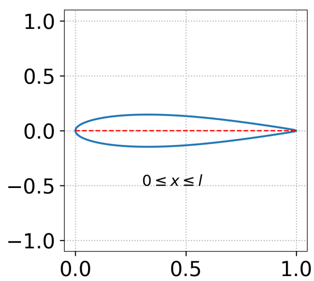


(a)



(b)

$b = 0.87$



(c)

Karman trefftz transform

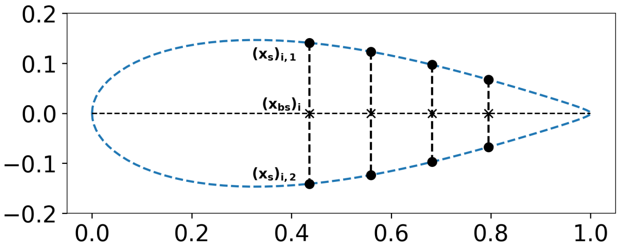
$$z = nb\frac{(\zeta + m)^n + (\zeta - m)^n}{(\zeta + m)^n - (\zeta - m)^n}$$

$$z = xi + y \qquad \zeta = \eta i + \theta$$

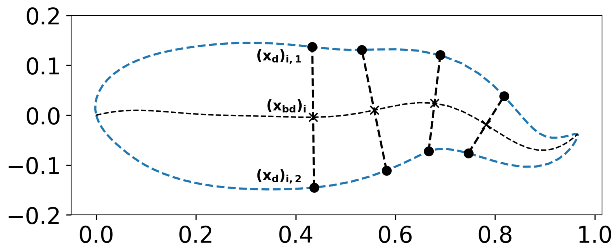
Shape defined by the following parameters:

$\eta_c, \theta_c, n, b, m, l$

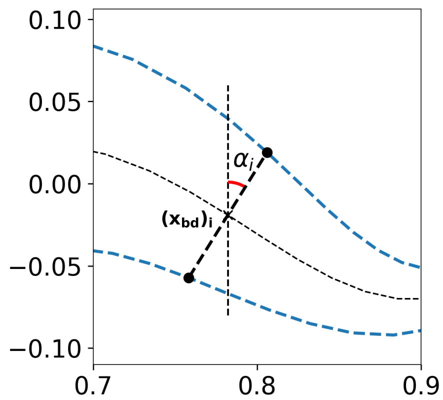
2D deformation



(a)



(b)



α: Local deformation angle

$$\vec{x}_d(t) = R\big(\alpha(t)\big)(\vec{x}_s - \vec{x}_{bs}) + \vec{x}_b(t)$$

$$R(\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix}$$

High-fidelity numerical simulation

- Computational fluid dynamics
- Incompressible flow
- Volume penalization method to account for body
- Uniform Cartesian grid / Finite difference / Chorin projection method

$$\begin{cases} \dfrac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\vec{u} = -\dfrac{1}{\rho}\nabla \mathrm{p} + \nu \Delta \vec{u} + \dfrac{\chi(\vec{x}, t)}{k}\left(\vec{u} - \vec{u}_{body}\right) \\ \nabla \cdot \vec{u} = 0 \end{cases}$$

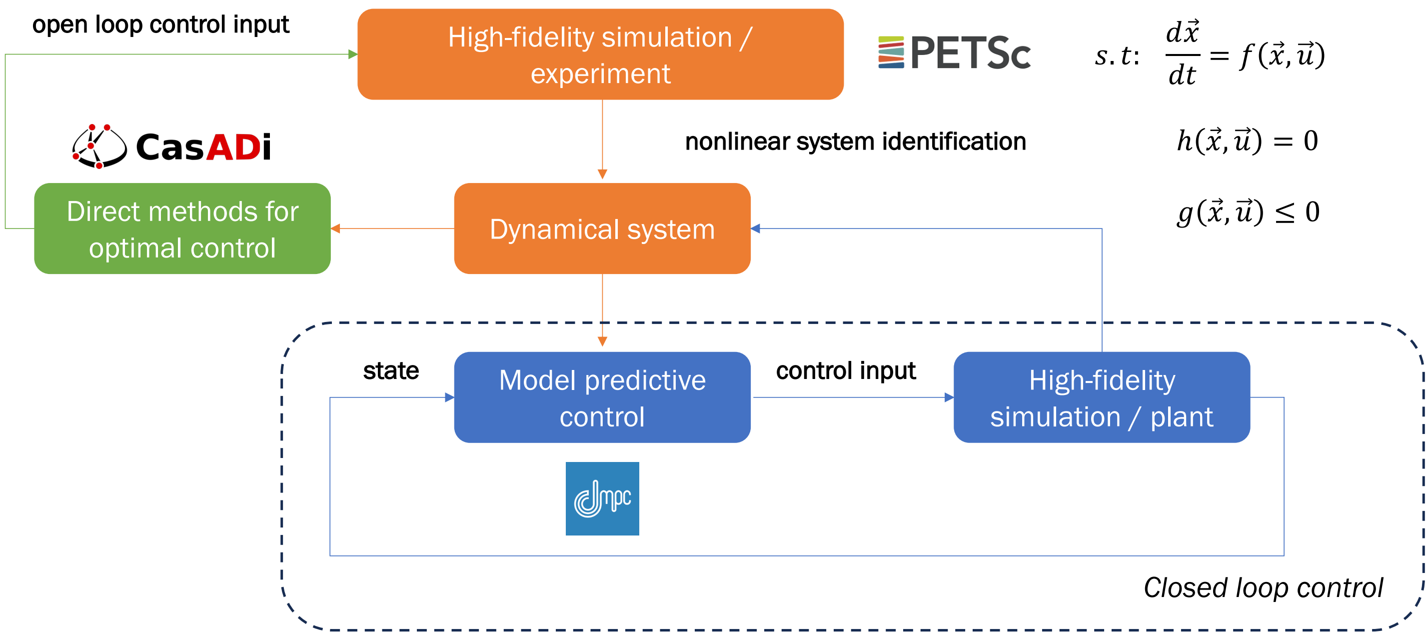
Trajectory optimization strategy

$$\min_{\vec{u}(t)} J(\vec{x},\vec{u}) = \int_{t_0}^{t_f} l(\vec{x},\vec{u})\,dt + \psi(\vec{x})$$

$$s.t: \quad \frac{d\vec{x}}{dt} = f(\vec{x},\vec{u})$$

$$h(\vec{x},\vec{u}) = 0$$

$$g(\vec{x},\vec{u}) \leq 0$$



Optimal control problem

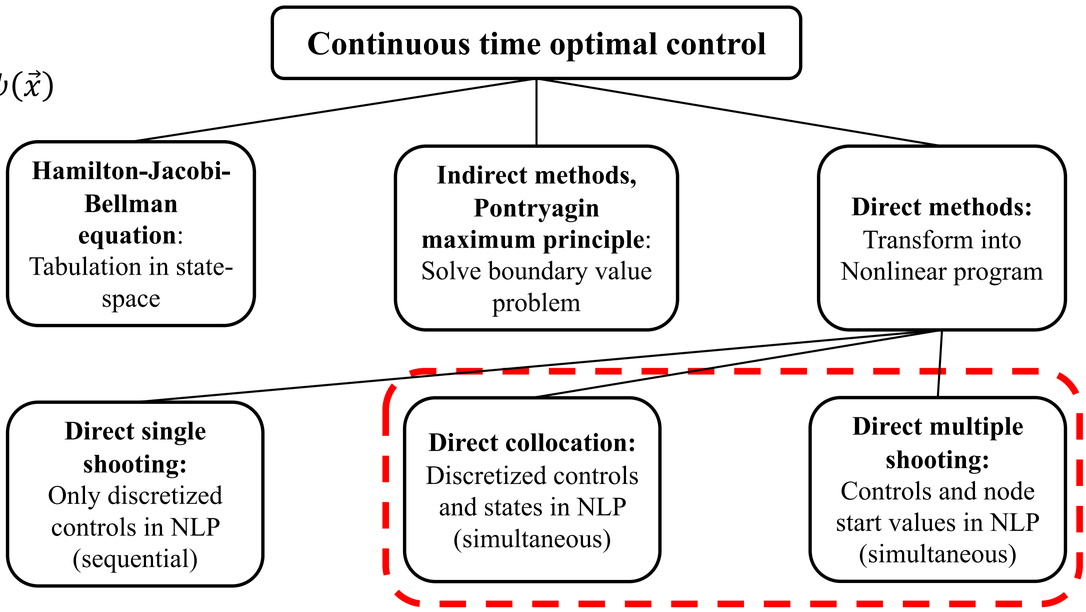
$$\min_{\vec{u}(t)} J(\vec{x},\vec{u}) = \int_{t_0}^{t_f} l(\vec{x},\vec{u})\,dt + \psi(\vec{x})$$

$$s.t{:}\quad \frac{d\vec{x}}{dt} = f(\vec{x},\vec{u})$$

$$\vec{x}(t = t_0) = \vec{x}_0$$
$$\vec{x}(t = t_f) = \vec{x}_F$$

$$\vec{x}_{lb} \le \vec{x}(t) \le \vec{x}_{ub}$$
$$\vec{u}_{lb} \le \vec{u}(t) \le \vec{u}_{ub}$$

$$h(\vec{x},\vec{u}) = 0$$
$$g(\vec{x},\vec{u}) \le 0$$



Direct multiple shooting

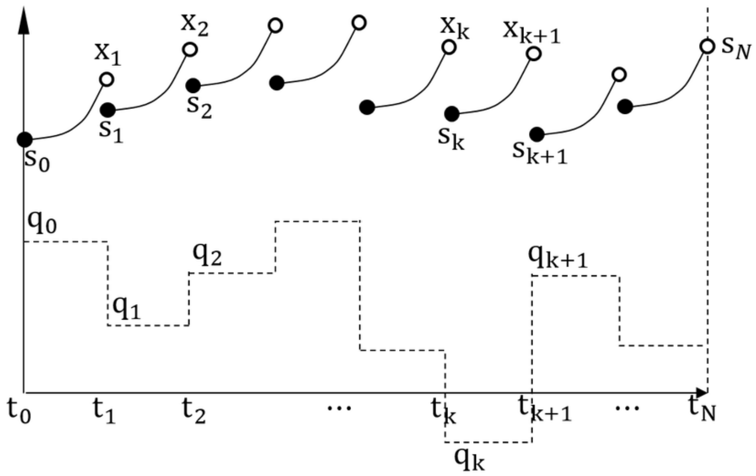
NLP:

$$\min_{s_i, q_i} \sum_{i=0}^{N-1} l_i(s_i, q_i) + \psi(s_N)$$

$$s.t: \; s_0 = x_0$$
$$s_N = x_F$$
$$s_{i+1} = \underbrace{F(s_i, q_i)}_{RK\ 4/5}$$

$$x_i^{lb} \le s_i \le x_i^{ub}$$
$$u_i^{lb} \le q_i \le u_i^{ub}$$

$$h(s_i, q_i) \le 0$$
$$g(s_i, q_i) = 0$$



Optimization problem solved with an interior point method (ipopt library)

Model predictive control

$$\min_{\vec{u}(t)} J(\vec{x},\vec{u}) = \int_{t_k}^{t_k+\Delta t_h} l(\vec{x},\vec{u})\,dt + V(\vec{x})$$

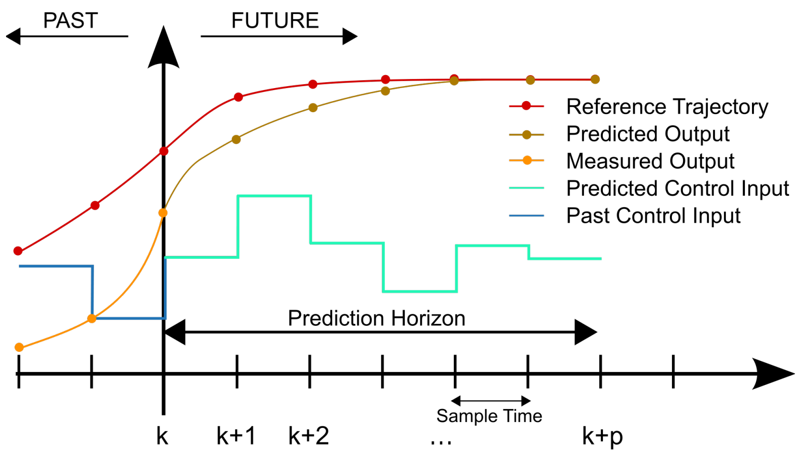
$$s.t: \quad \frac{d\vec{x}}{dt} = f(\vec{x},\vec{u})$$

$$\vec{x}(t = t_k) = (\vec{x}_k)_{measured}$$

$$\vec{x}_{lb} \leq \vec{x}(t) \leq \vec{x}_{ub}$$
$$\vec{u}_{lb} \leq \vec{u}(t) \leq \vec{u}_{ub}$$

$$h(\vec{x},\vec{u}) = 0$$
$$g(\vec{x},\vec{u}) \leq 0$$

$\xrightarrow{\text{yields}}$ $u^*(\vec{x})$: **Closed loop solution**



Δt_h: **Prediction horizon**

Sparse identification of nonlinear dynamics

$$\mathcal{D} := \left\{\left(\vec{x}_i, \frac{d\vec{x}_i}{dt}, \vec{u}_i\right)\right\}_{i=1}^{n_s}$$

$$\boldsymbol{X} = [\vec{x}_1 \quad \cdots \quad \vec{x}_{n_s}] \qquad \boldsymbol{Y} = [\vec{u}_1 \quad \cdots \quad \vec{u}_{n_s}]$$

$$\dot{\boldsymbol{X}} = \left[\frac{d\vec{x}_1}{dt} \quad \cdots \quad \frac{d\vec{x}_{n_s}}{dt}\right]$$

Data

$$\frac{d\vec{x}}{dt} = f(\vec{x}, \vec{u}) \Rightarrow f(\vec{x}, \vec{u}) \approx \boldsymbol{\Xi}\vec{\Theta}(\vec{x}, \vec{u})$$

$$\vec{\Theta}(\vec{x}, \vec{u}) = \begin{bmatrix} \theta_1(\vec{x}, \vec{u}) \\ \vdots \\ \theta_{n_f}(\vec{x}, \vec{u}) \end{bmatrix} \qquad \boldsymbol{\Xi} = \begin{bmatrix} \vdots \\ \vec{\xi}_k \\ \vdots \end{bmatrix}$$

Library of candidate basis functions

$$\min_{\vec{\xi}_k} \left\| \dot{\boldsymbol{X}}_k - \vec{\xi}_k \vec{\Theta}(\boldsymbol{X}, \boldsymbol{Y}) \right\|_2 + R(\vec{\xi}_k)$$

Sparse optimization

$$\vec{\Theta}(\vec{x}, \vec{u}) = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1 x_2 \\ u \end{bmatrix} \qquad f(\vec{x}, \vec{u}) \approx \begin{cases} a_1 + a_2 x_1 + a_3 x_2 + a_4 x_1 x_2 + a_5 u \\ b_1 + b_2 x_1 + b_3 x_2 + b_4 x_1 x_2 + b_5 u \end{cases}$$

Brunton, Steven L., Joshua L. Proctor, and J. Nathan Kutz. 2016.

Velocity tracking

OCP solved with MPC:

$$\min_{u_G^{0:N_h},\vec{c}_{0:N_h-1}} \left(u_G^{N_h} - u_{ref}^{N_h}\right)^2 + \sum_{i=0}^{N_h-1} \left(u_G^i - u_{ref}^i\right)^2 + \Delta\vec{c}_i^{\,T} \, R \, \Delta\vec{c}_i$$

s.t:

$$u_G^{i+1} = F\left(u_G^i \,, \vec{c}_i\right)$$

$$u_G^0 = (u_G)_{high\ fidelity}$$

$$\vec{c}_{min} \leq \vec{c}_i \leq \vec{c}_{max}$$

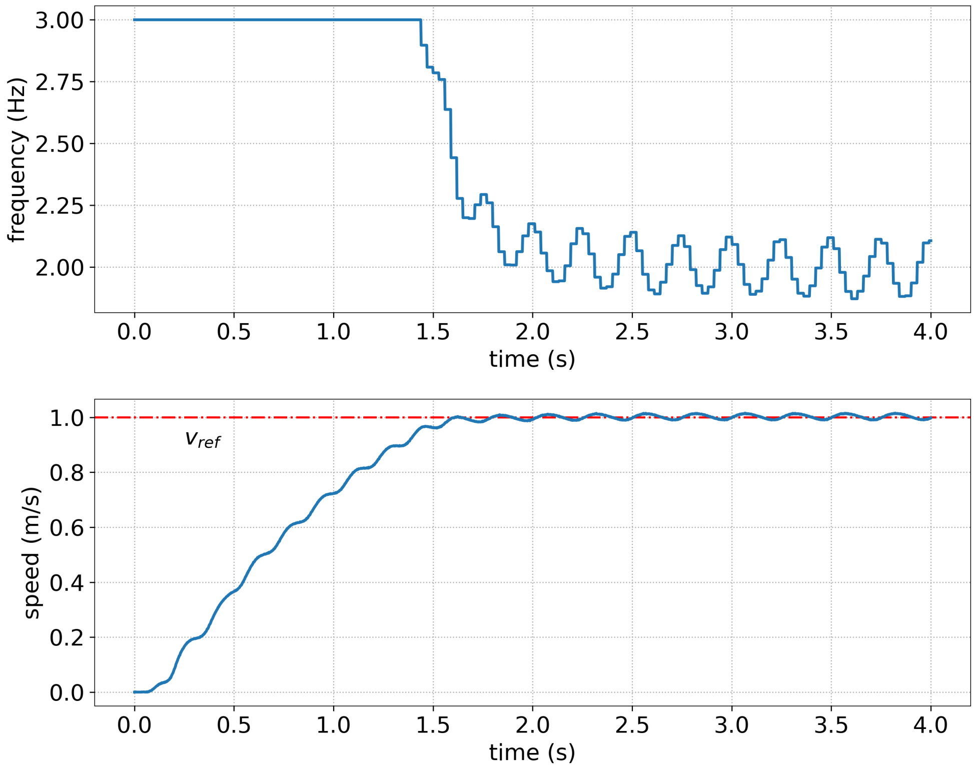
- Data generated by 5 high fidelity simulations of constant frequency

$$f = \{1.0 \quad 1.5 \quad 2.0 \quad 2.5 \quad 3.0\}$$

- SINDy model:

$$\frac{du_G}{dt} = 0.403u_G^2 - 0.095f^2$$

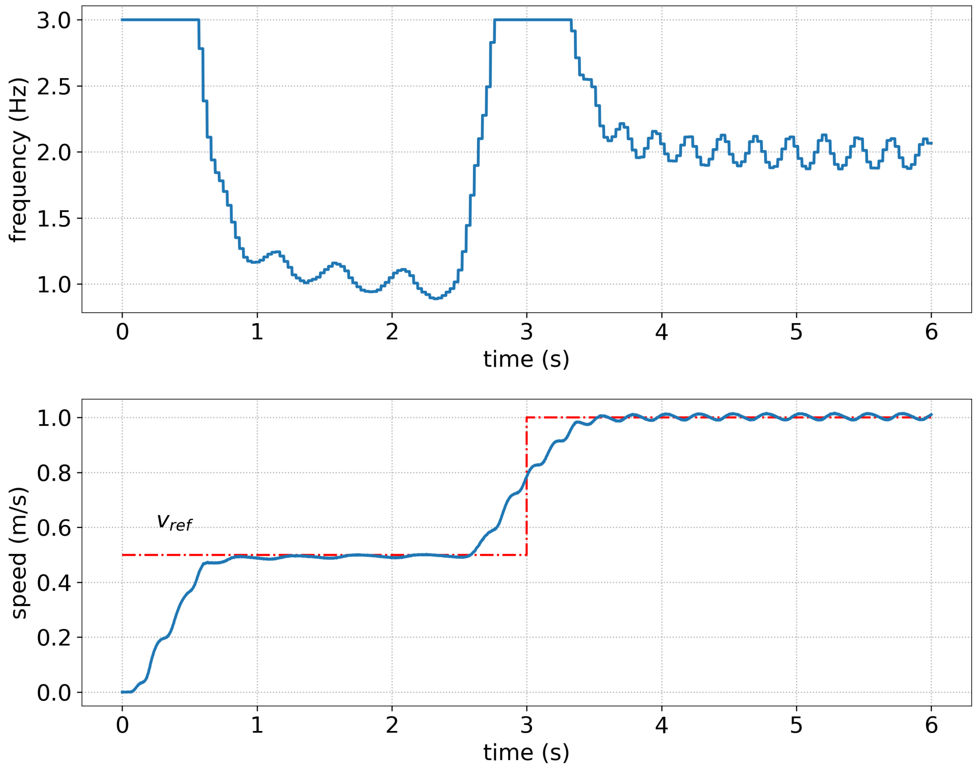
- $u_{ref}(t) = 1.0\, m/s$



- Same sindy model

$$v_{ref}(t) = \begin{cases} 0.5, & t < 3.0 \\ 1.0, & t \geq 3.0 \end{cases}$$

- Model predictive control anticipates the change in reference



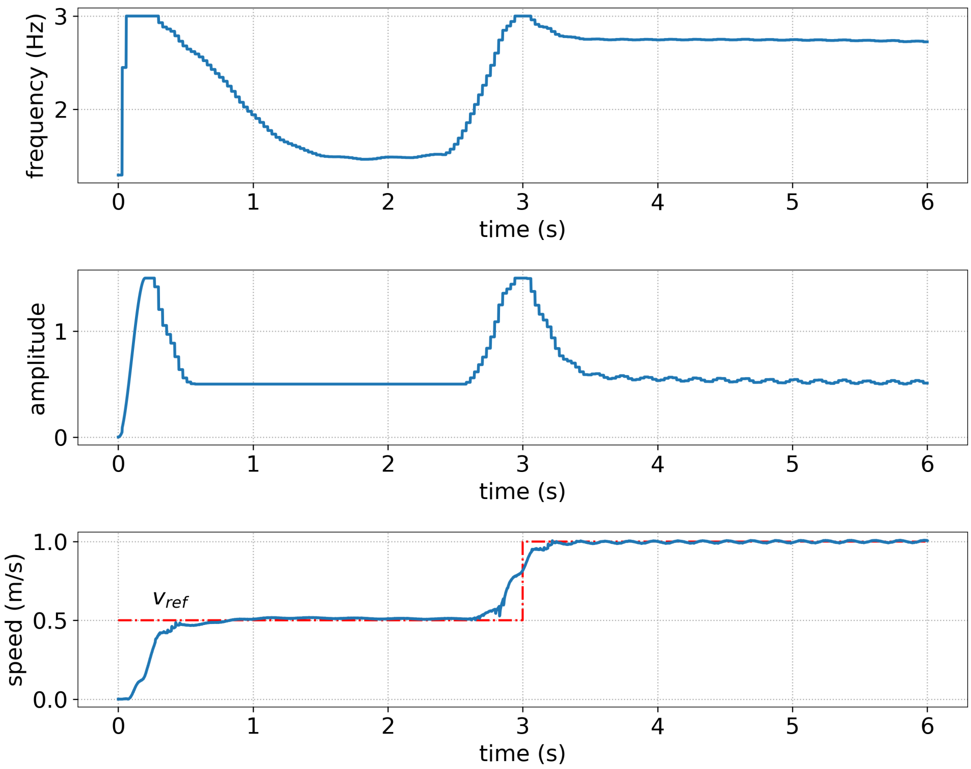
- Data generated by 15 high fidelity simulations of constant frequency

$b = \{0.5 \quad 1.0 \quad 1.5\}$

- SINDy model:

$$\frac{du_g}{dt} = -0.209u_g - 0.276u_G^2 - 0.436u_G^3 - 0.067fb - 0.109f^2b^2 + 0.148u_G fb - 0.261u_G^2 fb - 0.071u_G f^2 b - 0.344u_G fb^2$$

$$u_{ref}(t) = \begin{cases} 0.5, & t < 3.0 \\ 1.0, & t \geq 3.0 \end{cases}$$

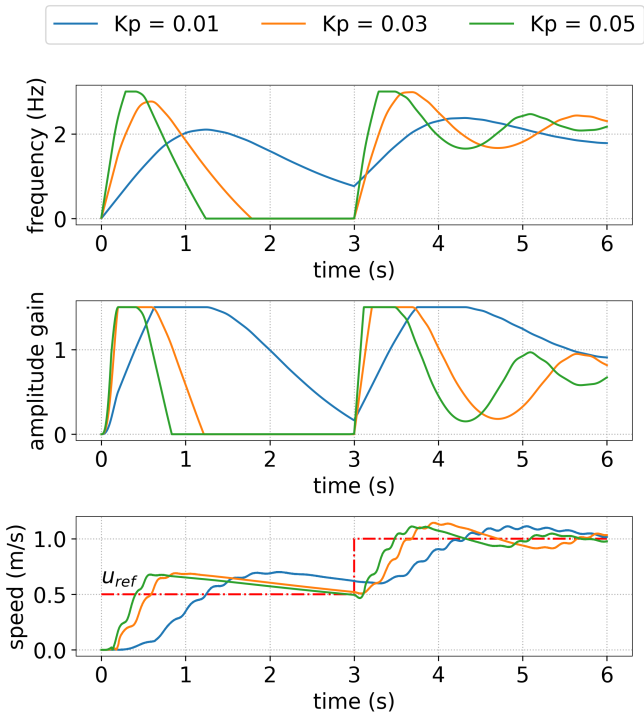


Proportional control

$$f(t_k) = f(t_{k-1}) + K_p e(t_k)$$

$$b(t_k) = b(t_{k-1}) + K_p e(t_k)$$

$$e(t_k) = \left|u_{ref}(t_k)\right| - |u_G(t_k)|$$

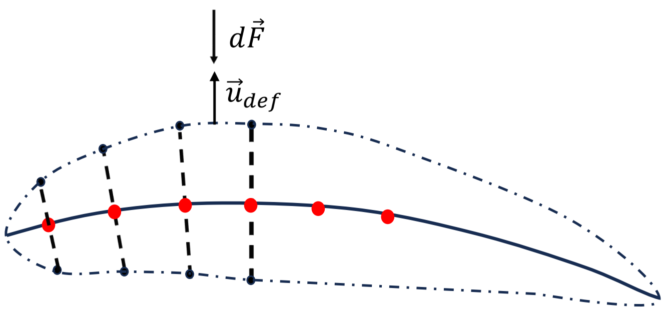


Cost of transport

$$P_{def} = \max\left(-\iint \vec{u}_{def} \cdot d\vec{F}, 0\right)$$

$$CoT(t) = \frac{\int_{t_0}^{t} P_{def}\, dt}{\left|\int_{t_0}^{t} u_G\, dt\right|} = \frac{e(t)}{|x(t) - x(t_0)|}$$

$$d\vec{F} = (-p\mathrm{I} + \tau) \cdot \hat{n}\, dS$$



Cost of transport minimization

OCP:

$$\min_{\vec{s}_{0:N_f},\vec{c}_{0:N_f-1}} w_1 e\left(t_f\right) + \sum_{i=0}^{N_f-1} \Delta\vec{c}_i^T \, R \Delta\vec{c}_i$$

s.t:

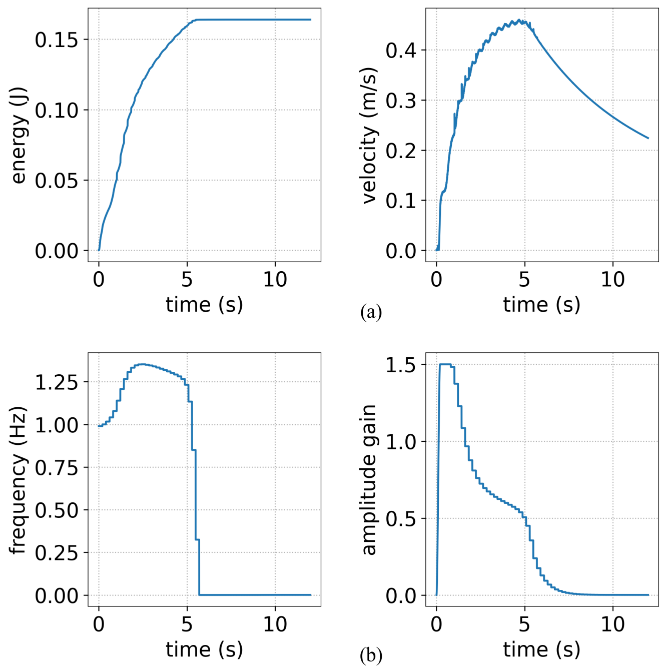
$$\vec{s}_{i+1} = F(\vec{s}_i \,, \vec{c}_i)$$
$$\vec{c}_{min} \leq \vec{c}_i \leq \vec{c}_{max}$$
$$\vec{s}_0 = [0 \quad 0 \quad 0]$$
$$x_{target} - \epsilon \leq x_{N_f} \leq x_{target} + \epsilon$$

Open loop solution

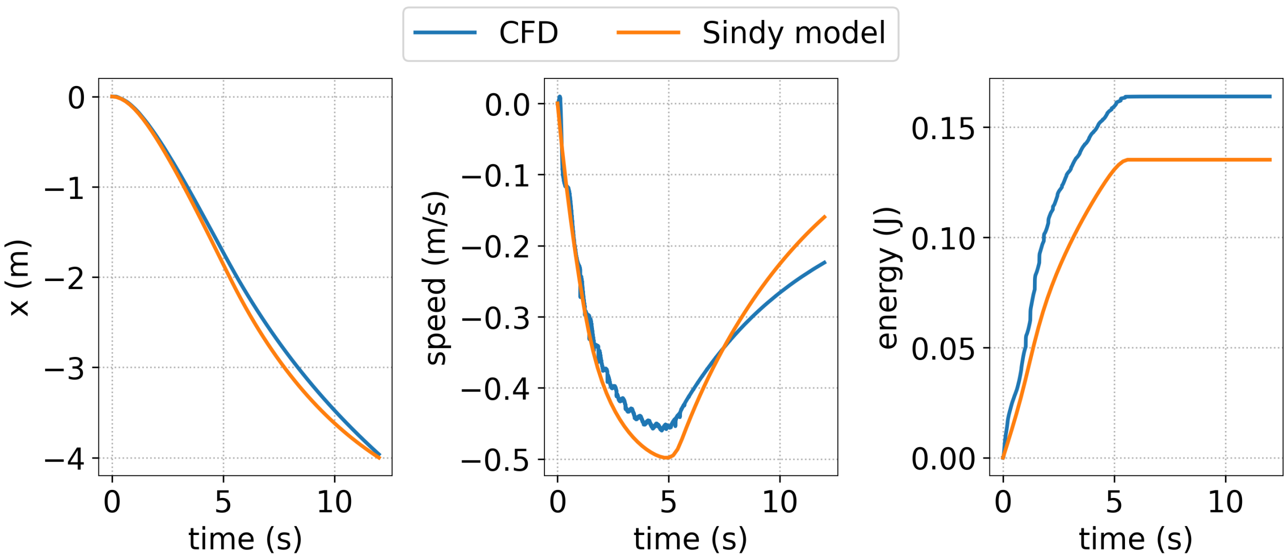
- Same data as previous test case
- SINDy model:

$$\frac{de}{dt} = 0.073fb - 0.068f^2b - 0.050f^2b^2 + 0.017f^3b + 0.031efb + 0.013e^2fb - 0.018ef^2b - 0.018efb^2$$

- $T_f = 12\ s \quad x_{target} = 4$
- Solution like burst and coast swimming



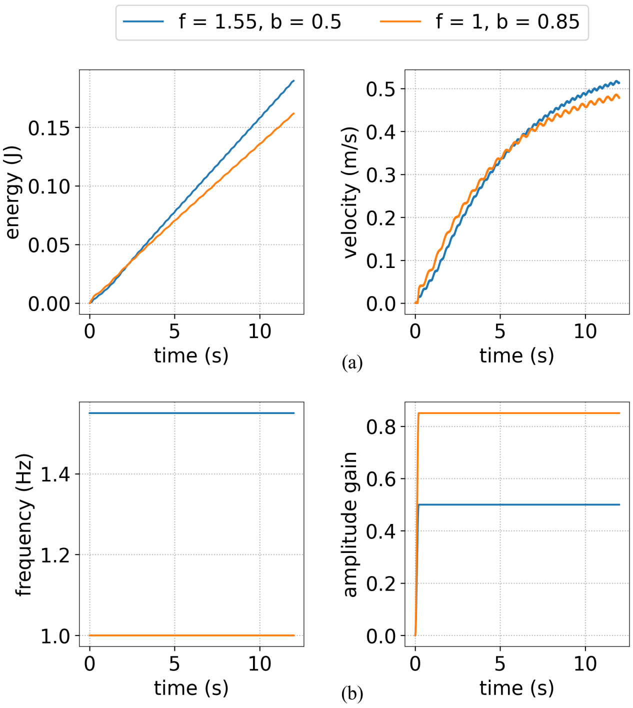
Model mismatch – Open loop control



Comparison with continuous swimming

Cases	Total energy $e(t_f)$ (J)	Terminal position $x(t_f)$ (m)
Burst-and-coast	0.164	3.96
$f = 1.55, b = 0.5$	0.19	3.98
$f = 1, b = 0.85$	0.162	3.96

- Second continuous swimming case slightly lower cost of transport
- Burst and coast solution is an optimum for the optimal control problem with the SINDy model.
- Different than that of the high-fidelity simulation due to model errors



Thank you

Questions ?