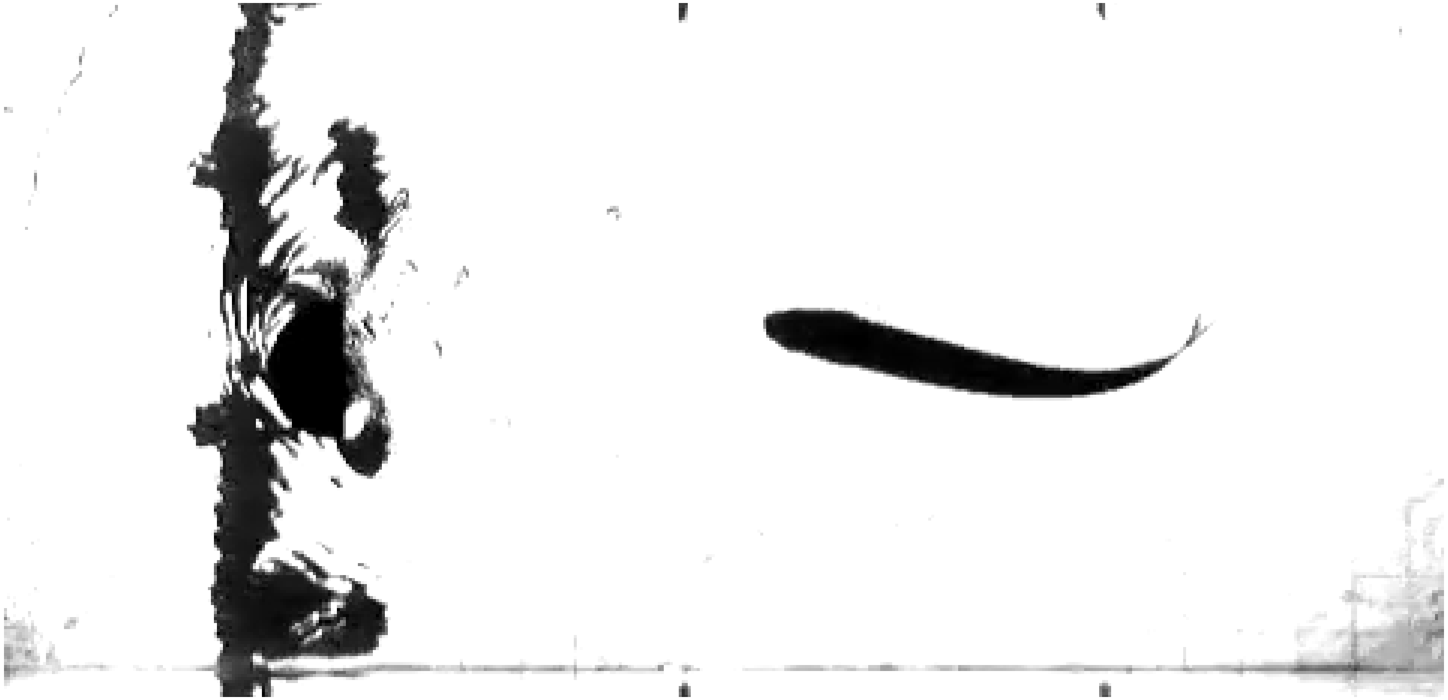
Teaching a Fish to Swim

A hybrid approach for fast fish simulation



Fish gotta swim, bird's gotta fly...

I've gotta sit and wonder why, why, why...

(Beal 2015)

John Z. Zhang | johnz@mit.edu

Outline

Outline

1. Background on Fish Locomotion

Outline

1. Background on Fish Locomotion
2. Differentiable Projective Dynamics

Outline

1. Background on Fish Locomotion
2. Differentiable Projective Dynamics
3. Hybrid Approach to FSI

Outline

1. Background on Fish Locomotion
2. Differentiable Projective Dynamics
3. Hybrid Approach to FSI
4. COMSOL training set

Outline

1. Background on Fish Locomotion
2. Differentiable Projective Dynamics
3. Hybrid Approach to FSI
4. COMSOL training set
5. Preliminary Results

Outline

1. Background on Fish Locomotion
2. Differentiable Projective Dynamics
3. Hybrid Approach to FSI
4. COMSOL training set
5. Preliminary Results
6. Discussion

Outline

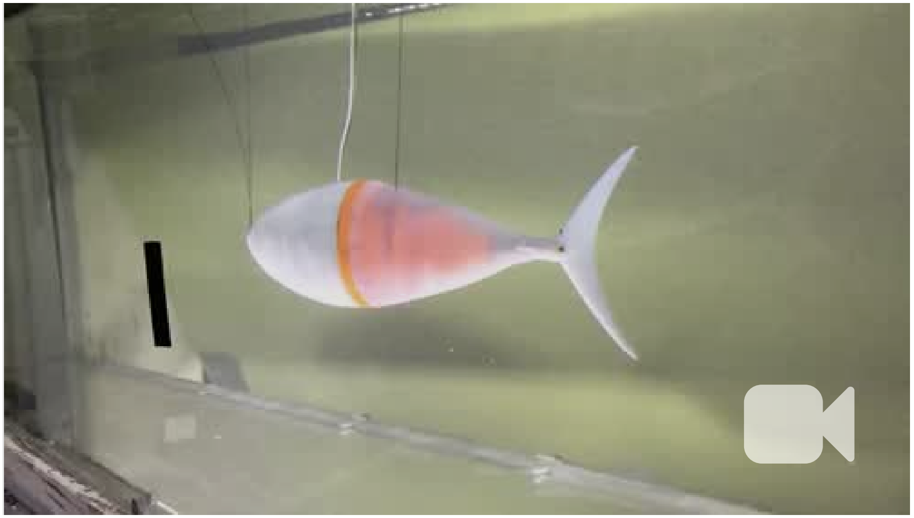
1. Background on Fish Locomotion
2. Differentiable Projective Dynamics
3. Hybrid Approach to FSI
4. COMSOL training set
5. Preliminary Results
6. Discussion
7. Future Directions

Yellowfin Tuna ~2 m

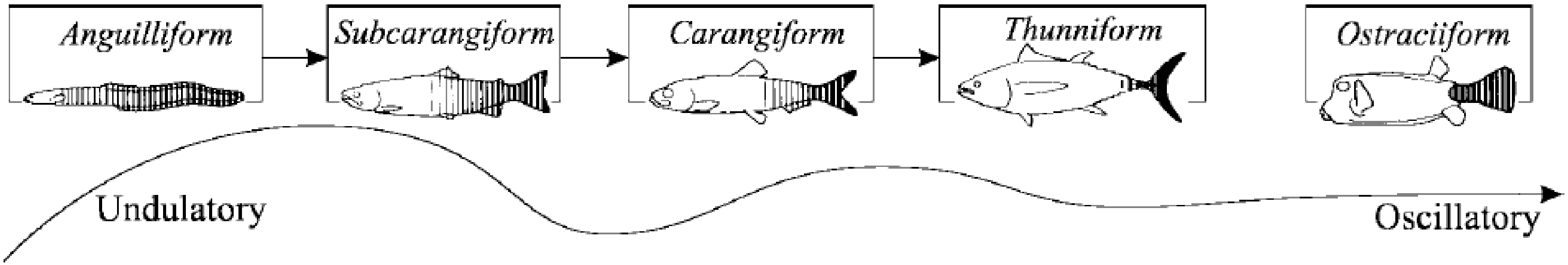
Humpback Whale ~ 15 m

Schooling Fish

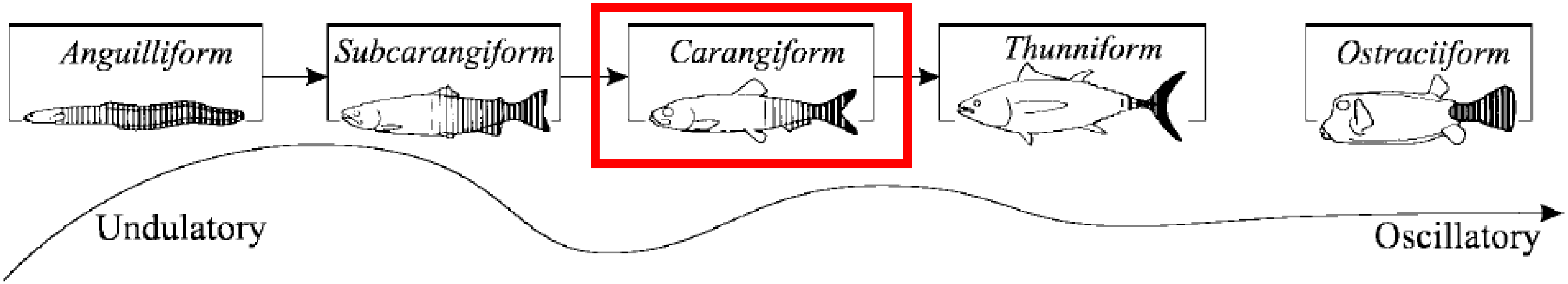
Tunabot 2019

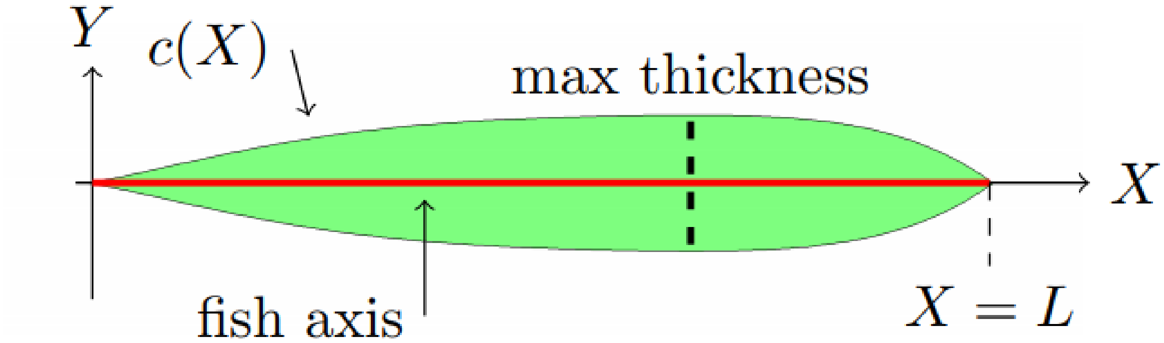


Gifs from *Our Planet* | *High Seas*



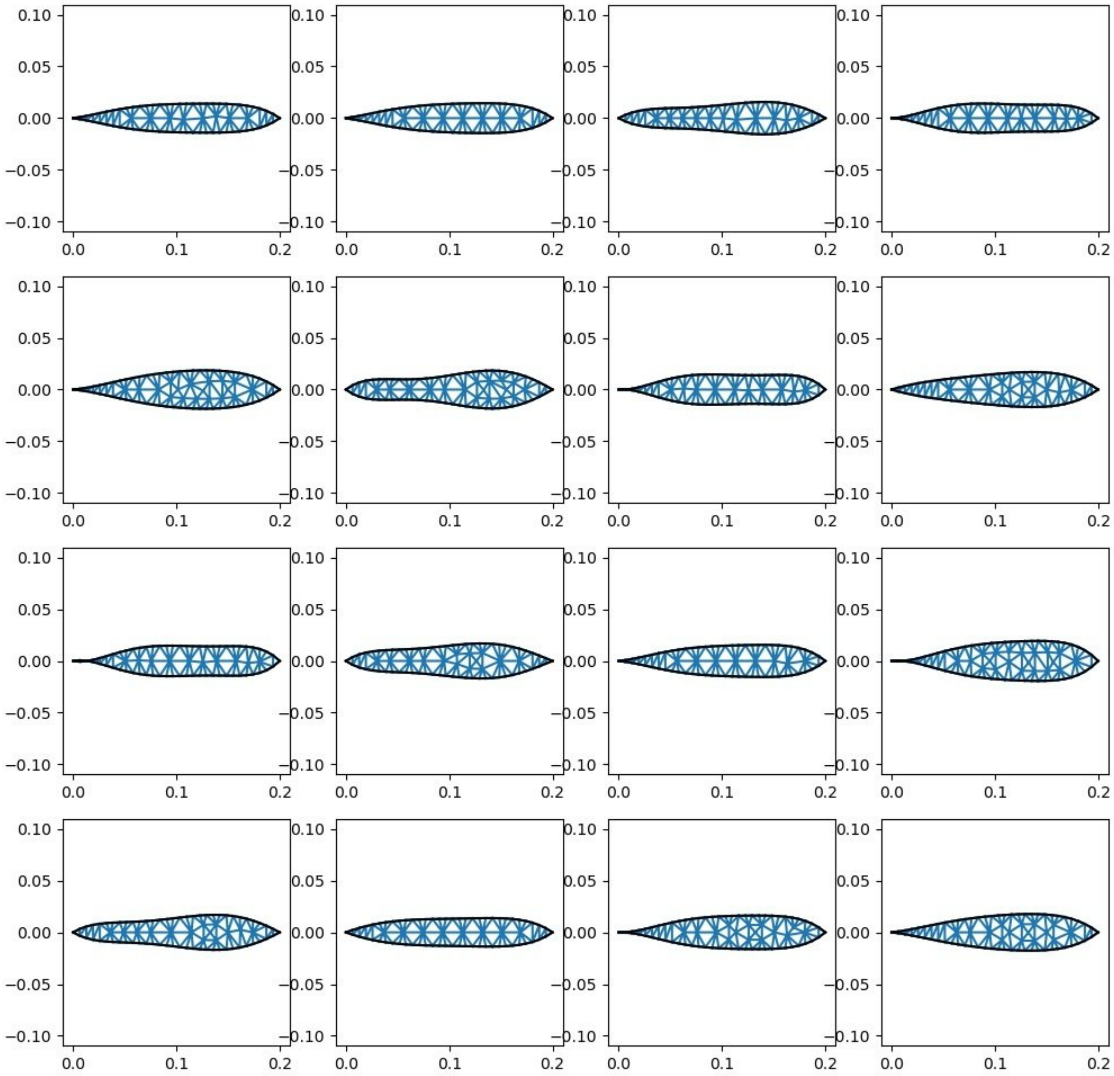
(Curatolo & Teresi 2015)





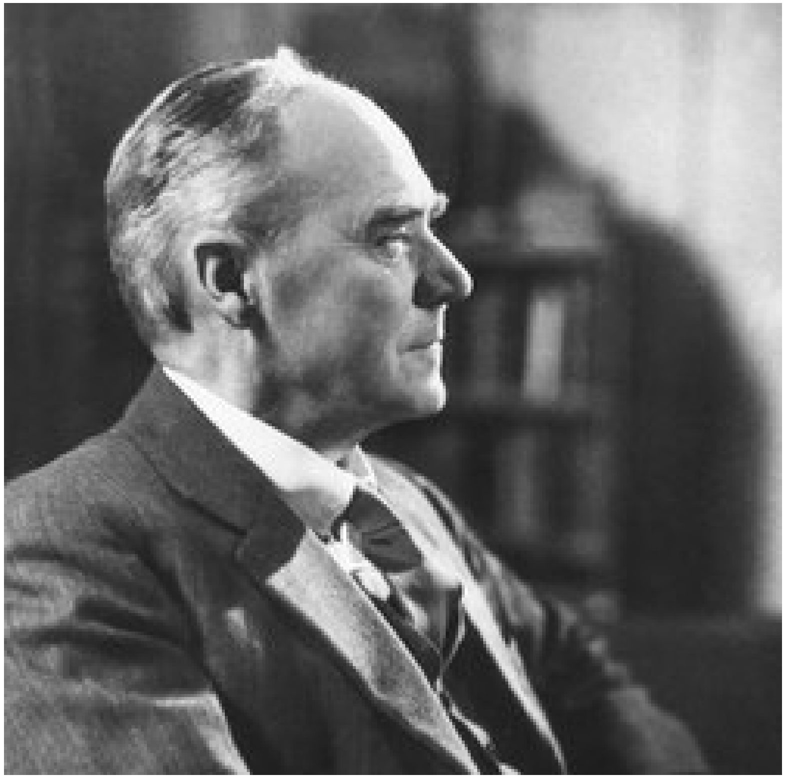
$$Y = a_1 X^5 + a_2 X^4 + a_3 X^3 + a_4 X^2 + a_5 X$$

(Curatolo & Teresi 2015)


0.10
0.05
0.00
−0.05
−0.10
0.0
0.1
0.2

Resistive Force Theory (RFT)

(1952)



Sir Geoffrey Ingram Taylor

(1886-1975)

Elongated Body Theory (EBT)

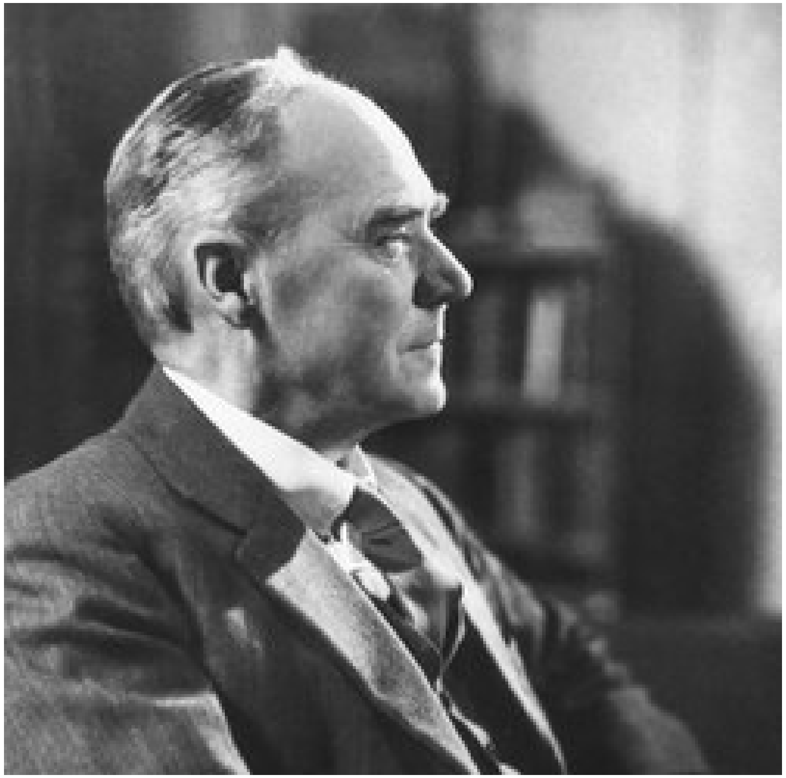
(1960)

Sir James Lighthill

(1924-1998)

Resistive Force Theory (RFT)

(1952)



Sir Geoffrey Ingram Taylor

(1886-1975)

Viscosity Dominant

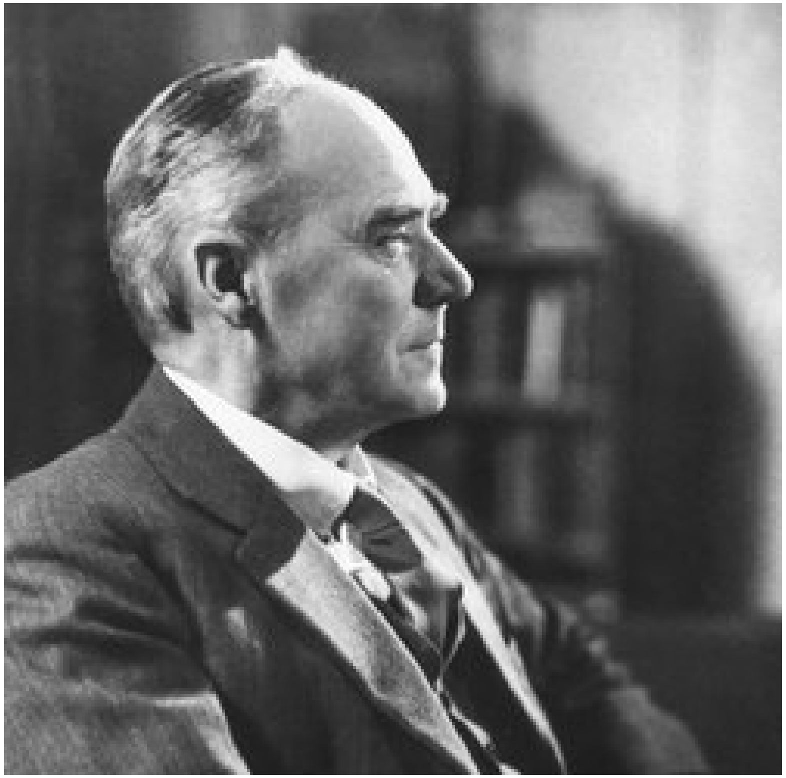
Elongated Body Theory (EBT)

(1960)

Sir James Lighthill

(1924-1998)

Resistive Force Theory (RFT)
(1952)



Sir Geoffrey Ingram Taylor
(1886-1975)

Viscosity Dominant

Elongated Body Theory (EBT)
(1960)

Sir James Lighthill
(1924-1998)

Inertia Dominant

Ming et al. 2018

Resistive Theory

$$F_s(s,t) = -\frac{1}{2}\rho H(s) C_D v^2(s,t)$$

Elongated Body (Reactive) Theory

$$F_a(s,t) = -\left(\frac{\partial}{\partial t} + U\frac{\partial}{\partial x}\right)\left[V(s,t)m(s)\right]$$

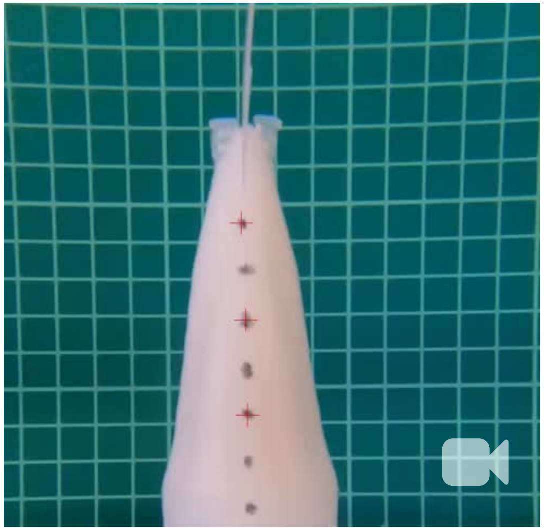
Combined Theory

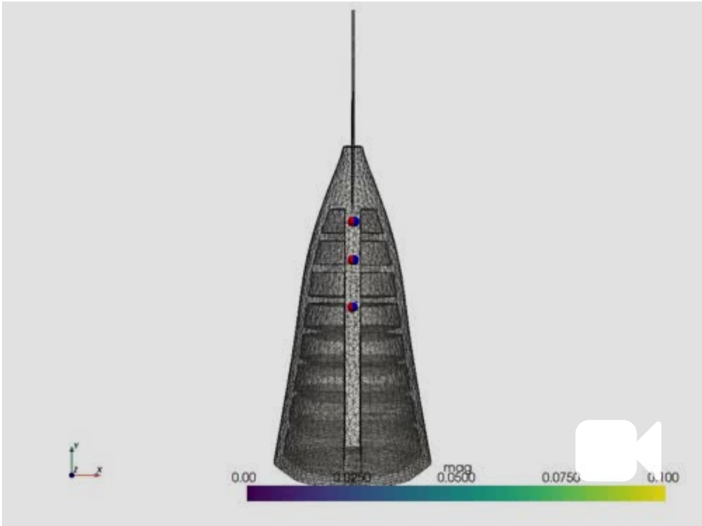
$$F_y(s,t) = C_a(s)F_a(s,t) + C_s(s)F_s(s,t)$$

Problem

How to simulate hydrodynamics **accurately** and **efficiently**?

How can the simulation be used for **computational design**?





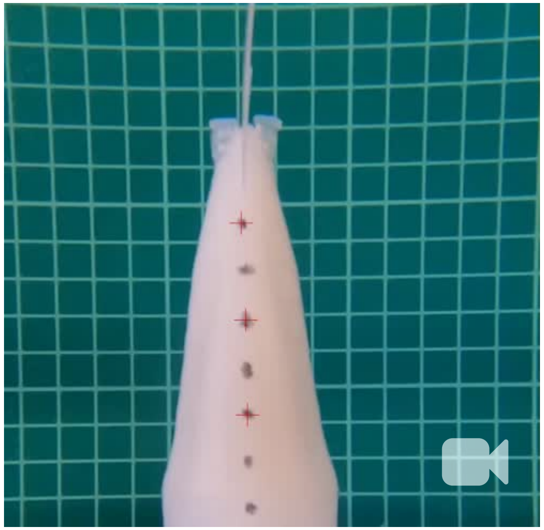
Problem

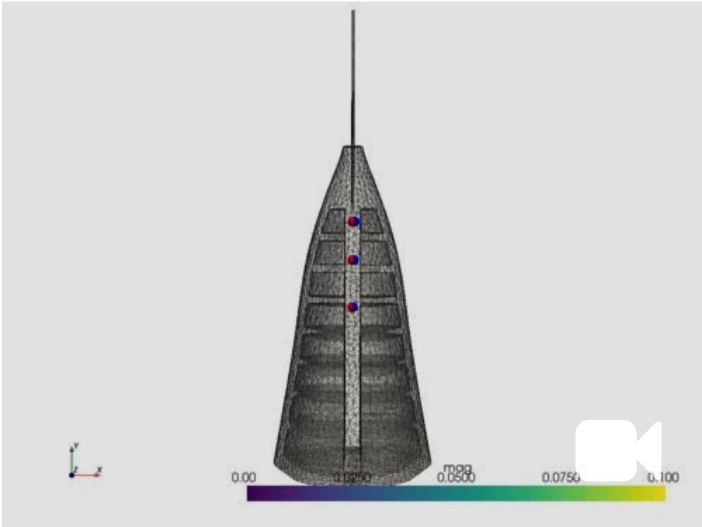
Heuristic

efficient, but inaccurate

How to simulate hydrodynamics **accurately** and **efficiently**?

How can the simulation be used for **computational design**?





Problem

Heuristic

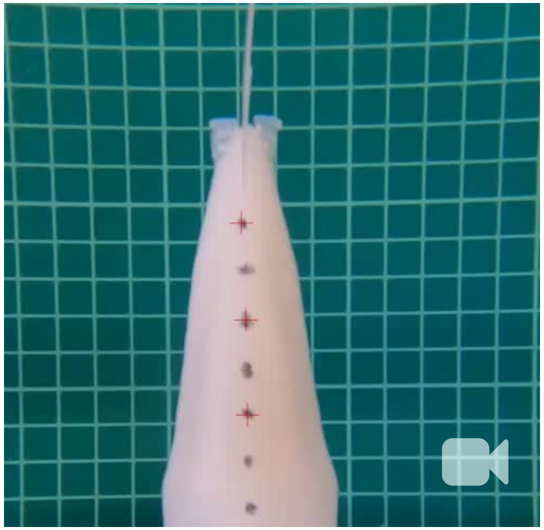
efficient, but inaccurate

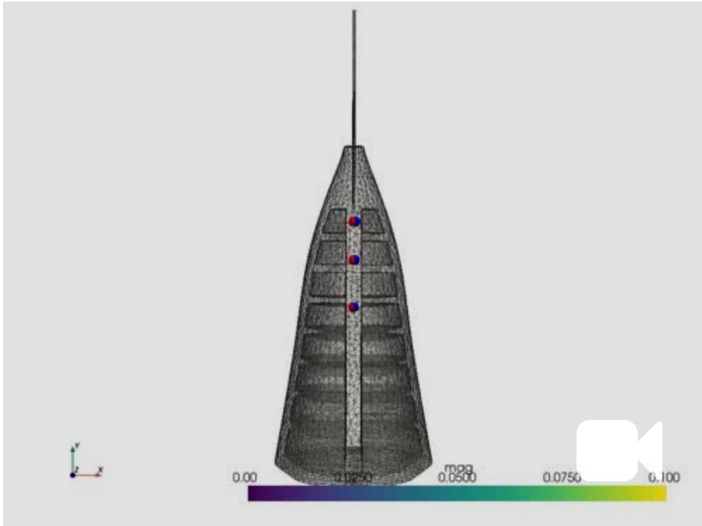
How to simulate hydrodynamics **accurately** and **efficiently**?

How can the simulation be used for **computational design**?

FSI

accurate, but inefficient





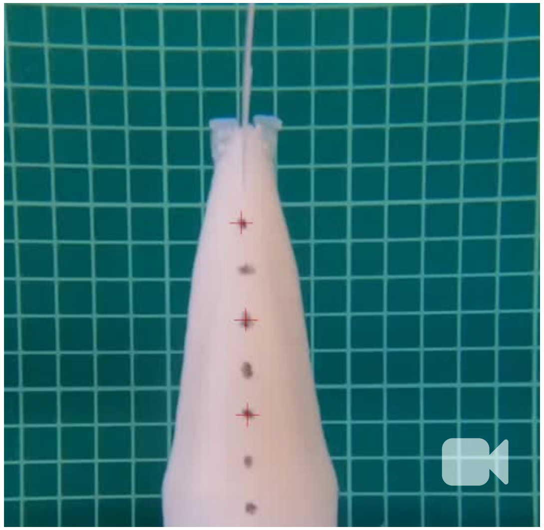
Problem

How to simulate hydrodynamics **accurately** and **efficiently**?

How can the simulation be used for **computational design**?

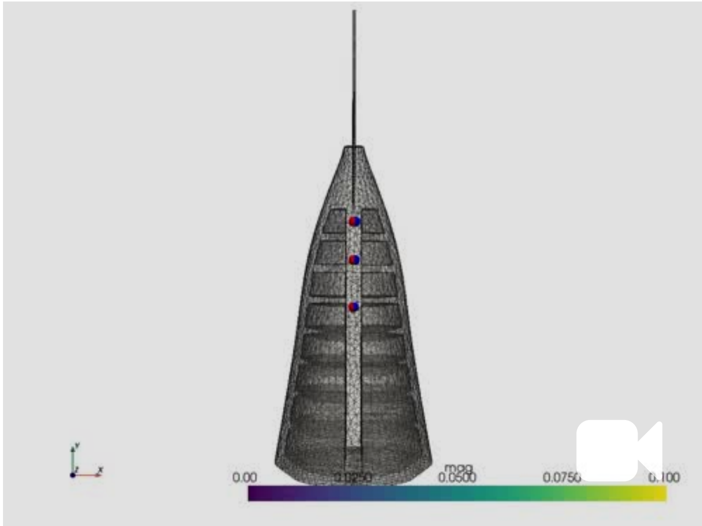
Heuristic

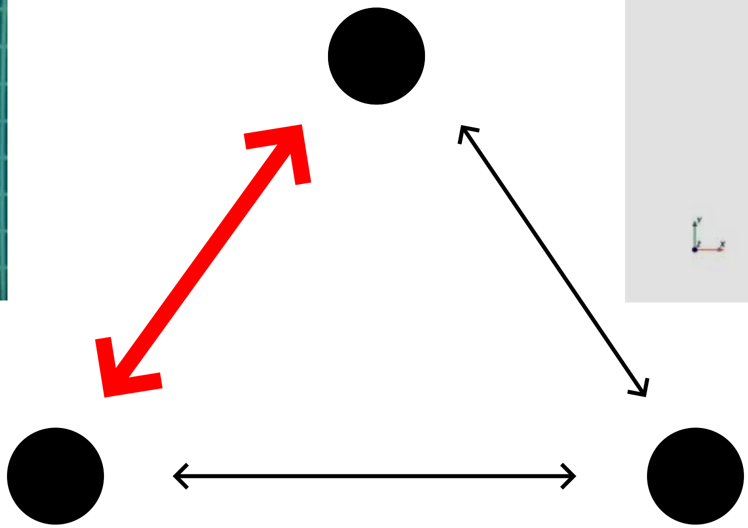
efficient, but inaccurate

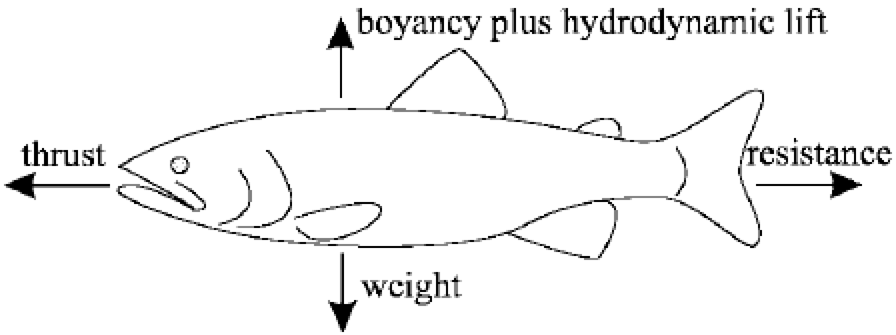


FSI

accurate, but inefficient

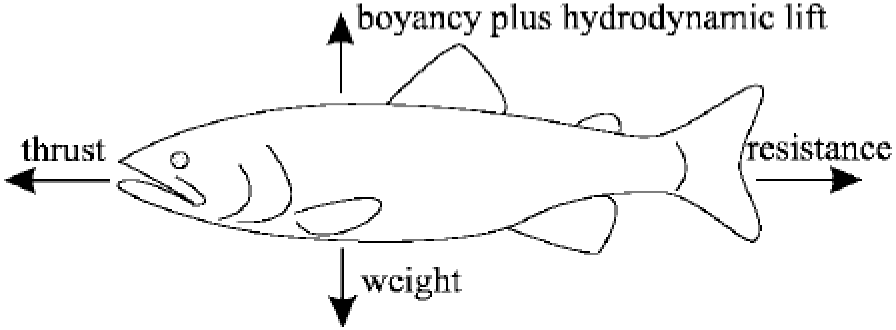






F_T F_D

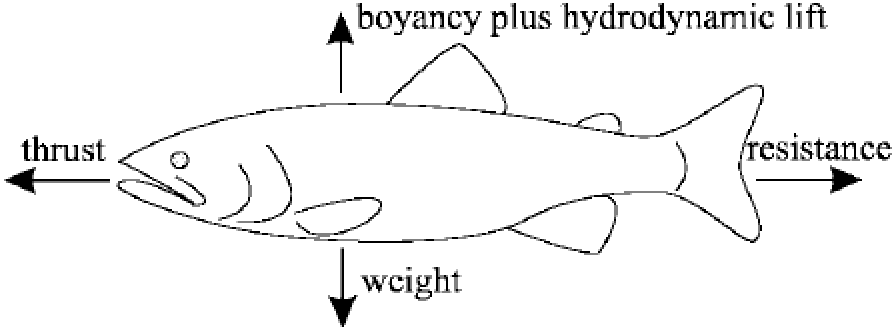
$$m\dot{v} = F_\mathrm{T} - F_\mathrm{D}$$

F_{T}  F_{D}

$$m\dot{v} = F_{\mathrm{T}} - F_{\mathrm{D}}$$

$$0 = F_{\mathrm{T}} - F_{\mathrm{D}}$$

F_{T}

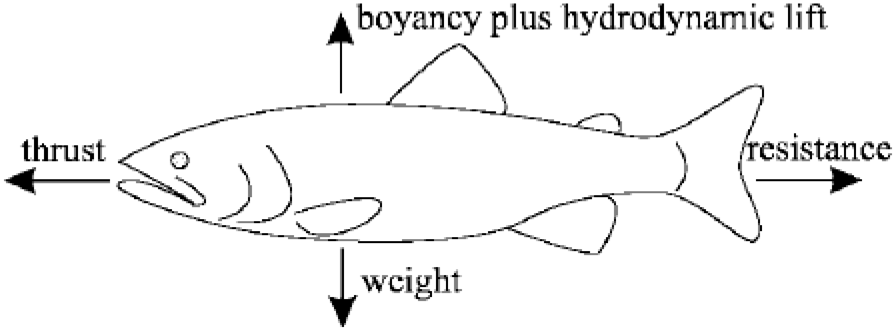


F_{D}

$$m\dot{v} = F_{\mathrm{T}} - F_{\mathrm{D}}$$

$$0 = F_{\mathrm{T}} - F_{\mathrm{D}}$$

$$0 = F_{\mathrm{T}} - \frac{1}{2}\rho A C_d v_{\mathrm{max}}^2$$

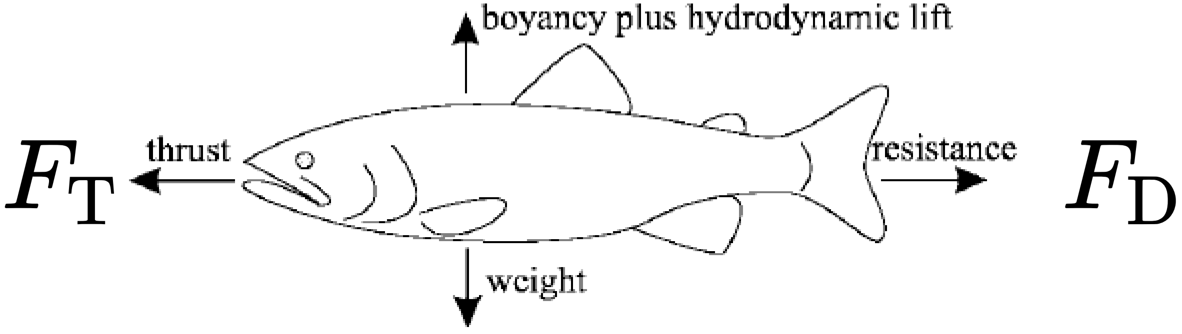
F_{T}  F_{D}

$$m\dot{v} = F_{\mathrm{T}} - F_{\mathrm{D}}$$

$$0 = F_{\mathrm{T}} - F_{\mathrm{D}}$$

$$0 = F_{\mathrm{T}} - \frac{1}{2}\rho A C_d v_{\mathrm{max}}^2$$

$$v_{\mathrm{max}} = \sqrt{\frac{2F_{\mathrm{T}}}{\rho A C_d}}$$



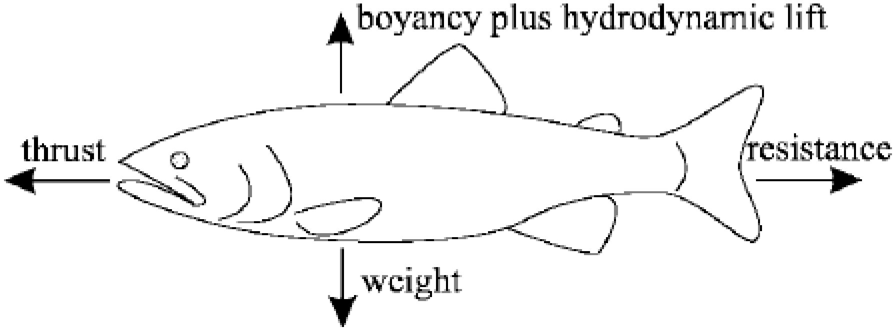
$$m\dot{v} = F_{\mathrm{T}} - F_{\mathrm{D}} \qquad F_{\mathrm{T}} = \hat{f}_t(\text{actuation}, \text{shape})$$

$$0 = F_{\mathrm{T}} - F_{\mathrm{D}}$$

$$0 = F_{\mathrm{T}} - \frac{1}{2}\rho A C_d v_{\max}^2$$

$$v_{\max} = \sqrt{\frac{2F_{\mathrm{T}}}{\rho A C_d}}$$

F_{T}



F_{D}

$$m\dot{v} = F_{\mathrm{T}} - F_{\mathrm{D}}$$

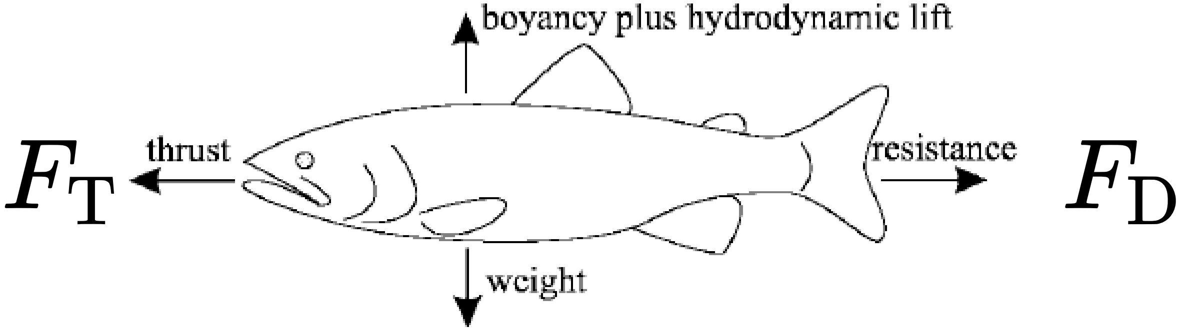
$$F_{\mathrm{T}} = \hat{f}_t(\text{actuation}, \text{shape})$$

$$F_{\mathrm{D}} = \hat{f}_d(\text{v}, \text{shape})$$

$$0 = F_{\mathrm{T}} - F_{\mathrm{D}}$$

$$0 = F_{\mathrm{T}} - \frac{1}{2}\rho A C_d v_{\max}^2$$

$$v_{\max} = \sqrt{\frac{2F_{\mathrm{T}}}{\rho A C_d}}$$



$$m\dot{v} = F_{\mathrm{T}} - F_{\mathrm{D}}$$

$$F_{\mathrm{T}} = \hat{f}_t(\mathbf{actuation}, \mathbf{shape})$$

$$F_{\mathrm{D}} = \hat{f}_d(\mathrm{v}, \mathbf{shape})$$

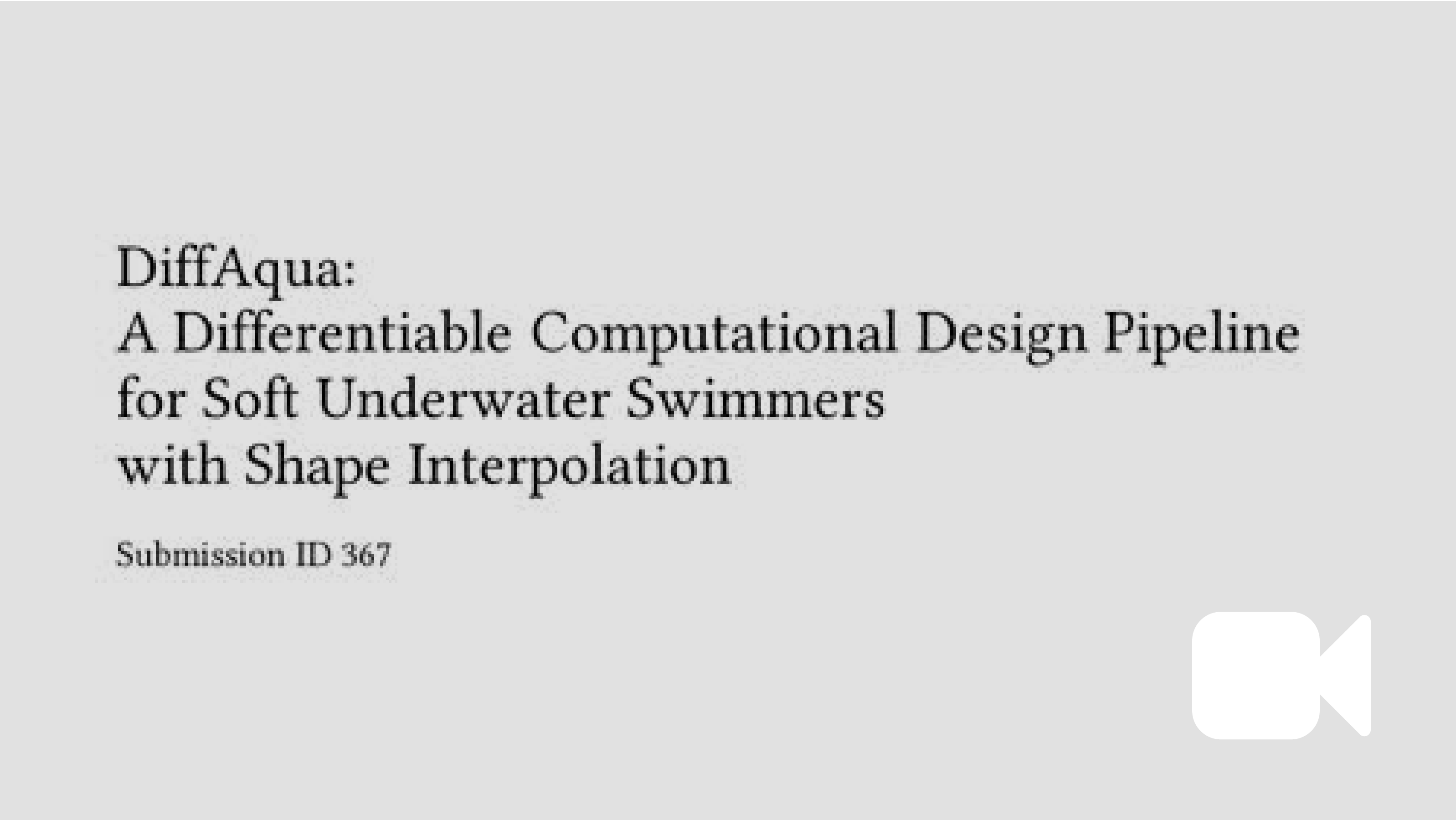
$$0 = F_{\mathrm{T}} - F_{\mathrm{D}}$$

$$0 = F_{\mathrm{T}} - \frac{1}{2}\rho A C_d v_{\max}^2$$

given an initial design, can we optimize this further?

$$v_{\max} = \sqrt{\frac{2F_{\mathrm{T}}}{\rho A C_d}}$$

DiffPD: Existing framework for differentiable simulation of soft bodies from MIT CDFG



(Du 2021, Ma 2021)

Projective Dynamics in 2D

$\mathbf{q} \in \mathbb{R}^{2m}$ $\dot{\mathbf{q}} \in \mathbb{R}^{2m}$ m is the number of particles

position velocity

(Kavan PBA 2014)

Projective Dynamics in 2D

$\mathbf{q} \in \mathbb{R}^{2m}$ $\dot{\mathbf{q}} \in \mathbb{R}^{2m}$ m is the number of particles

position velocity

Discrete time: $t_1, t_2, \ldots$

(Kavan PBA 2014)

Projective Dynamics in 2D

$\mathbf{q} \in \mathbb{R}^{2m}$ $\dot{\mathbf{q}} \in \mathbb{R}^{2m}$ m is the number of particles

position velocity

Discrete time: $t_1, t_2, \ldots$ state at t_n: $\begin{pmatrix} \mathbf{q}_n \\ \dot{\mathbf{q}}_n \end{pmatrix}$

(Kavan PBA 2014)

Projective Dynamics in 2D

$$\mathbf{q} \in \mathbb{R}^{2m} \qquad \dot{\mathbf{q}} \in \mathbb{R}^{2m}$$

position velocity

m is the number of particles

Discrete time: $t_1, t_2, \ldots$

state at t_n: $\begin{pmatrix} \mathbf{q}_n \\ \dot{\mathbf{q}}_n \end{pmatrix}$

External forces: $\mathbf{f}_{\text{ext}} \in \mathbb{R}^{2m}$

(Kavan PBA 2014)

Projective Dynamics in 2D

$\mathbf{q} \in \mathbb{R}^{2m}$ $\dot{\mathbf{q}} \in \mathbb{R}^{2m}$ m is the number of particles

position velocity

Discrete time: $t_1, t_2, \ldots$

state at t_n: $\begin{pmatrix} \mathbf{q}_n \\ \dot{\mathbf{q}}_n \end{pmatrix}$

External forces: $\mathbf{f}_{\text{ext}} \in \mathbb{R}^{2m}$

Internal forces: $\mathbf{f}_{\text{int}} = -\nabla E_{\text{int}}(\mathbf{q})$

(Kavan PBA 2014)

Projective Dynamics in 2D

$$\mathbf{q} \in \mathbb{R}^{2m} \quad \dot{\mathbf{q}} \in \mathbb{R}^{2m}$$

m is the number of particles

position velocity

Discrete time: $t_1, t_2, \ldots$

state at t_n: $\begin{pmatrix} \mathbf{q}_n \\ \dot{\mathbf{q}}_n \end{pmatrix}$

External forces: $\mathbf{f}_{\text{ext}} \in \mathbb{R}^{2m}$

Internal forces: $\mathbf{f}_{\text{int}} = -\nabla E_{\text{int}}(\mathbf{q})$

Muscle fibers: $E(\mathbf{q}) = \frac{w}{2}|(1-a)\mathbf{Fm}|^2$

(Kavan PBA 2014)

Projective Dynamics in 2D

$\mathbf{q} \in \mathbb{R}^{2m}$ $\dot{\mathbf{q}} \in \mathbb{R}^{2m}$ m is the number of particles

position velocity

Discrete time: $t_1, t_2, \ldots$ state at t_n: $\begin{pmatrix} \mathbf{q}_n \\ \dot{\mathbf{q}}_n \end{pmatrix}$

External forces: $\mathbf{f}_{\text{ext}} \in \mathbb{R}^{2m}$

Internal forces: $\mathbf{f}_{\text{int}} = -\nabla E_{\text{int}}(\mathbf{q})$

Muscle fibers: $E(\mathbf{q}) = \frac{w}{2}|(1-a)\mathbf{Fm}|^2$

Implicit Euler:

(Kavan PBA 2014)

Projective Dynamics in 2D

$\mathbf{q} \in \mathbb{R}^{2m}$ $\dot{\mathbf{q}} \in \mathbb{R}^{2m}$ m is the number of particles

position velocity

Discrete time: $t_1, t_2, \ldots$ state at t_n: $\begin{pmatrix} \mathbf{q}_n \\ \dot{\mathbf{q}}_n \end{pmatrix}$

External forces: $\mathbf{f}_{\text{ext}} \in \mathbb{R}^{2m}$

Internal forces: $\mathbf{f}_{\text{int}} = -\nabla E_{\text{int}}(\mathbf{q})$

Muscle fibers: $E(\mathbf{q}) = \frac{w}{2}|(1-a)\mathbf{F}\mathbf{m}|^2$

Implicit Euler:

$$\mathbf{q}_{n+1} = \mathbf{q}_n + h\dot{\mathbf{q}}_{n+1}$$

$$\dot{\mathbf{q}}_{n+1} = \dot{\mathbf{q}}_n + h\mathbf{M}^{-1}[\mathbf{f}_{\text{int}}(\mathbf{q}_{n+1}) + \mathbf{f}_{\text{ext}}]$$

(Kavan PBA 2014)

Projective Dynamics in 2D (cont'd)

Must solve nonlinear system of equations,

equivalent to optimization problem.

$$\min_{\mathbf{q}_{i+1}} \frac{1}{2h^2}\left|\mathbf{M}^{\frac{1}{2}}(\mathbf{q}_{i+1}-\mathbf{y})\right|^2 + E_{\text{int}}(\mathbf{q}_{i+1})$$

(Bouaziz 2014)

Projective Dynamics in 2D (cont'd)

Must solve nonlinear system of equations,

equivalent to optimization problem.

$$\min_{\mathbf{q}_{i+1}} \frac{1}{2h^2} \left|\mathbf{M}^{\frac{1}{2}}(\mathbf{q}_{i+1} - \mathbf{y})\right|^2 + E_{\text{int}}(\mathbf{q}_{i+1})$$

Using Newton-Raphson

$$\mathbf{A}_{\text{N}} = \frac{1}{h^2}\mathbf{M} + \nabla^2 E_{\text{int}}(\mathbf{q}_{i+1})$$

(Bouaziz 2014)

Projective Dynamics in 2D (cont'd)

Must solve nonlinear system of equations, equivalent to optimization problem.

$$\min_{\mathbf{q}_{i+1}} \ \frac{1}{2h^2}\left|\mathbf{M}^{\frac{1}{2}}(\mathbf{q}_{i+1} - \mathbf{y})\right|^2 + E_{\text{int}}(\mathbf{q}_{i+1})$$

Using Newton-Raphson

$$\mathbf{A}_{\text{N}} = \frac{1}{h^2}\mathbf{M} + \nabla^2 E_{\text{int}}(\mathbf{q}_{i+1})$$

In Projective Dynamics, we decouple nonlinear material model from linear systems of equations

Minimize by alternating between a local and **global step**

(Bouaziz 2014)

Projective Dynamics in 2D (cont'd)

Must solve nonlinear system of equations, equivalent to optimization problem.

$$\min_{\mathbf{q}_{i+1}} \frac{1}{2h^2} |\mathbf{M}^{\frac{1}{2}}(\mathbf{q}_{i+1} - \mathbf{y})|^2 + E_{\text{int}}(\mathbf{q}_{i+1})$$

Using Newton-Raphson

$$\mathbf{A}_{\text{N}} = \frac{1}{h^2}\mathbf{M} + \nabla^2 E_{\text{int}}(\mathbf{q}_{i+1})$$

In Projective Dynamics, we decouple nonlinear material model from linear systems of equations

Minimize by alternating between a local and **global step**

$$\mathbf{A} = \frac{1}{h^2}\mathbf{M} + \sum_e w_e \mathbf{G}_e^\top \mathbf{G}_e$$

(Bouaziz 2014)

Projective Dynamics in 2D (cont'd)

Must solve nonlinear system of equations,

equivalent to optimization problem.

$$\min_{\mathbf{q}_{i+1}} \frac{1}{2h^2} \left|\mathbf{M}^{\frac{1}{2}}(\mathbf{q}_{i+1} - \mathbf{y})\right|^2 + E_{\text{int}}(\mathbf{q}_{i+1})$$

Using Newton-Raphson

$$\mathbf{A}_{\text{N}} = \frac{1}{h^2}\mathbf{M} + \nabla^2 E_{\text{int}}(\mathbf{q}_{i+1})$$

In Projective Dynamics, we decouple nonlinear material model from linear systems of equations

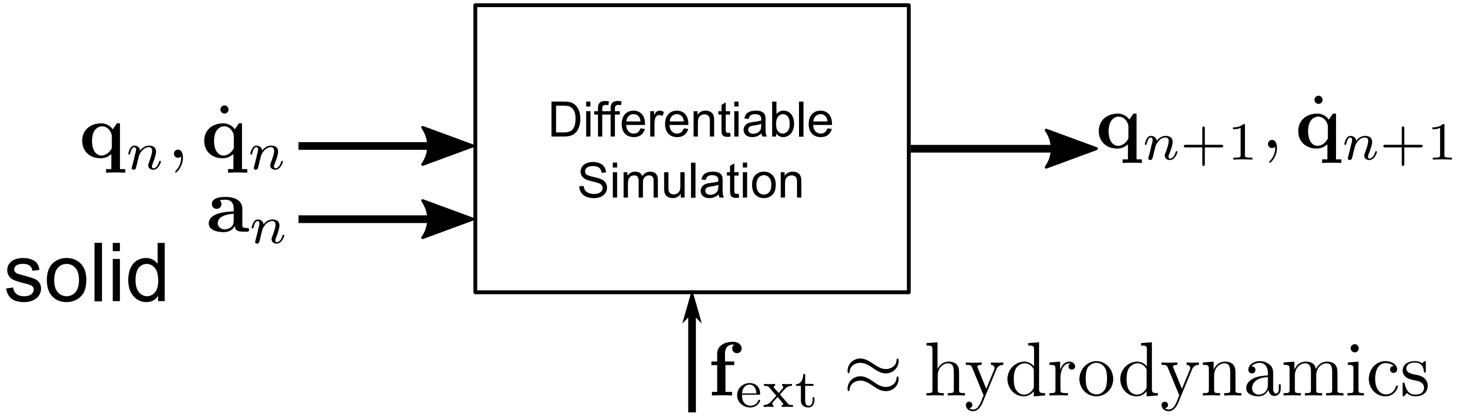
Minimize by alternating between a local and **global step**

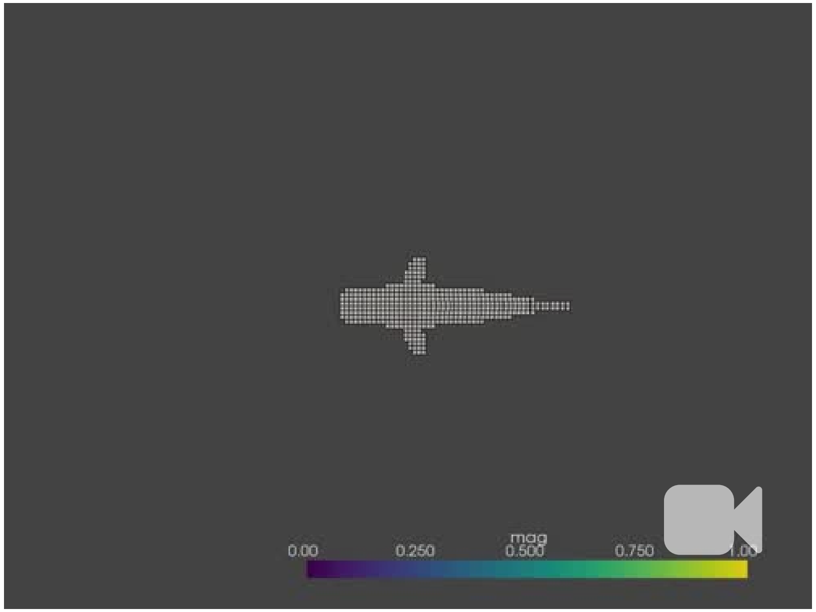
$$\mathbf{A} = \frac{1}{h^2}\mathbf{M} + \sum_e w_e \mathbf{G}_e^\top \mathbf{G}_e$$

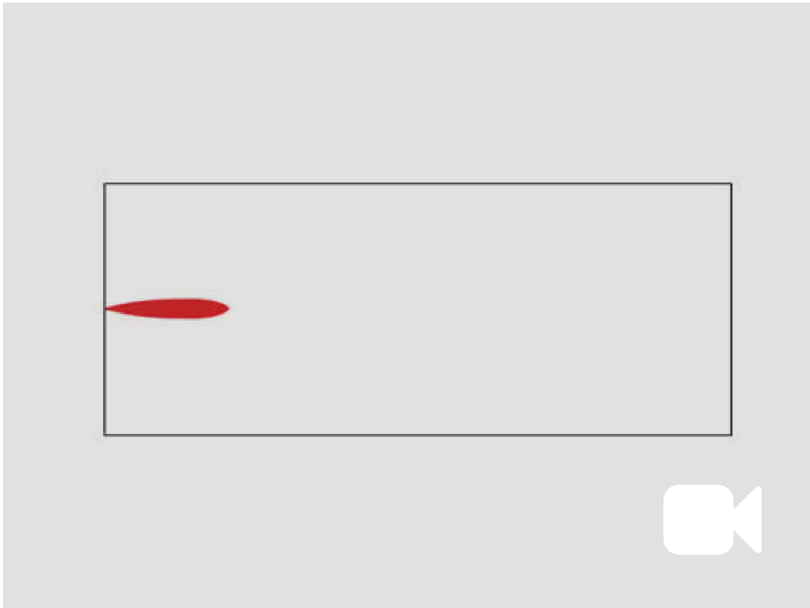
$$\mathbf{A}\mathbf{q} = \mathbf{b}$$ Prefactor! $\mathbf{b}$ changes on each time step

(Bouaziz 2014)

Projective Dynamics in 2D (cont'd)





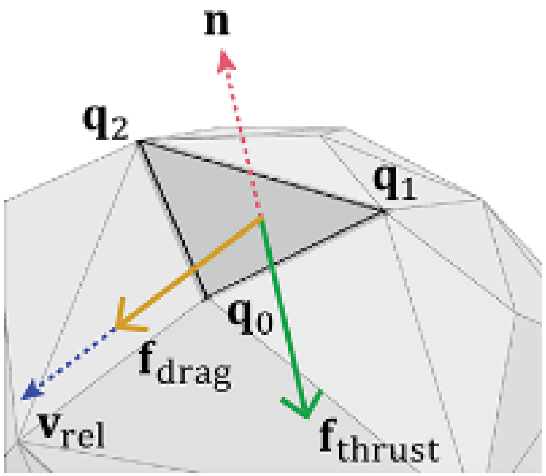


Min et al. 2019

$$\mathbf{f}_{\mathrm{drag}} \propto C_d(\Phi)|\mathbf{v}_{\mathrm{rel}}|^2\mathbf{d} \qquad \mathbf{d} = \frac{\mathbf{v}_{\mathrm{rel}}}{|\mathbf{v}_{\mathrm{rel}}|}$$

$$\mathbf{f}_{\mathrm{thrust}} \propto C_t(\Phi)|\mathbf{v}_{\mathrm{rel}}|^2\mathbf{n}$$

$$\Phi = \frac{\pi}{2} - \cos^{-1}(\mathbf{n}\cdot\mathbf{v}_{\mathrm{rel}})$$

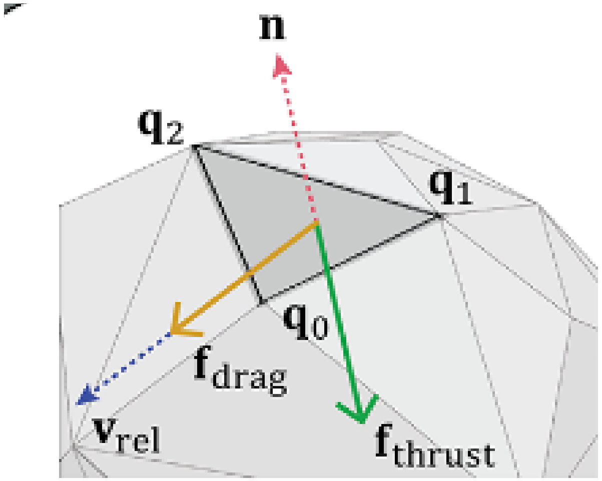


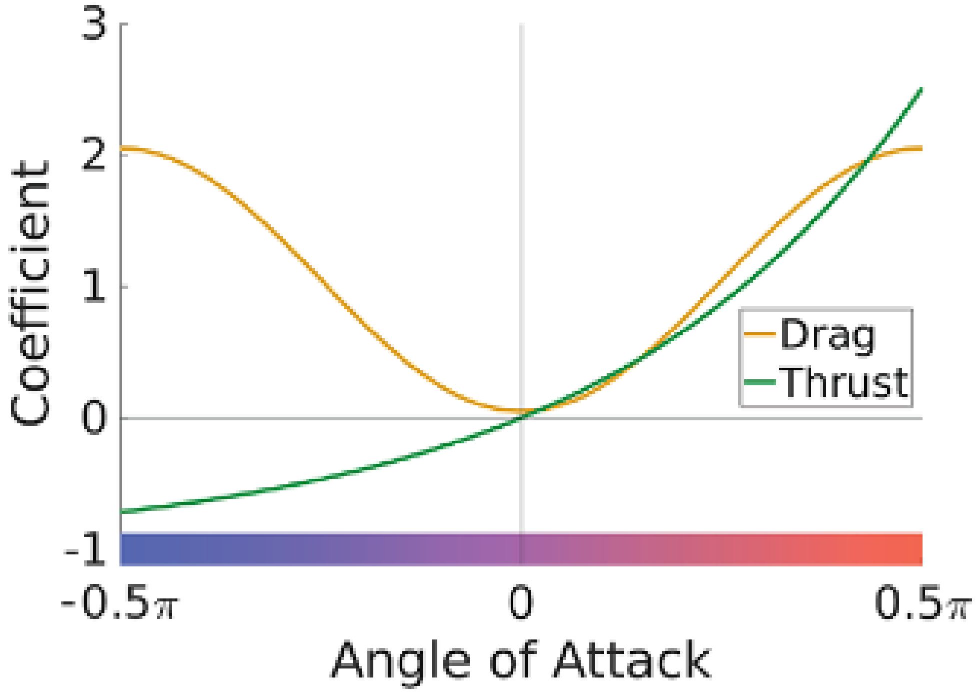
Min et al. 2019

$$\mathbf{f}_{\text{drag}} \propto C_d(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{d} \qquad \mathbf{d} = \frac{\mathbf{v}_{\text{rel}}}{|\mathbf{v}_{\text{rel}}|}$$

$$\mathbf{f}_{\text{thrust}} \propto C_t(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{n}$$

$$\Phi = \frac{\pi}{2} - \cos^{-1}(\mathbf{n} \cdot \mathbf{v}_{\text{rel}})$$





Min et al. 2019

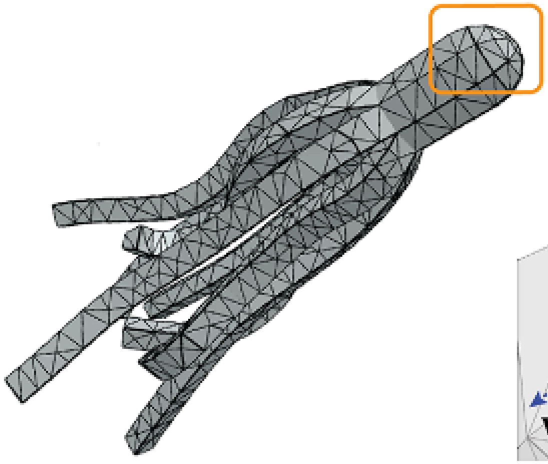
$$\mathbf{f}_{\text{drag}} = \frac{1}{2}\rho A C_d(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{d} \qquad \mathbf{d} = \frac{\mathbf{v}_{\text{rel}}}{|\mathbf{v}_{\text{rel}}|}$$

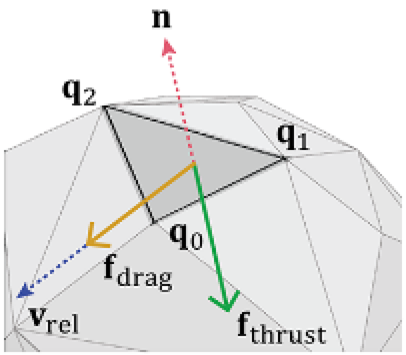
$$\mathbf{f}_{\text{thrust}} = -\frac{1}{2}\rho A C_t(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{n}$$

Min et al. 2019

$$\mathbf{f}_{\text{drag}} = \frac{1}{2}\rho A C_d(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{d} \qquad \mathbf{d} = \frac{\mathbf{v}_{\text{rel}}}{|\mathbf{v}_{\text{rel}}|}$$

$$\mathbf{f}_{\text{thrust}} = -\frac{1}{2}\rho A C_t(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{n}$$

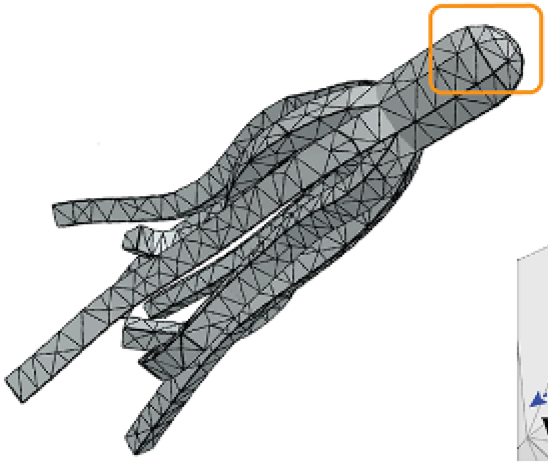


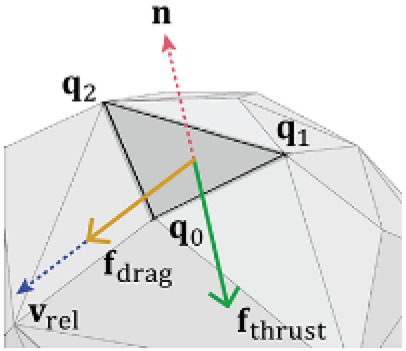


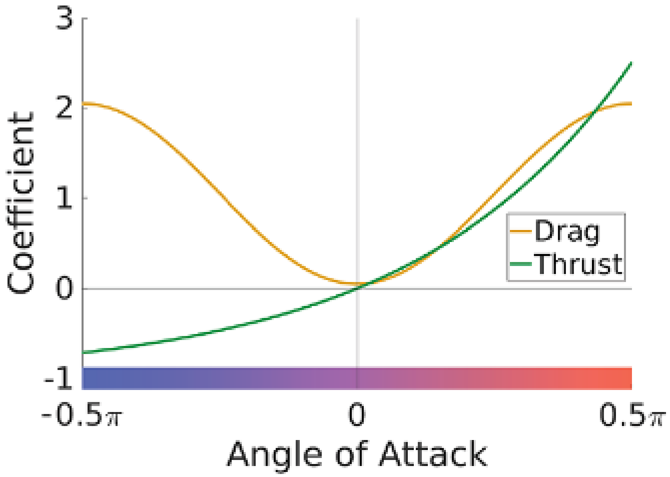
Min et al. 2019

$$\mathbf{f}_{\text{drag}} = \frac{1}{2}\rho A C_d(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{d} \qquad \mathbf{d} = \frac{\mathbf{v}_{\text{rel}}}{|\mathbf{v}_{\text{rel}}|}$$

$$\mathbf{f}_{\text{thrust}} = -\frac{1}{2}\rho A C_t(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{n}$$



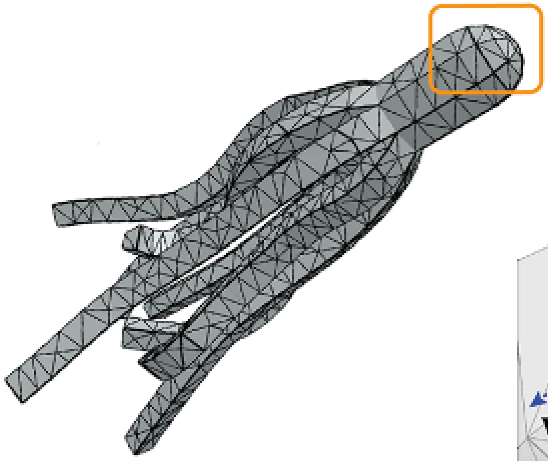


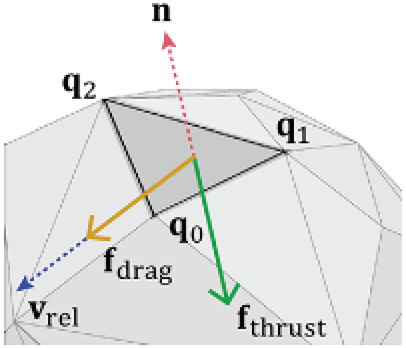


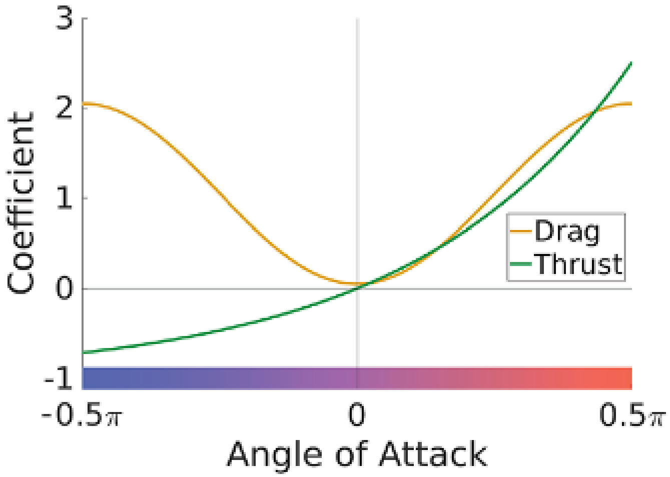
Min et al. 2019

$$\mathbf{f}_{\text{drag}} = \frac{1}{2}\rho A C_d(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{d} \qquad \mathbf{d} = \frac{\mathbf{v}_{\text{rel}}}{|\mathbf{v}_{\text{rel}}|}$$

$$\mathbf{f}_{\text{thrust}} = -\frac{1}{2}\rho A C_t(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{n}$$





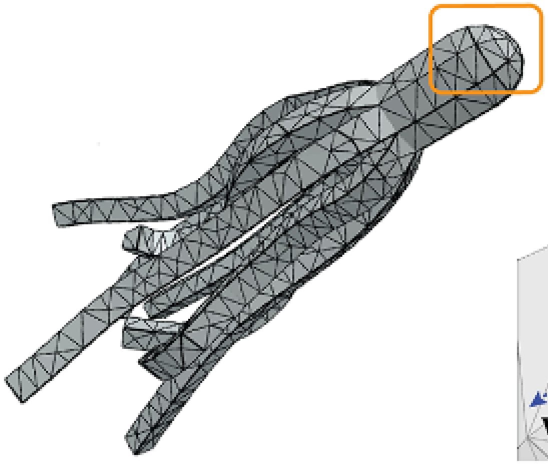


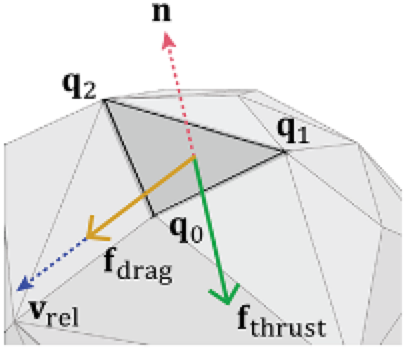
$$\mathbf{v}_{\text{rel}} = \mathbf{v}_{\text{water}} - \frac{1}{3}(\mathbf{v}_0 + \mathbf{v}_1 + \mathbf{v}_2)$$

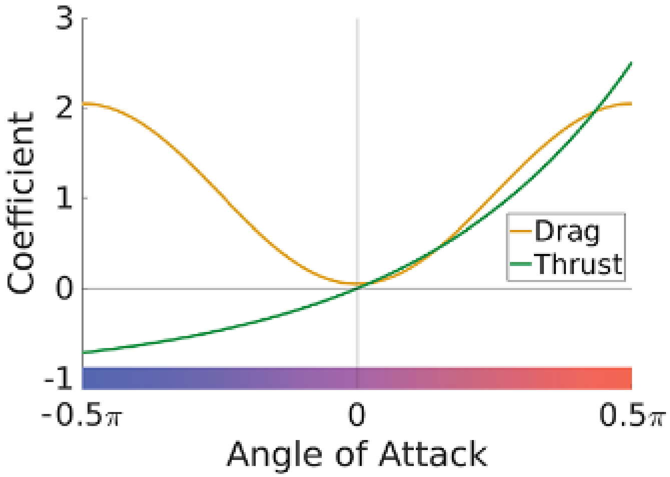
Min et al. 2019

$$\mathbf{f}_{\text{drag}} = \frac{1}{2}\rho A C_d(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{d} \qquad \mathbf{d} = \frac{\mathbf{v}_{\text{rel}}}{|\mathbf{v}_{\text{rel}}|}$$

$$\mathbf{f}_{\text{thrust}} = -\frac{1}{2}\rho A C_t(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{n}$$







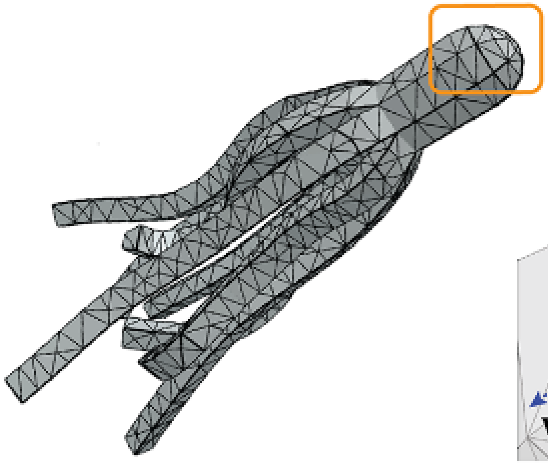
$$\mathbf{v}_{\text{rel}} = \mathbf{v}_{\text{water}} - \frac{1}{3}(\mathbf{v}_0 + \mathbf{v}_1 + \mathbf{v}_2)$$

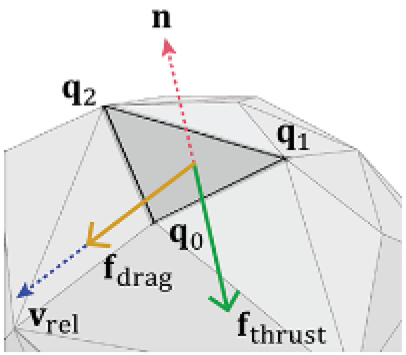
$$\Phi = \frac{\pi}{2} - \cos^{-1}(\mathbf{n}\cdot\mathbf{v}_{\text{rel}})$$

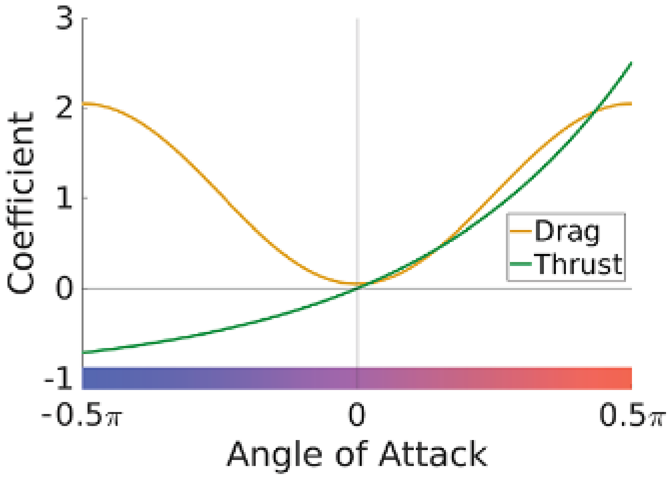
Min et al. 2019

$$\mathbf{f}_{\text{drag}} = \frac{1}{2}\rho A C_d(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{d} \qquad \mathbf{d} = \frac{\mathbf{v}_{\text{rel}}}{|\mathbf{v}_{\text{rel}}|}$$

$$\mathbf{f}_{\text{thrust}} = -\frac{1}{2}\rho A C_t(\Phi)|\mathbf{v}_{\text{rel}}|^2\mathbf{n}$$

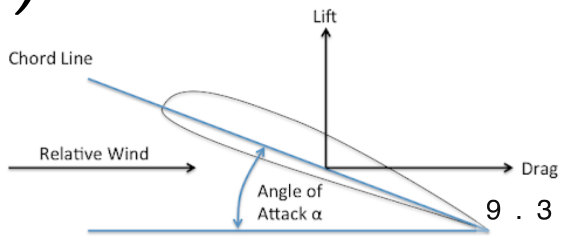




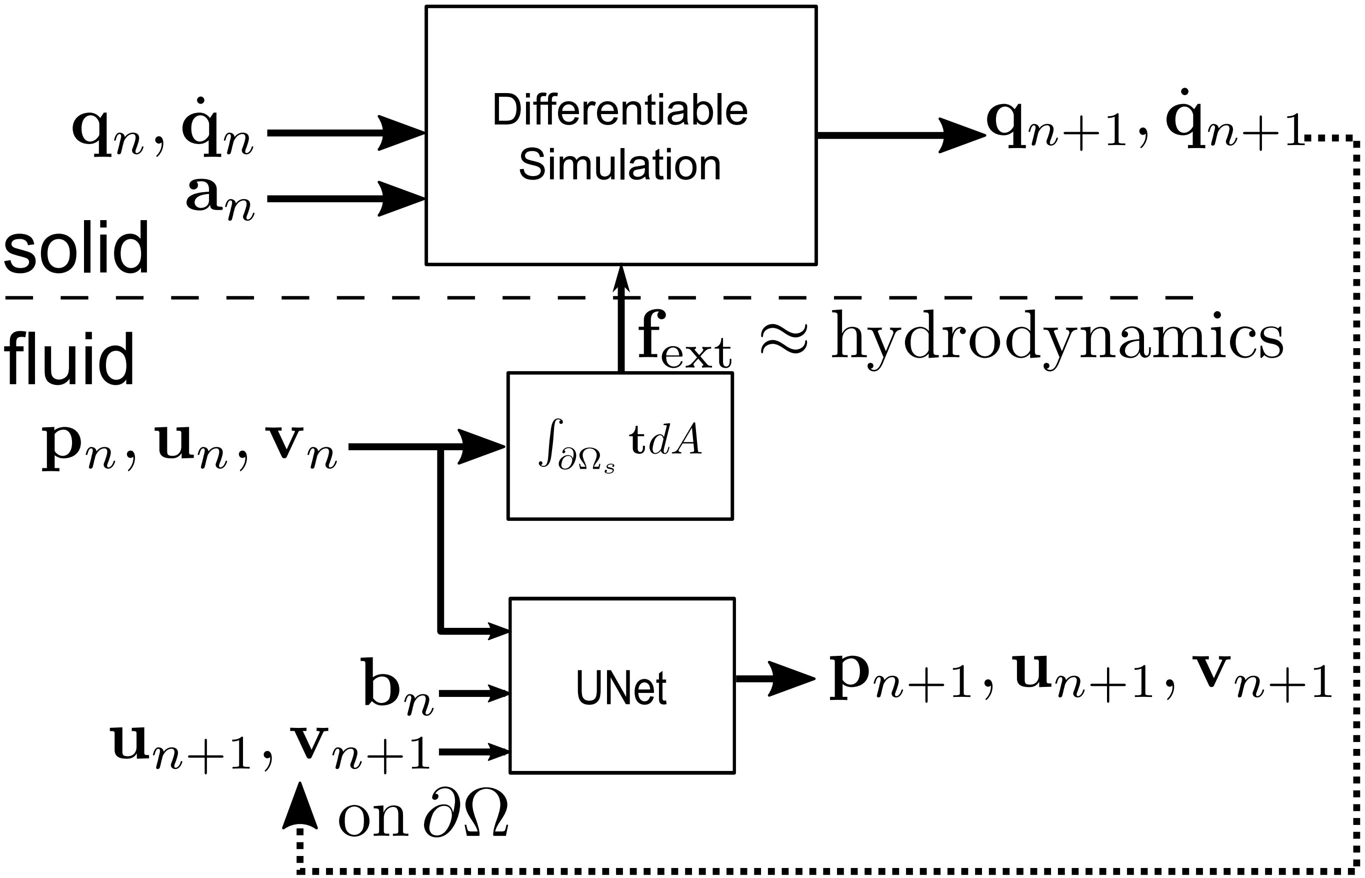


$$\mathbf{v}_{\text{rel}} = \mathbf{v}_{\text{water}} - \frac{1}{3}(\mathbf{v}_0 + \mathbf{v}_1 + \mathbf{v}_2)$$

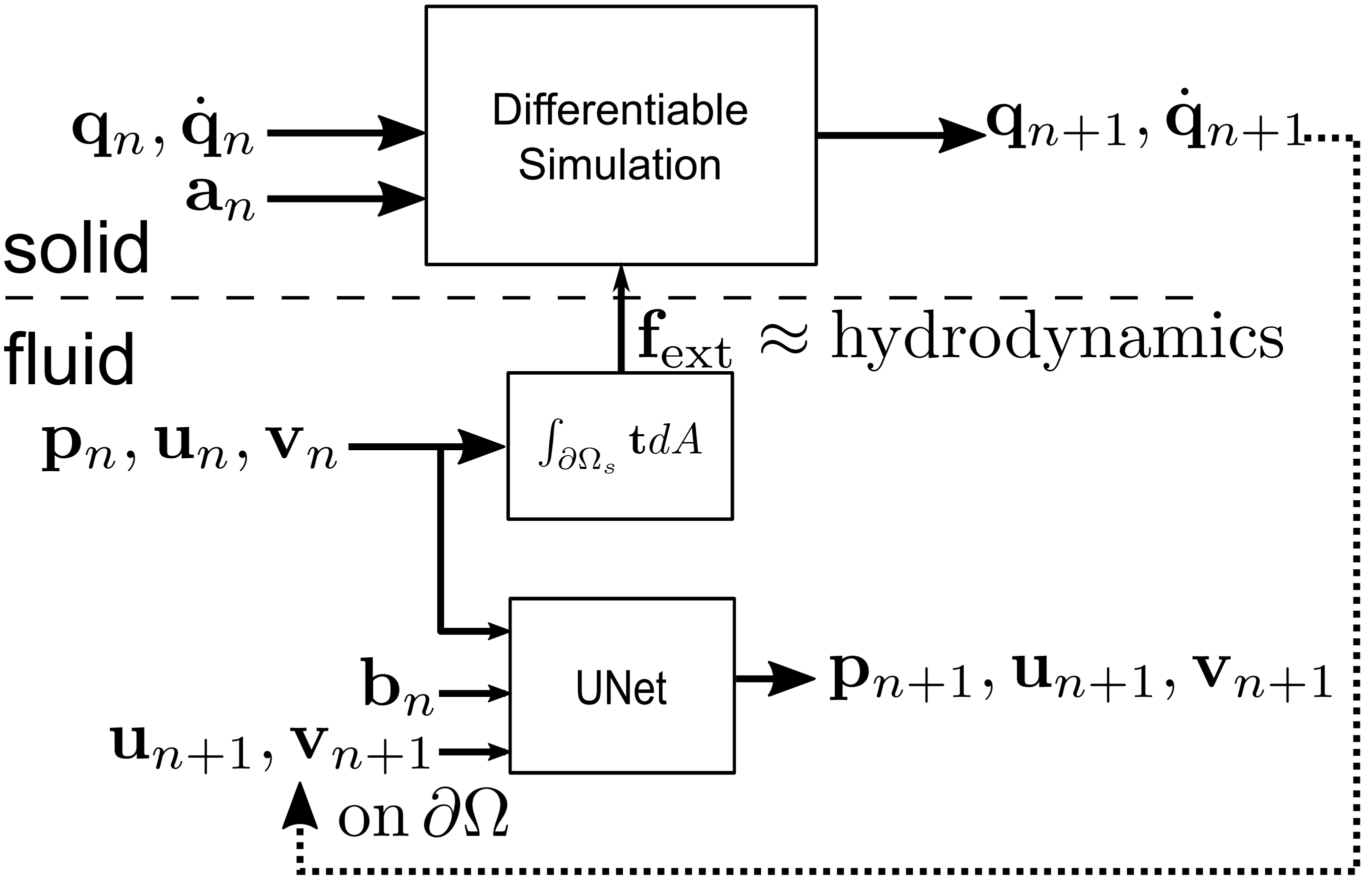
$$\Phi = \frac{\pi}{2} - \cos^{-1}(\mathbf{n}\cdot\mathbf{v}_{\text{rel}})$$



Proposed Solution



Proposed Solution



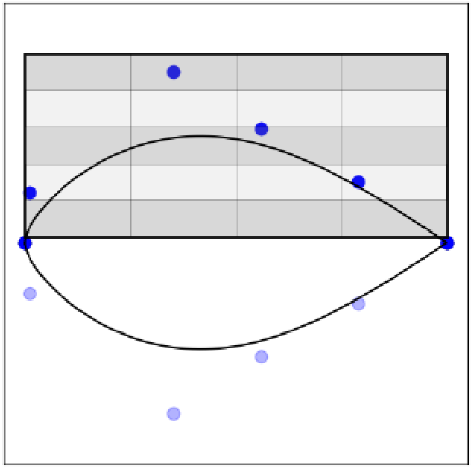
$$\mathbf{f}_{\text{pressure}} = \int_{\partial\Omega} p\mathbf{n}\, dl$$

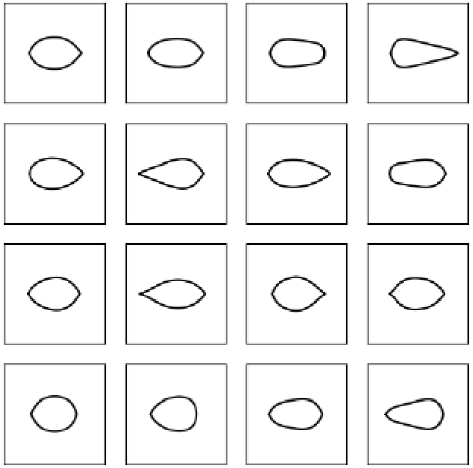
$$\mathbf{f}_{\text{viscous}} = \int_{\partial\Omega} \mu\mathbf{n} \times \omega\, dl \quad \omega = \nabla \times \mathbf{u}$$

Numerical investigation of minimum drag profiles in laminar flow using deep learning surrogates

Li-Wei Chen, Berkay Alp Cakal, Xiangyu Hu, Nils Thuerey

September 2020





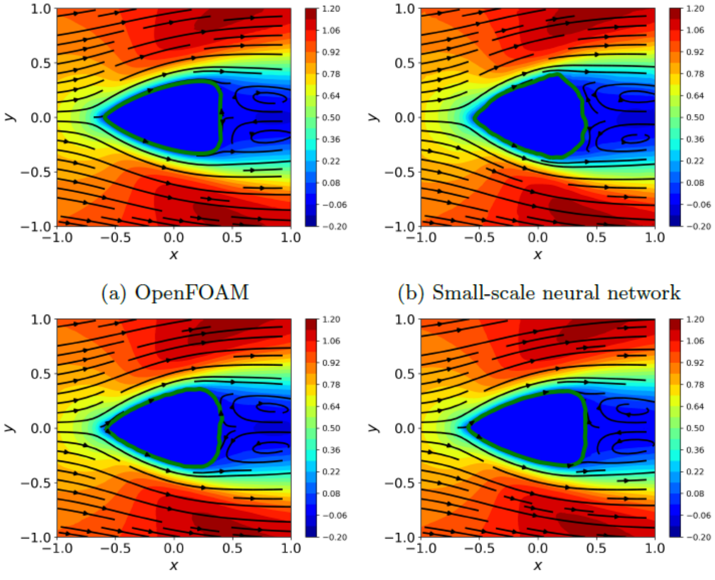
Name	# of flowfields	Re	NN models
Dataset-1	2500	1	small, medium & large
Dataset-40	2500	40	small, medium & large
Dataset-Range	3028	0.5-42.5	large

Table 1: Three datasets for training the neural network models.

Numerical investigation of minimum drag in laminar flow using deep learning surr

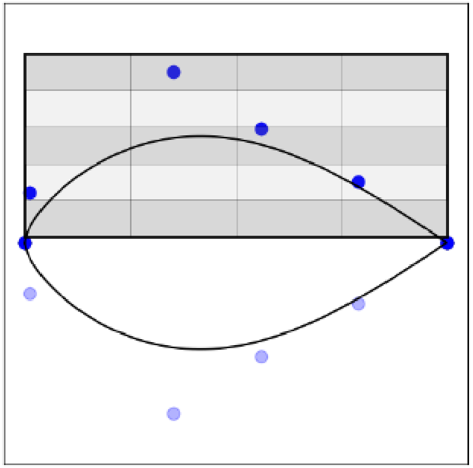
Li-Wei Chen, Berkay Alp Cakal, Xiangyu Hu, Nils T

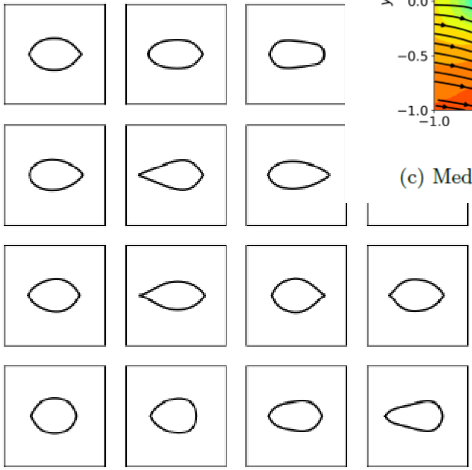
September 2020



(a) OpenFOAM (b) Small-scale neural network

(c) Medium-scale neural network (d) Large-scale neural network





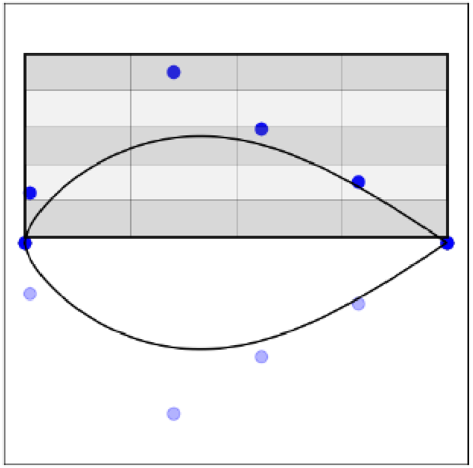
Name	# of flowfields	Re	NN models
Dataset-1	2500	1	small, medium & large
Dataset-40	2500	40	small, medium & large
Dataset-Range	3028	0.5-42.5	large

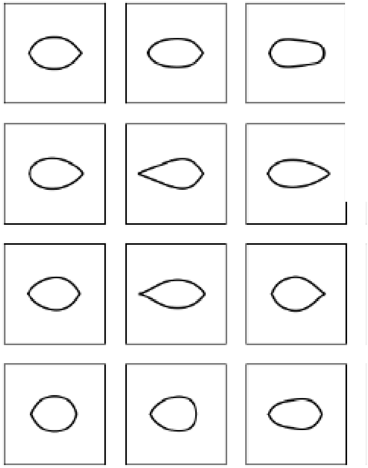
Table 1: Three datasets for training the neural network models.

Numerical investigation of minimum drag in laminar flow using deep learning surr

Li-Wei Chen, Berkay Alp Cakal, Xiangyu Hu, Nils T

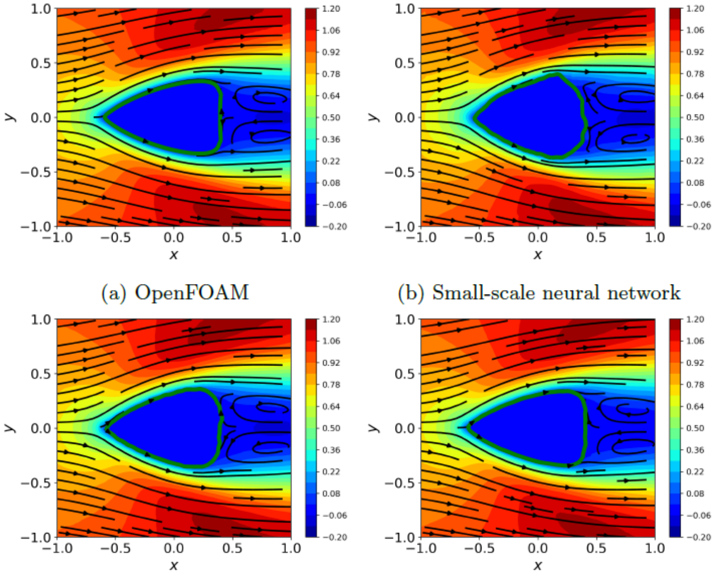
September 2020





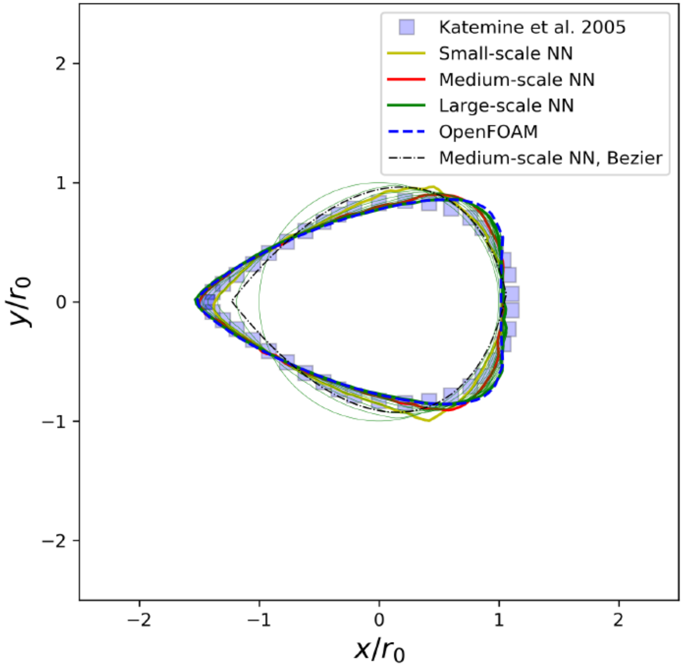
Name	# of flowfields	Re	NN models
Dataset-1	2500	1	small, medium & l
Dataset-40	2500	40	small, medium & l
Dataset-Range	3028	0.5-42.5	large

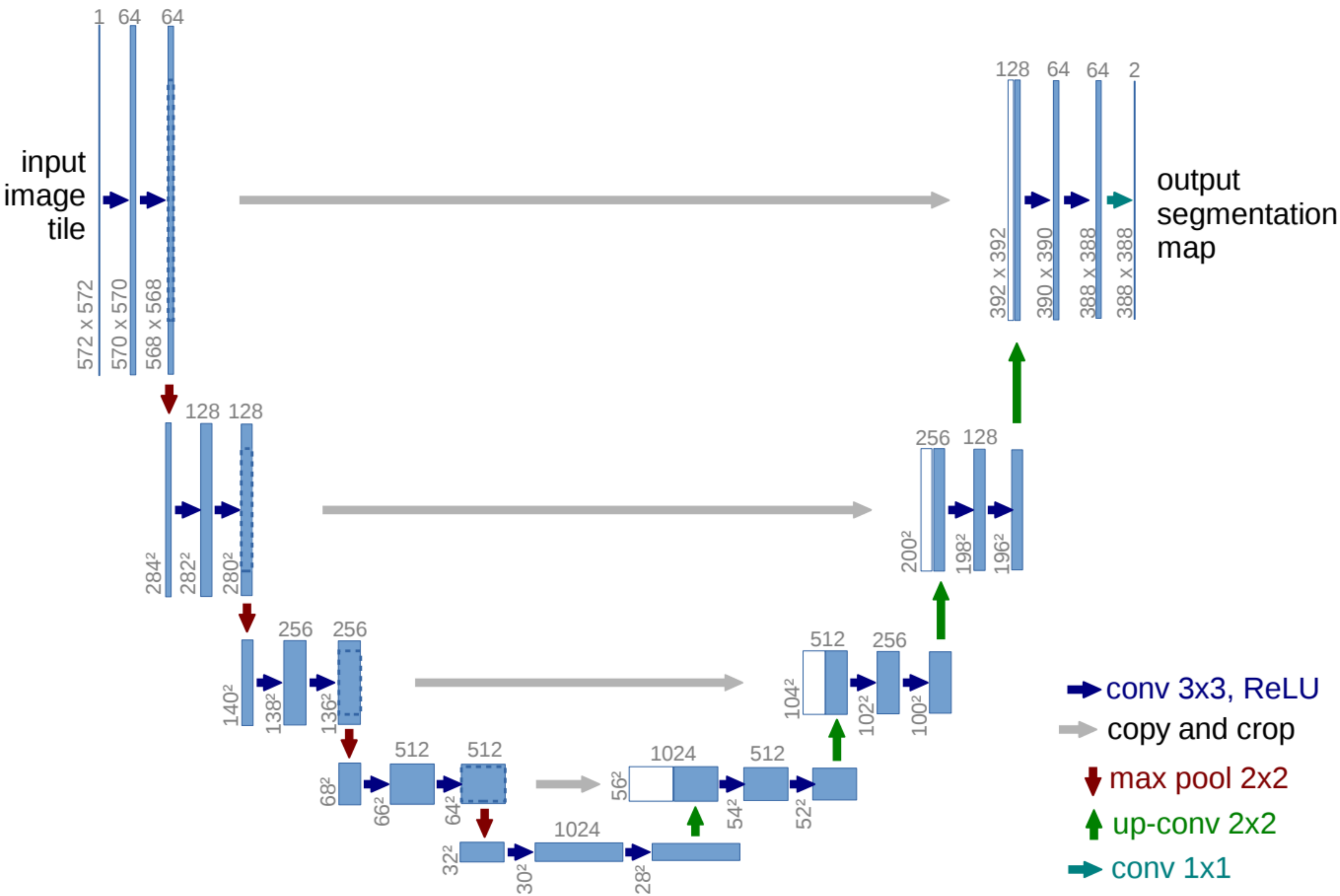
Table 1: Three datasets for training the neural network mode



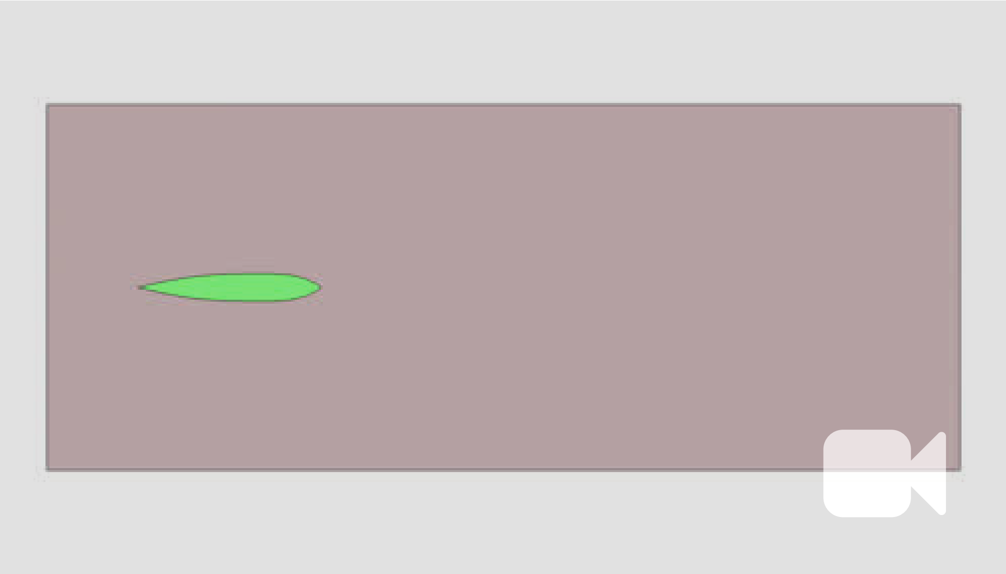
(a) OpenFOAM (b) Small-scale neural network

(c) Medium-scale neural network (d) Large-scale neural network



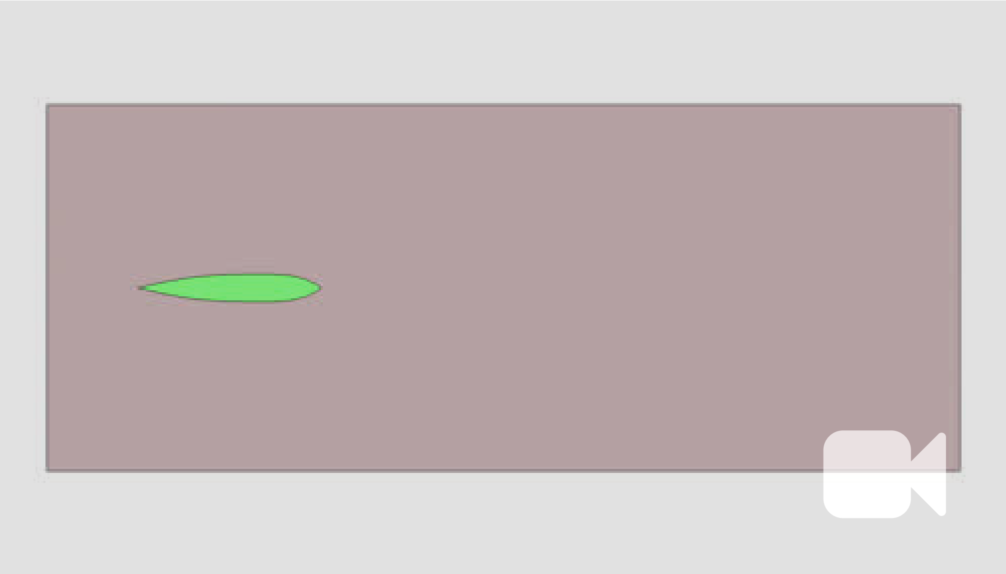


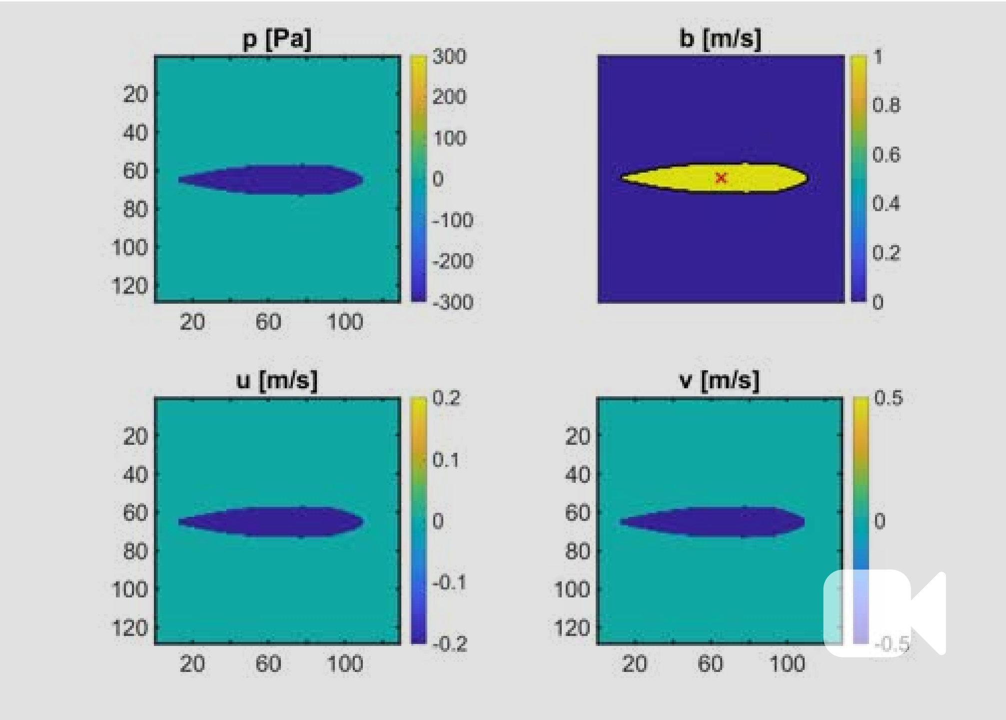
(Ronneberger 2015)



COMSOL FSI
Modified from

(Curatolo & Teresi 2015)





COMSOL FSI
Modified from
(Curatolo & Teresi 2015)

↓

Post Processed training data in MATLAB

128 x 128 image sequence

Using COMSOL for FSI dataset

Using COMSOL for FSI dataset

Fluid

Using COMSOL for FSI dataset

Fluid

$$\underbrace{\rho \frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u} \cdot \nabla)\mathbf{u}}_{\frac{D\mathbf{u}}{DT}} + \nabla p + \underbrace{\nabla \cdot \tau}_{\eta \nabla^2 \mathbf{u}} = \underbrace{\mathbf{F}}_{=\mathbf{0}}$$

Using COMSOL for FSI dataset

Fluid

$$\underbrace{\rho\frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla)\mathbf{u}}_{\frac{D\mathbf{u}}{DT}} + \nabla p + \underbrace{\nabla\cdot\tau}_{\eta\nabla^2\mathbf{u}} = \underbrace{\mathbf{F}}_{=\mathbf{0}}$$

$$\nabla\cdot\mathbf{u} = 0$$

Using COMSOL for FSI dataset

Fluid

$$\underbrace{\rho\frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla)\mathbf{u}}_{\frac{D\mathbf{u}}{DT}} + \nabla p + \underbrace{\nabla\cdot\tau}_{\eta\nabla^2\mathbf{u}} = \underbrace{\mathbf{F}}_{=\mathbf{0}}$$

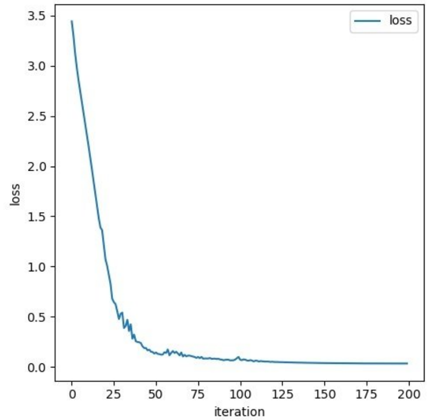
$$\nabla\cdot\mathbf{u} = 0$$

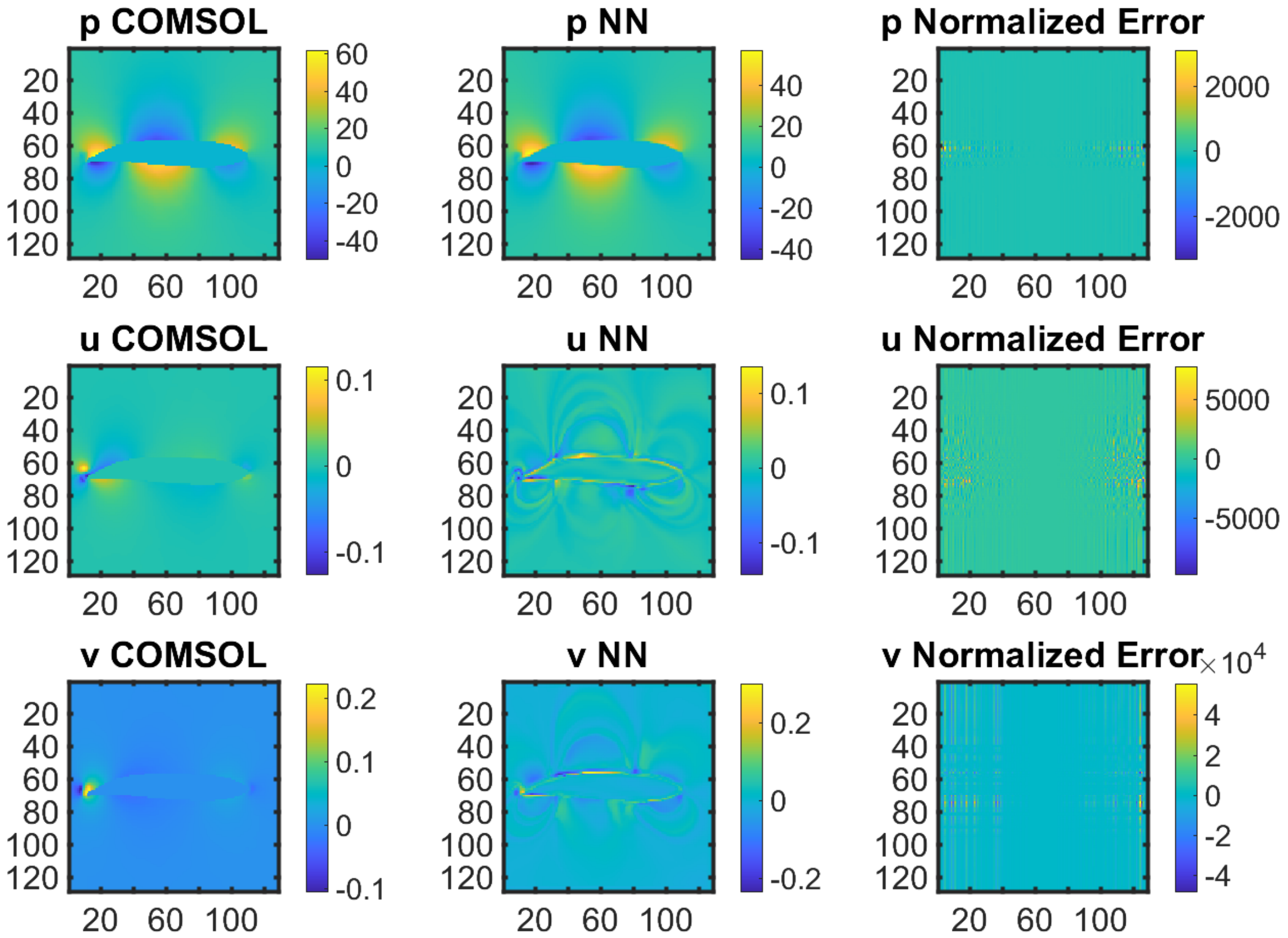
Structure Boundary Condition

$$\mathbf{t} = -\mathbf{n}\cdot(-p\mathbf{I} + \underbrace{\tau}_{\eta\nabla^2\mathbf{u}})$$

Preliminary Results

after 200 epochs



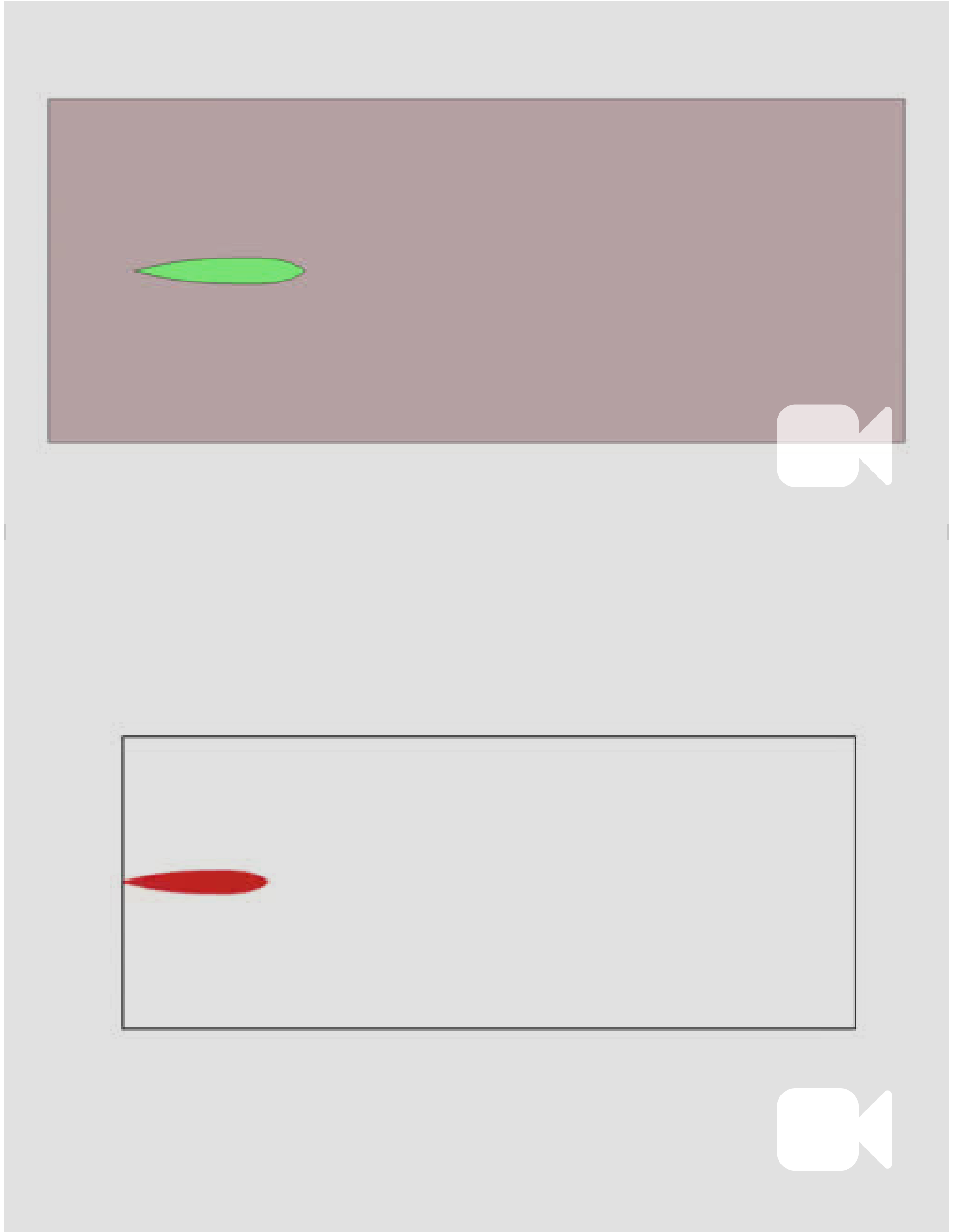


0.58%

66%

156%

The End Goal



Discussion

Parameter	COMSOL FSI	DiffPD
dt	0.01	0.01
frames	200	200
machine	local desktop	google cloud platform
run time	> 2 hours	5.4 seconds

Discussion

Parameter	COMSOL FSI	DiffPD
dt	0.01	0.01
frames	200	200
machine	local desktop	google cloud platform
run time	> 2 hours	5.4 seconds

Use NN as a more sophisticated
differentiable look up table.

Discussion

Parameter	COMSOL FSI	DiffPD
dt	0.01	0.01
frames	200	200
machine	local desktop	google cloud platform
run time	> 2 hours	5.4 seconds

Use NN as a more sophisticated differentiable look up table.

Training takes time, but afterwards simulation will run fast.

Discussion

Parameter	COMSOL FSI	DiffPD
dt	0.01	0.01
frames	200	200
machine	local desktop	google cloud platform
run time	> 2 hours	5.4 seconds

Use NN as a more sophisticated differentiable look up table.

Training takes time, but afterwards simulation will run fast.

Use the hybrid simulator for control, design optimization, fun, etc.

Future Directions

- Find a big computer and train with more data
- Integrate NN output around fish body and use output of network as surrogate hydrodynamics
- Physics-informed Neural Network similar to Raissi 2018 and Wandel 2021
- Consider using just the vorticity instead of the velocities for 2D
- Extend to 3D
- Compare with physical experiments

So long and thanks for all the fish!

Many thanks to Aaron, Courbin, Lisa, Pierre!