

# Wing Design via SAIL

Leonid Scott

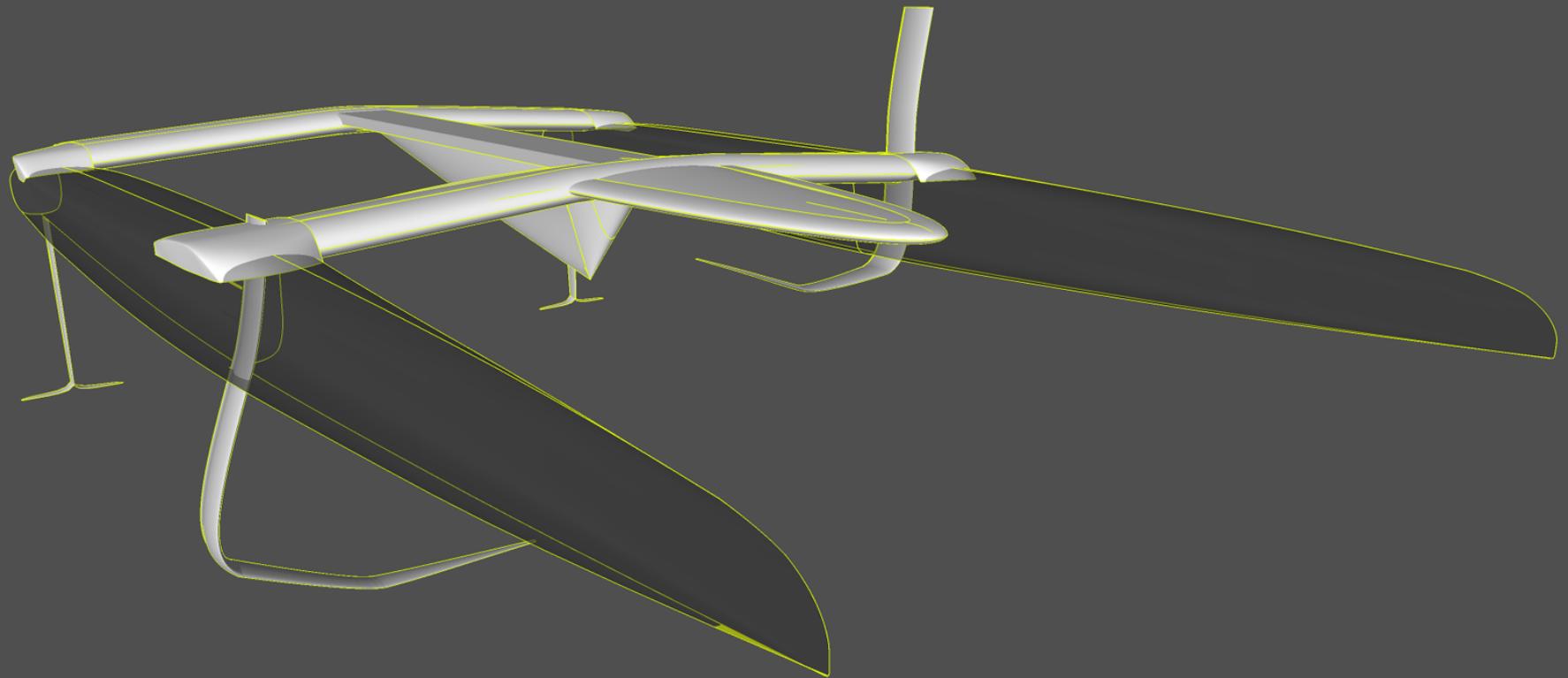
Spring 2019 Computer Science Senior

Seminar

University of Minnesota Morris



<https://www.southernspars.com/wp-content/uploads/2017/06/RG170611AmCupL2R5compressed.jpg>



AC50 Foils - MV / CATSAILINGNEWS.COM

# Foil design is hard

# Foil design is hard

## Naiver Stokes

$$r : \rho \left( \frac{\partial u_r}{\partial t} + u_r \frac{\partial u_r}{\partial r} + \frac{u_\phi}{r \sin(\theta)} \frac{\partial u_r}{\partial \phi} + \frac{u_\theta}{r} \frac{\partial u_r}{\partial \theta} - \frac{u_\phi^2 + u_\theta^2}{r} \right) = -\frac{\partial p}{\partial r} + \rho g_r +$$

$$\mu \left[ \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial u_r}{\partial r} \right) + \frac{1}{r^2 \sin(\theta)^2} \frac{\partial^2 u_r}{\partial \phi^2} + \frac{1}{r^2 \sin(\theta)} \frac{\partial}{\partial \theta} \left( \sin(\theta) \frac{\partial u_r}{\partial \theta} \right) - 2 \frac{u_r + \frac{\partial u_\theta}{\partial \theta} + u_\theta \cot(\theta)}{r^2} - \frac{2}{r^2 \sin(\theta)} \frac{\partial u_\phi}{\partial \phi} \right]$$

$$\phi : \rho \left( \frac{\partial u_\phi}{\partial t} + u_r \frac{\partial u_\phi}{\partial r} + \frac{u_\phi}{r \sin(\theta)} \frac{\partial u_\phi}{\partial \phi} + \frac{u_\theta}{r} \frac{\partial u_\phi}{\partial \theta} + \frac{u_r u_\phi + u_\phi u_\theta \cot(\theta)}{r} \right) = -\frac{1}{r \sin(\theta)} \frac{\partial p}{\partial \phi} + \rho g_\phi +$$

$$\mu \left[ \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial u_\phi}{\partial r} \right) + \frac{1}{r^2 \sin(\theta)^2} \frac{\partial^2 u_\phi}{\partial \phi^2} + \frac{1}{r^2 \sin(\theta)} \frac{\partial}{\partial \theta} \left( \sin(\theta) \frac{\partial u_\phi}{\partial \theta} \right) + \frac{2 \sin(\theta) \frac{\partial u_r}{\partial \phi} + 2 \cos(\theta) \frac{\partial u_\theta}{\partial \phi} - u_\phi}{r^2 \sin(\theta)^2} \right]$$

$$\theta : \rho \left( \frac{\partial u_\theta}{\partial t} + u_r \frac{\partial u_\theta}{\partial r} + \frac{u_\phi}{r \sin(\theta)} \frac{\partial u_\theta}{\partial \phi} + \frac{u_\theta}{r} \frac{\partial u_\theta}{\partial \theta} + \frac{u_r u_\theta - u_\phi^2 \cot(\theta)}{r} \right) = -\frac{1}{r} \frac{\partial p}{\partial \theta} + \rho g_\theta +$$

$$\mu \left[ \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial u_\theta}{\partial r} \right) + \frac{1}{r^2 \sin(\theta)^2} \frac{\partial^2 u_\theta}{\partial \phi^2} + \frac{1}{r^2 \sin(\theta)} \frac{\partial}{\partial \theta} \left( \sin(\theta) \frac{\partial u_\theta}{\partial \theta} \right) + \frac{2}{r^2} \frac{\partial u_r}{\partial \theta} - \frac{u_\theta + 2 \cos(\theta) \frac{\partial u_\phi}{\partial \phi}}{r^2 \sin(\theta)^2} \right].$$

[https://en.wikipedia.org/wiki/Navier%20%93Stokes\\_equations](https://en.wikipedia.org/wiki/Navier%20%93Stokes_equations)

# Foil design is hard

## Naiver Stokes

$$r : \rho \left( \frac{\partial u_r}{\partial t} + u_r \frac{\partial u_r}{\partial r} + \frac{u_\phi}{r \sin(\theta)} \frac{\partial u_r}{\partial \phi} + \frac{u_\theta}{r} \frac{\partial u_r}{\partial \theta} - \frac{u_\phi^2 + u_\theta^2}{r} \right) = -\frac{\partial p}{\partial r} + \rho g_r +$$

$$\mu \left[ \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial u_r}{\partial r} \right) + \frac{1}{r^2 \sin(\theta)^2} \frac{\partial^2 u_r}{\partial \phi^2} + \frac{1}{r^2 \sin(\theta)} \frac{\partial}{\partial \theta} \left( \sin(\theta) \frac{\partial u_r}{\partial \theta} \right) - 2 \frac{u_r + \frac{\partial u_\theta}{\partial \theta} + u_\theta \cot(\theta)}{r^2} - \frac{2}{r^2 \sin(\theta)} \frac{\partial u_\phi}{\partial \phi} \right]$$

$$\phi : \rho \left( \frac{\partial u_\phi}{\partial t} + u_r \frac{\partial u_\phi}{\partial r} + \frac{u_\phi}{r \sin(\theta)} \frac{\partial u_\phi}{\partial \phi} + \frac{u_\theta}{r} \frac{\partial u_\phi}{\partial \theta} + \frac{u_r u_\phi + u_\theta u_\theta \cot(\theta)}{r} \right) = -\frac{1}{r \sin(\theta)} \frac{\partial p}{\partial \phi} + \rho g_\phi +$$

$$\mu \left[ \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial u_\phi}{\partial r} \right) + \frac{1}{r^2 \sin(\theta)^2} \frac{\partial^2 u_\phi}{\partial \phi^2} + \frac{1}{r^2 \sin(\theta)} \frac{\partial}{\partial \theta} \left( \sin(\theta) \frac{\partial u_\phi}{\partial \theta} \right) + \frac{2 \sin(\theta) \frac{\partial u_r}{\partial \phi} + 2 \cos(\theta) \frac{\partial u_\theta}{\partial \phi} - u_\phi}{r^2 \sin(\theta)^2} \right]$$

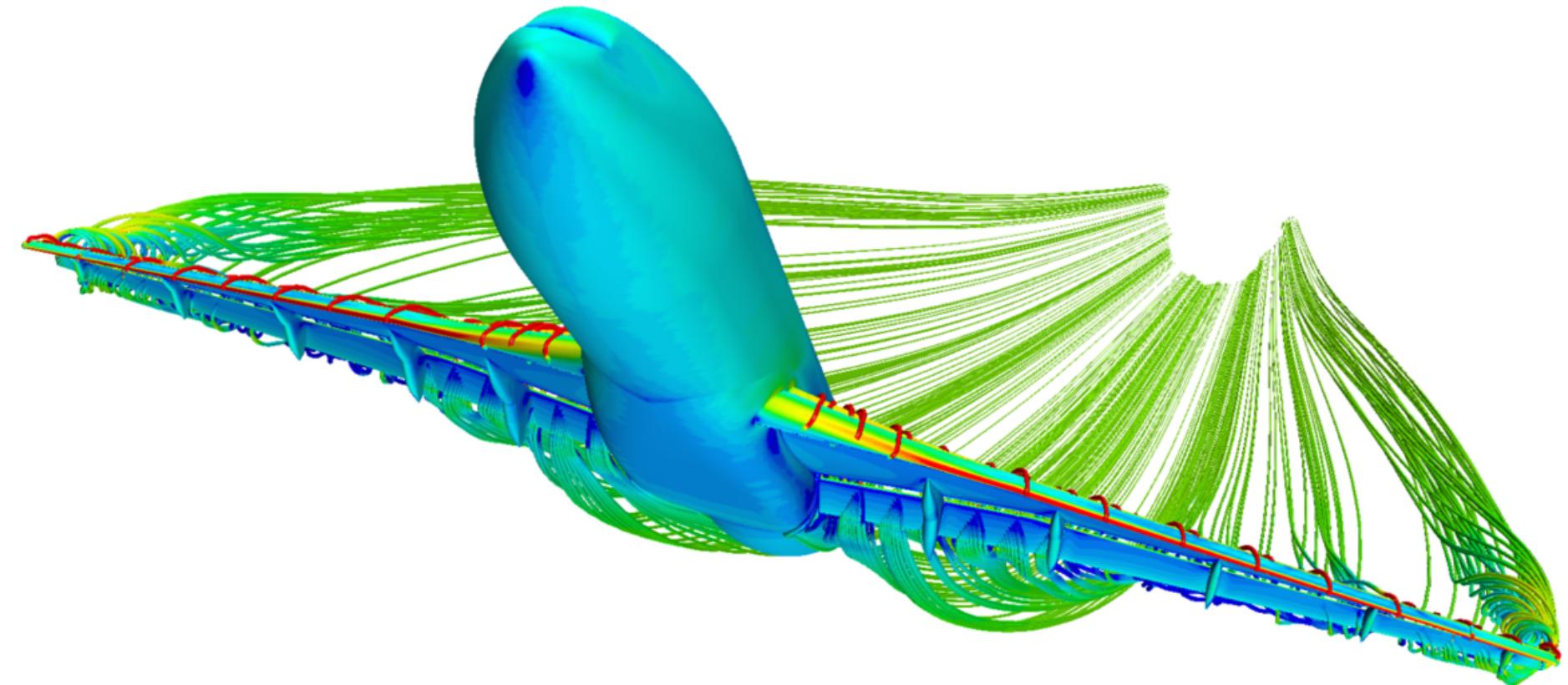
$$\theta : \rho \left( \frac{\partial u_\theta}{\partial t} + u_r \frac{\partial u_\theta}{\partial r} + \frac{u_\phi}{r \sin(\theta)} \frac{\partial u_\theta}{\partial \phi} + \frac{u_\theta}{r} \frac{\partial u_\theta}{\partial \theta} + \frac{u_r u_\theta - u_\phi^2 \cot(\theta)}{r} \right) = -\frac{1}{r} \frac{\partial p}{\partial \theta} + \rho g_\theta +$$

$$\mu \left[ \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial u_\theta}{\partial r} \right) + \frac{1}{r^2 \sin(\theta)^2} \frac{\partial^2 u_\theta}{\partial \phi^2} + \frac{1}{r^2 \sin(\theta)} \frac{\partial}{\partial \theta} \left( \sin(\theta) \frac{\partial u_\theta}{\partial \theta} \right) + \frac{2}{r^2} \frac{\partial u_r}{\partial \theta} - \frac{u_\theta + 2 \cos(\theta) \frac{\partial u_\phi}{\partial \phi}}{r^2 \sin(\theta)^2} \right].$$

[https://en.wikipedia.org/wiki/Navier%20%93Stokes\\_equations](https://en.wikipedia.org/wiki/Navier%20%93Stokes_equations)

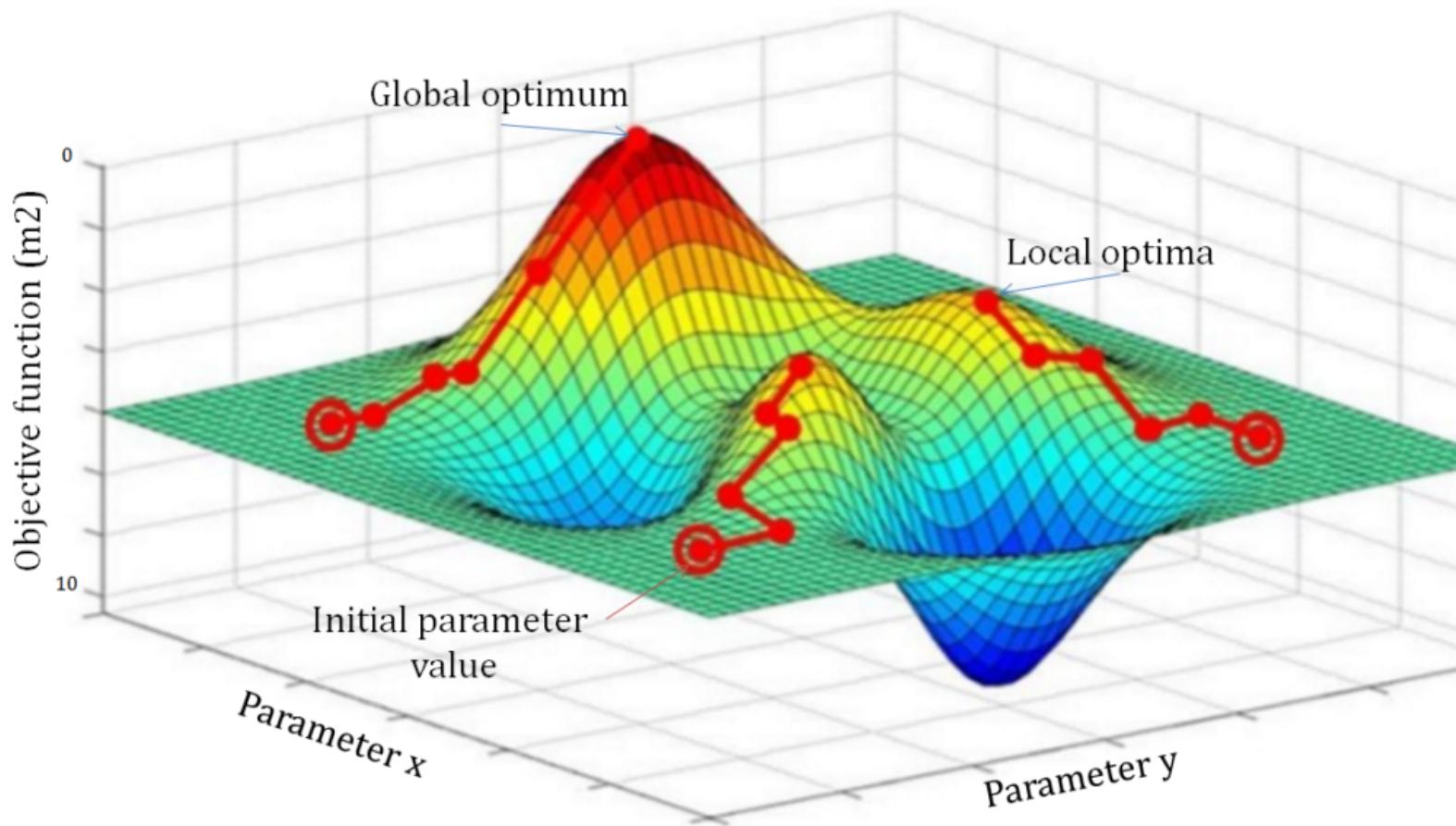
NOPE

# Computational Fluid Dynamics (CFD)



<http://www.oerc.ox.ac.uk/news/ashton-focus>

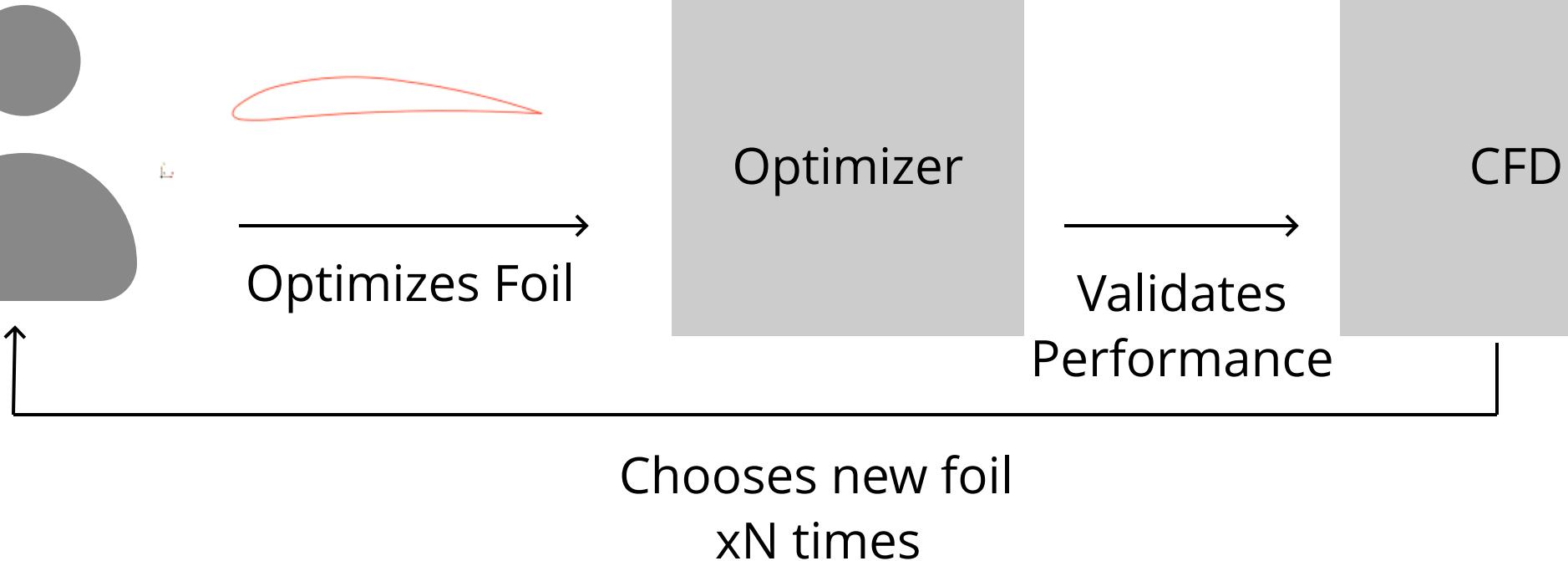
# Optimizers



# timizers

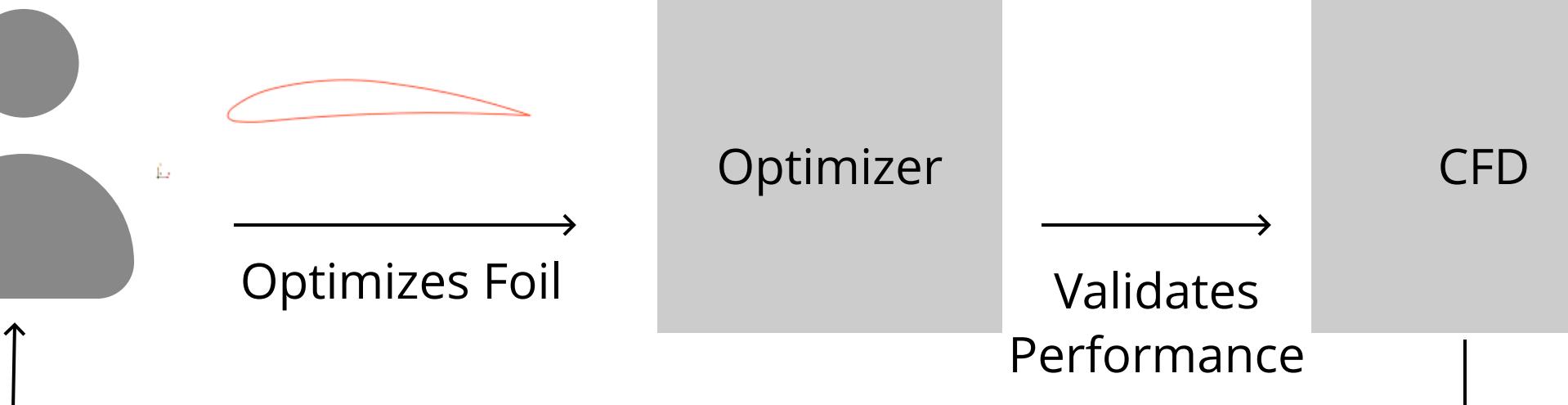
AUTODESK®

## Extended Design Cycle



# Optimizers

## Virtual Design Cycle

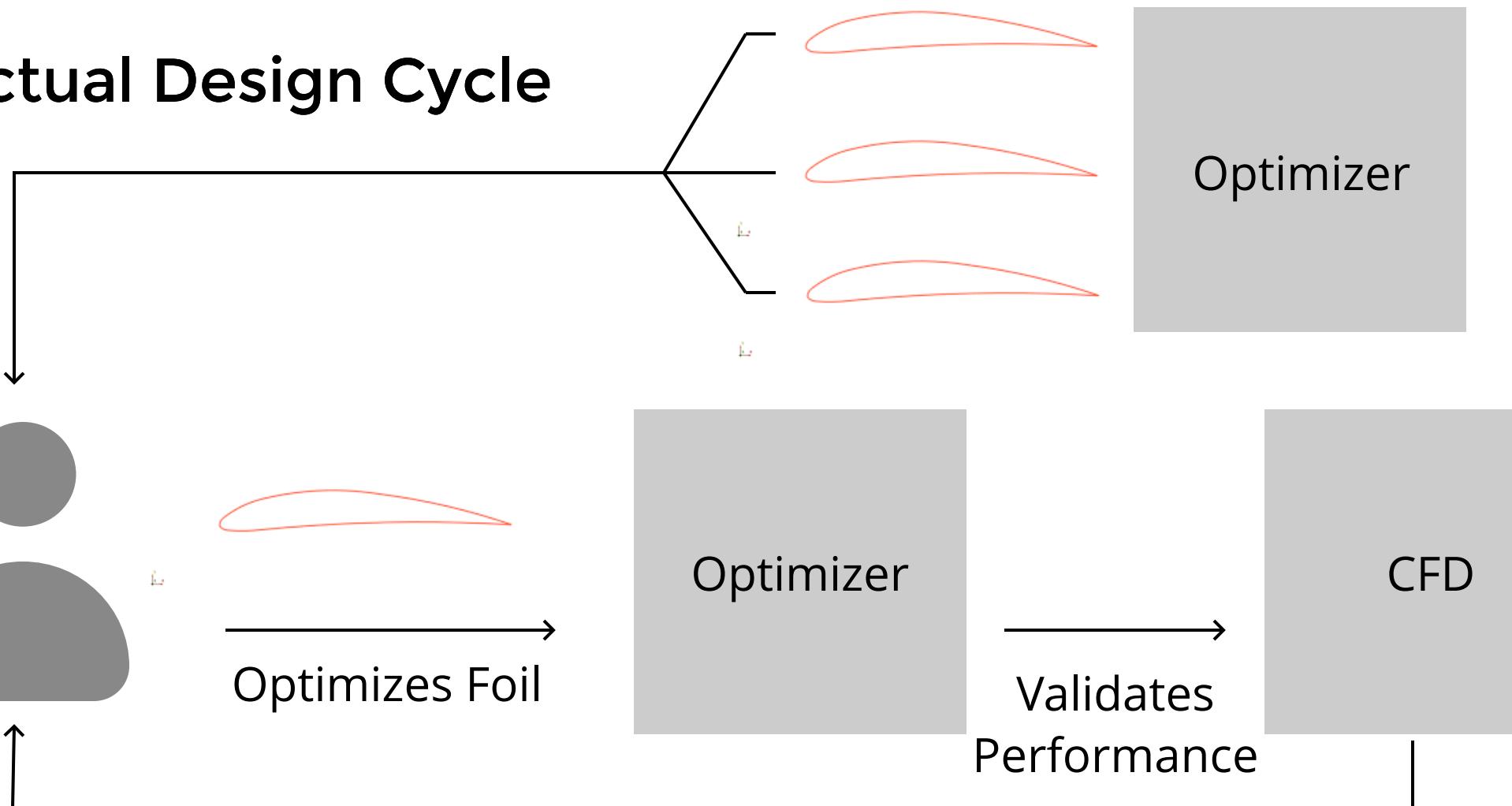


AUTODESK®

Chooses new foil  
xN times

# timizers

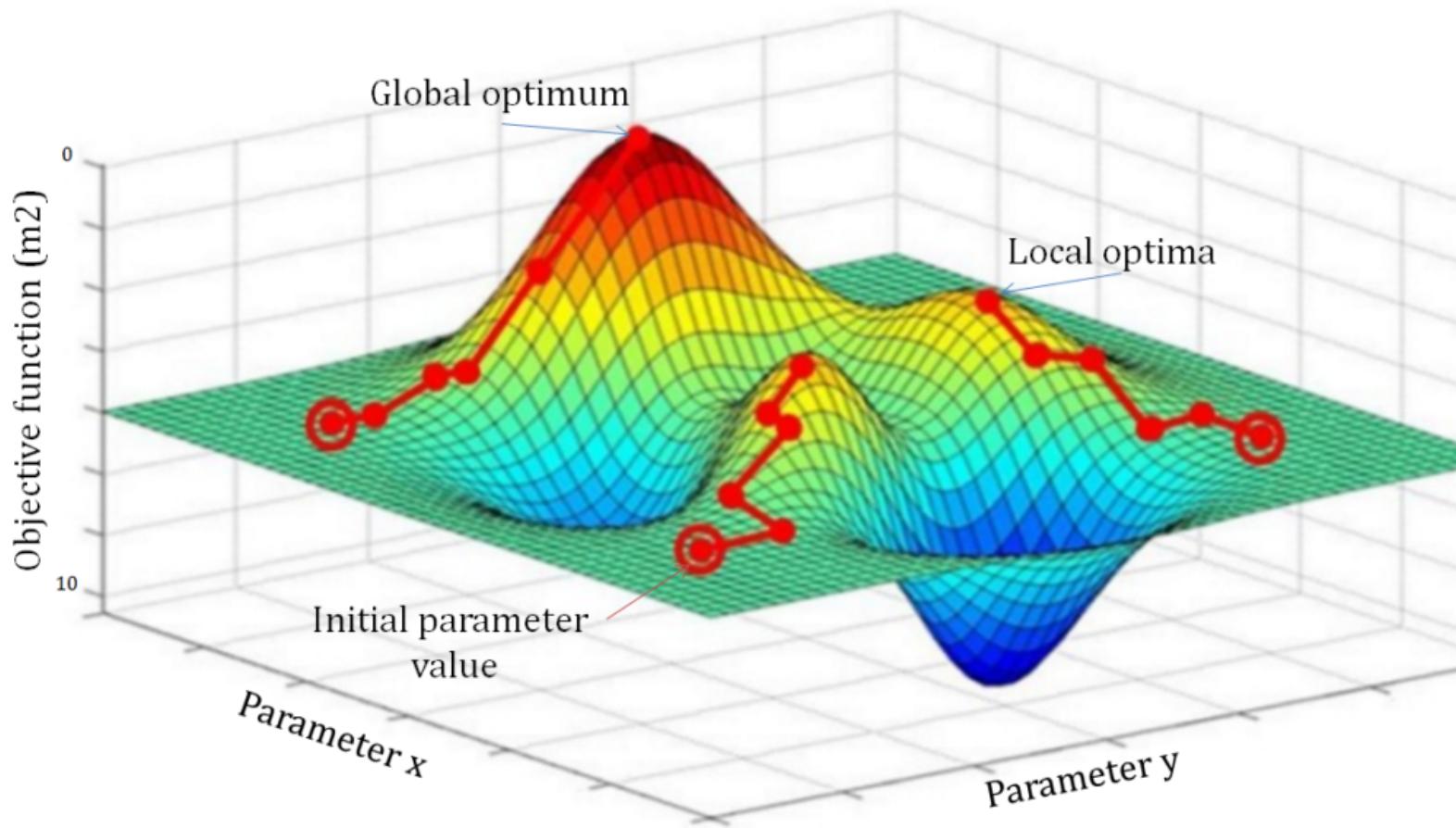
# Actual Design Cycle



AUTODESK®

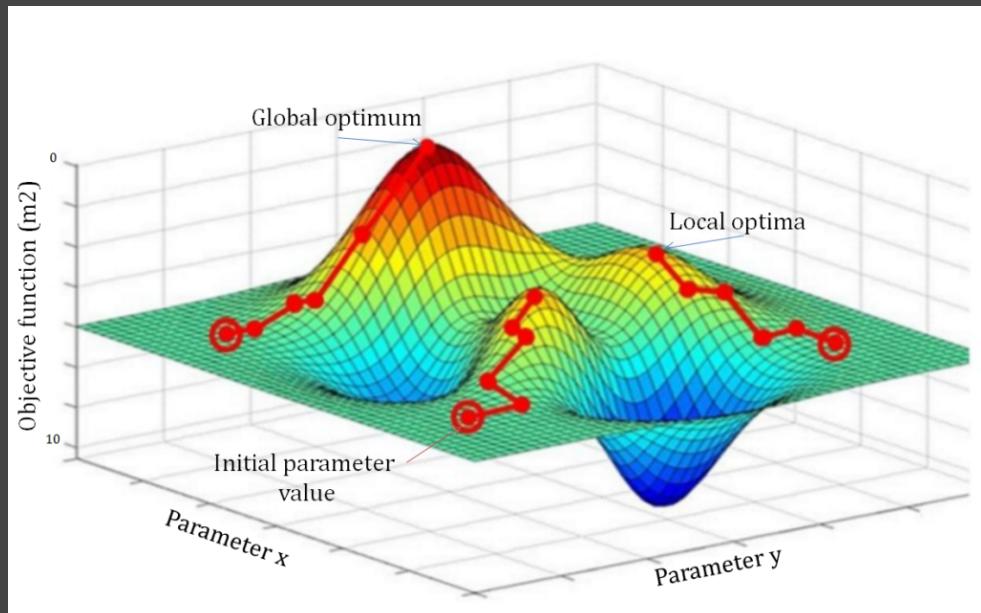
Chooses new foil  
xN times

# Optimizers



# Optimizers

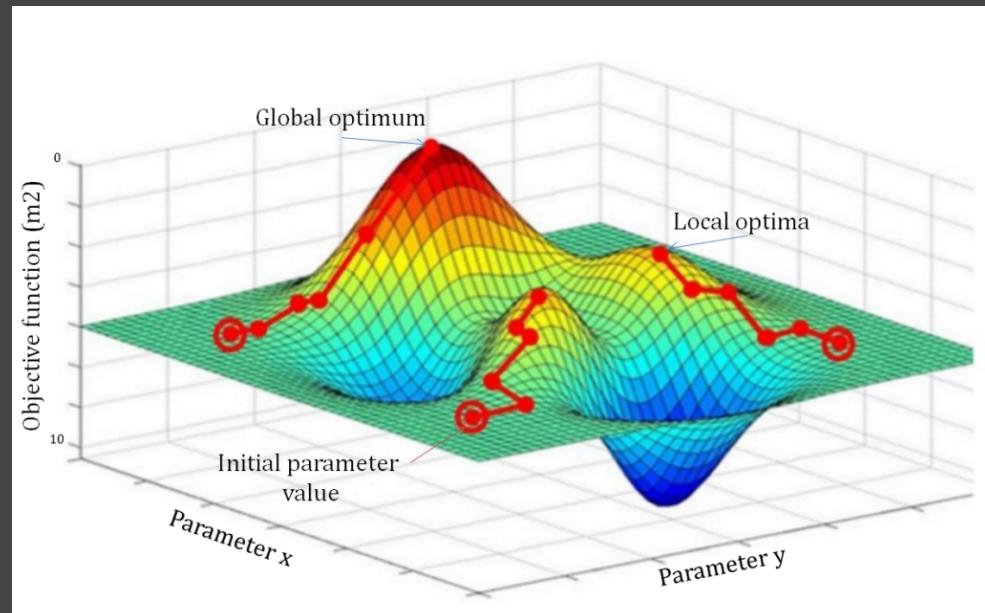
Why use the optimizer this way?



# Optimizers

## Why use the optimizer this way?

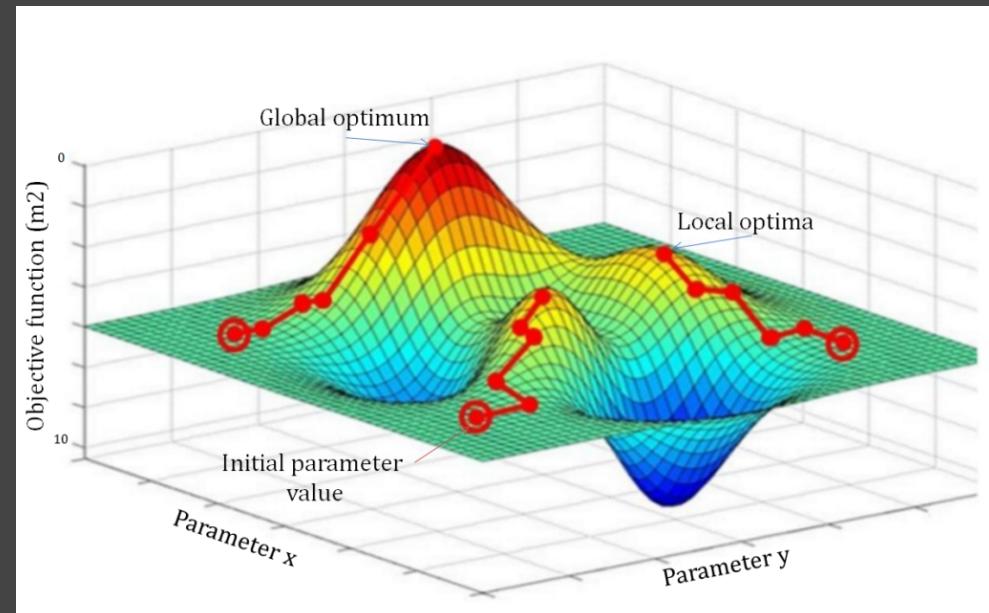
- The global optima might not account for some outside factor



# Optimizers

## Why use the optimizer this way?

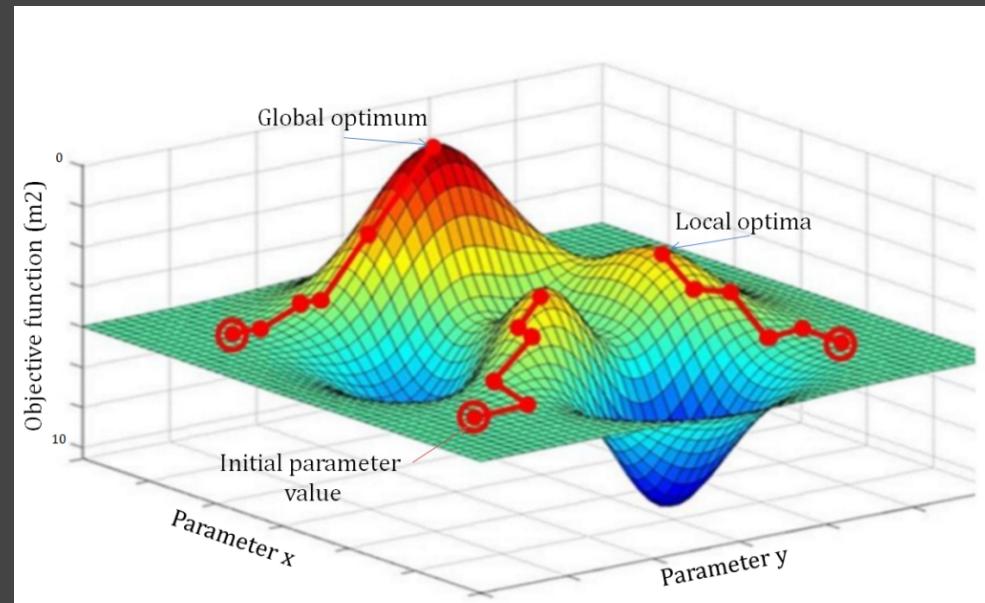
- The global optima might not account for some outside factor
- The optimizer might optimize towards the wrong optima



# Optimizers

## Why use the optimizer this way?

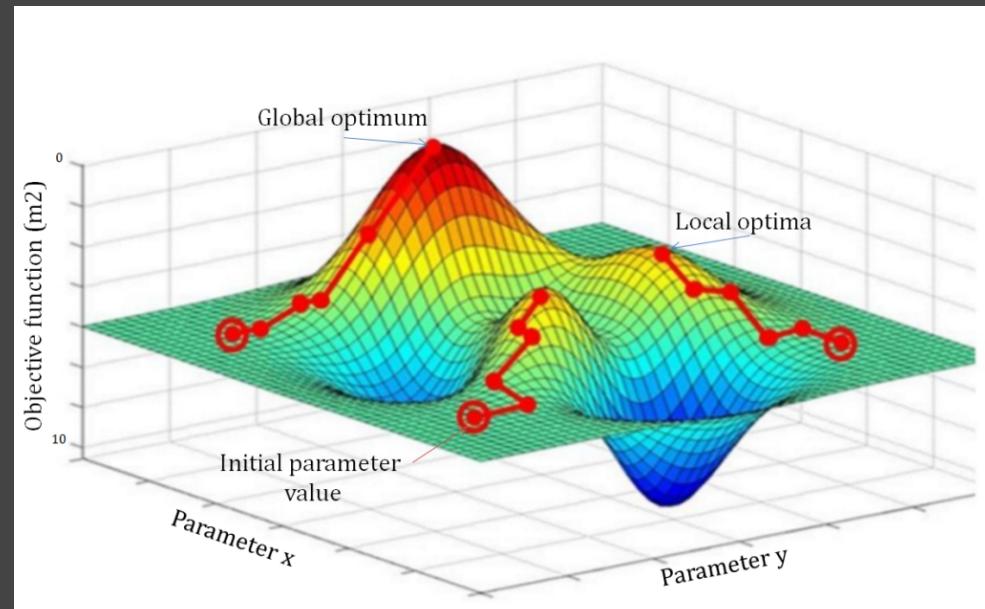
- The global optima might not account for some outside factor
- The optimizer might optimize towards the wrong optima
- Inspiration



# Optimizers

## Why use the optimizer this way?

- The global optima might not account for some outside factor
- The optimizer might optimize towards the wrong optima
- Inspiration



Illumination

# Evolutionary Algorithms

[https://www.youtube.com/embed/bwB6PulBS9A?  
enablejsapi=1&mute=1&autoplay=1](https://www.youtube.com/embed/bwB6PulBS9A?enablejsapi=1&mute=1&autoplay=1)

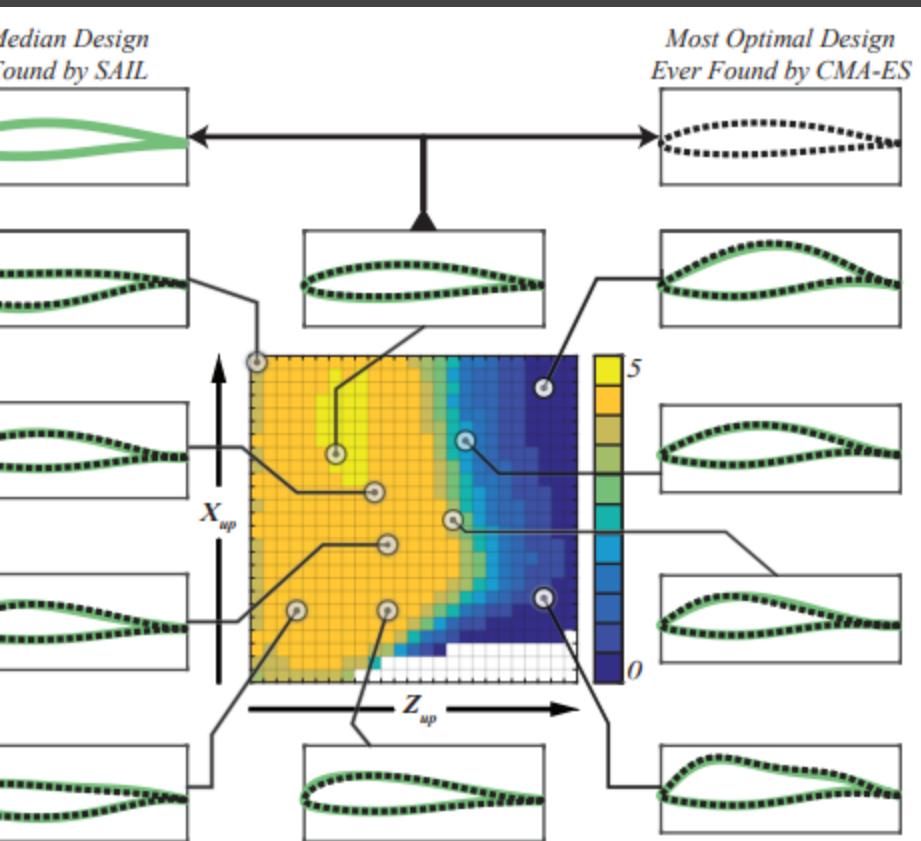
<https://youtu.be/bwB6PulBS9A>

# Evolutionary Algorithms

[https://www.youtube.com/embed/ncyFAmrWgWc?  
start=90&mute=1&enablejsapi=1](https://www.youtube.com/embed/ncyFAmrWgWc?start=90&mute=1&enablejsapi=1)

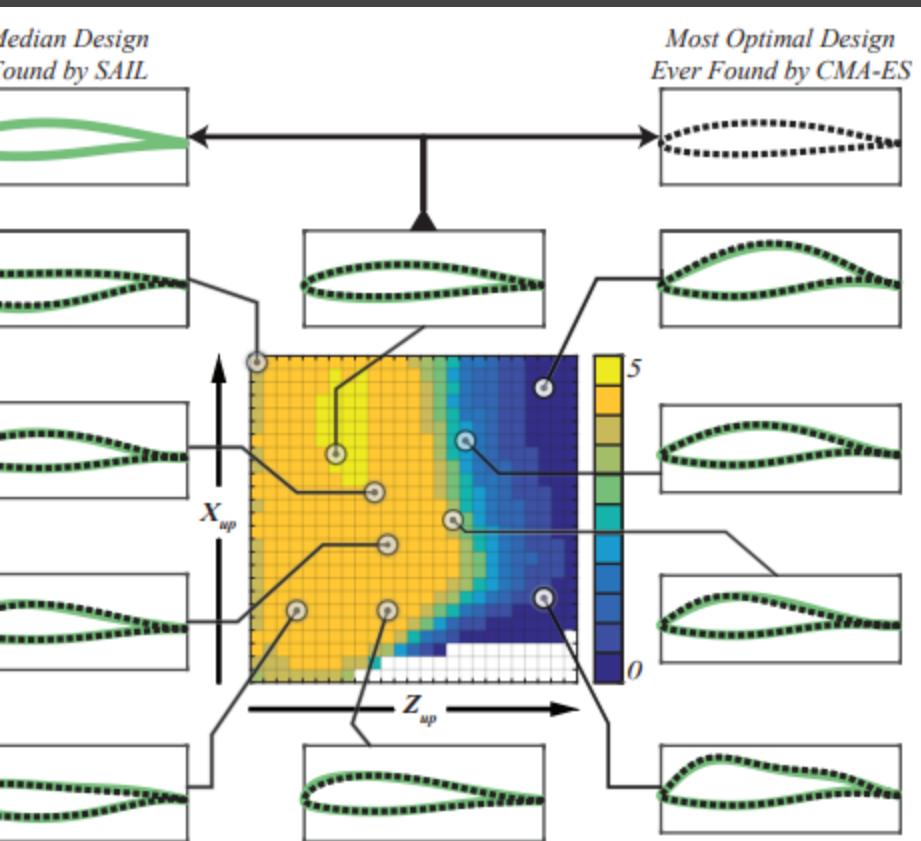
[https://youtu.be/bwB6PulBS9  
A  
https://youtu.be/ncyFAmrWgWc?  
t=90](https://youtu.be/bwB6PulBS9A?t=90)

# Data-Efficient Design Exploration through Surrogate- Assisted Illumination



SAIL is  
designed to:

# Data-Efficient Design Exploration through Surrogate-Assisted Illumination

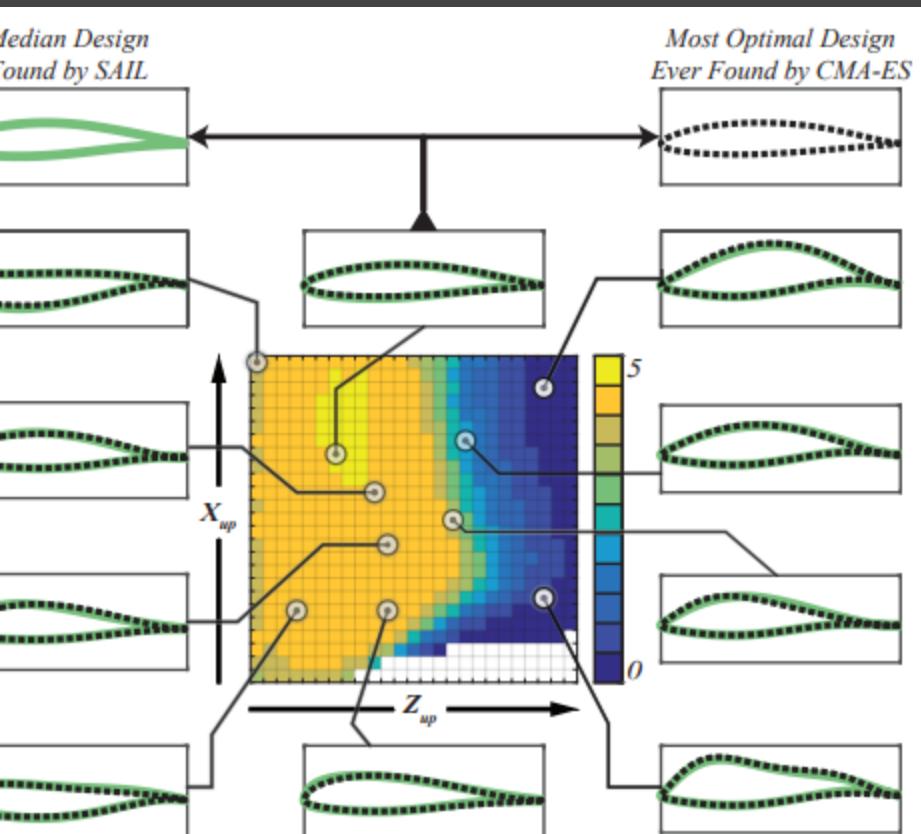


Taken from [1]

SAIL is  
designed to:

- Illuminate the problem space

# Data-Efficient Design Exploration through Surrogate-Assisted Illumination

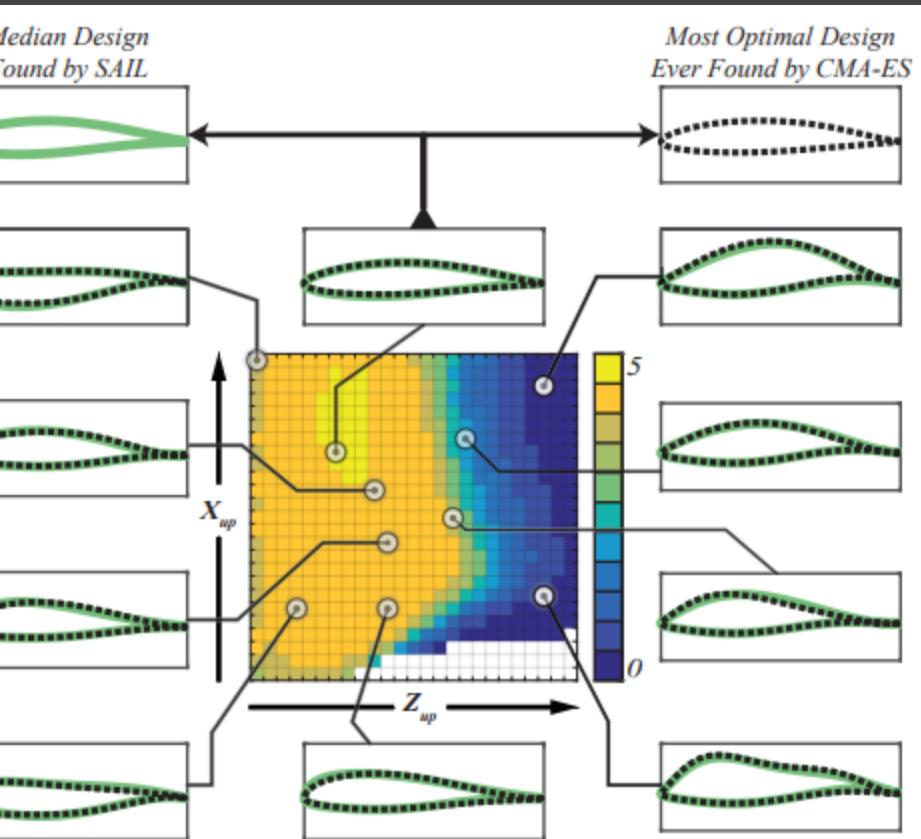


Taken from [1]

SAIL is  
designed to:

- Illuminate the problem space
- Be Data Efficient

# Data-Efficient Design Exploration through Surrogate-Assisted Illumination



SAIL is  
designed to:

- Illuminate the problem space
- Be Data Efficient
- Accurately model the underlying function

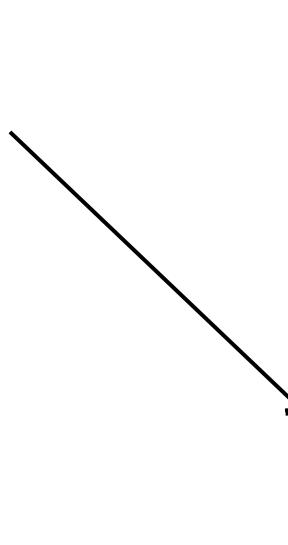
# SAIL

# SAIL

**MAP-Elites**  
For Illumination

**Bayesian Optimization**  
Quality Control of Gaussian Process

**Gaussian Processes**  
to model problem space



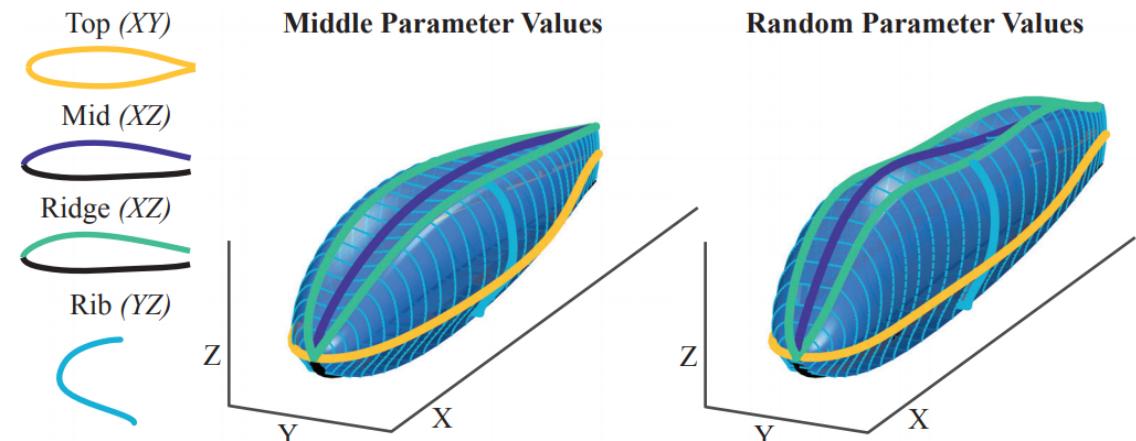
```
graph TD; A[MAP-Elites  
For Illumination] --> C[Bayesian Optimization  
Quality Control of Gaussian Process]; B[Gaussian Processes  
to model problem space] --> C;
```

A diagram illustrating the components of SAIL. At the top left is the acronym "SAIL". Below it, two main components are shown: "MAP-Elites For Illumination" on the left and "Gaussian Processes to model problem space" on the right. Arrows point from both of these components upwards towards the word "Bayesian" at the top, indicating their relationship to the overall framework.

Bayesian  
Optimization  
Quality Control of  
Gaussian Process

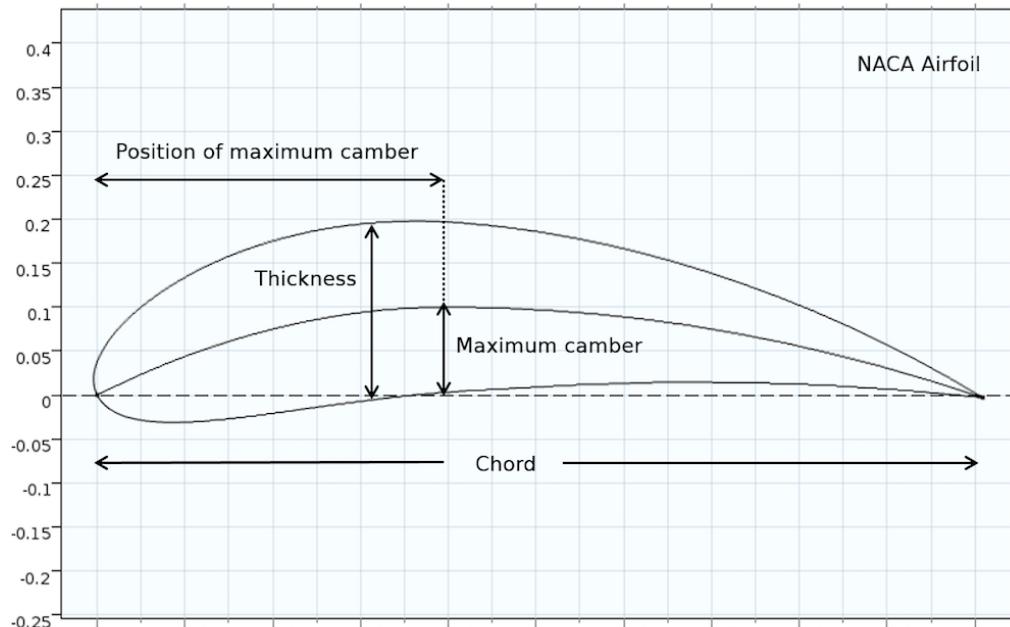
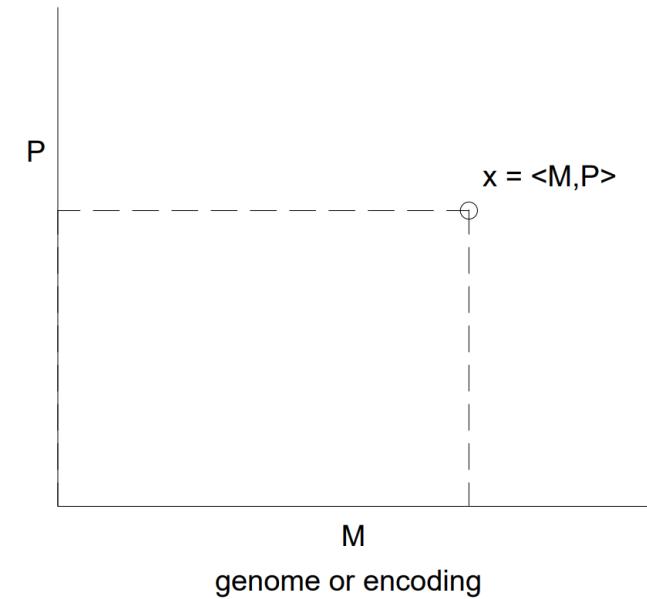
## MAP-Elites For Illumination

Gaussian  
Processes  
to model problem  
space



Taken from [1]

# MAP-Elites Illumination

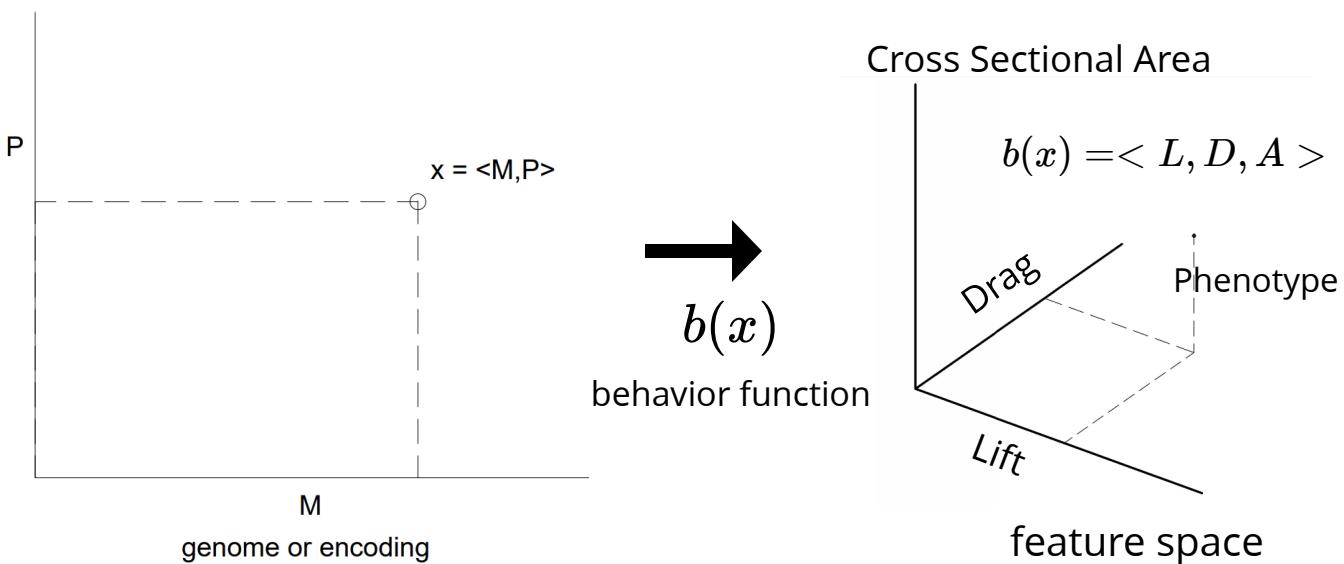


## Genomes

How we express a physical object in a compact form

<https://www.comsol.com/blogs/optimize-naca-airfoil-designs-with-a-simulation-app/>

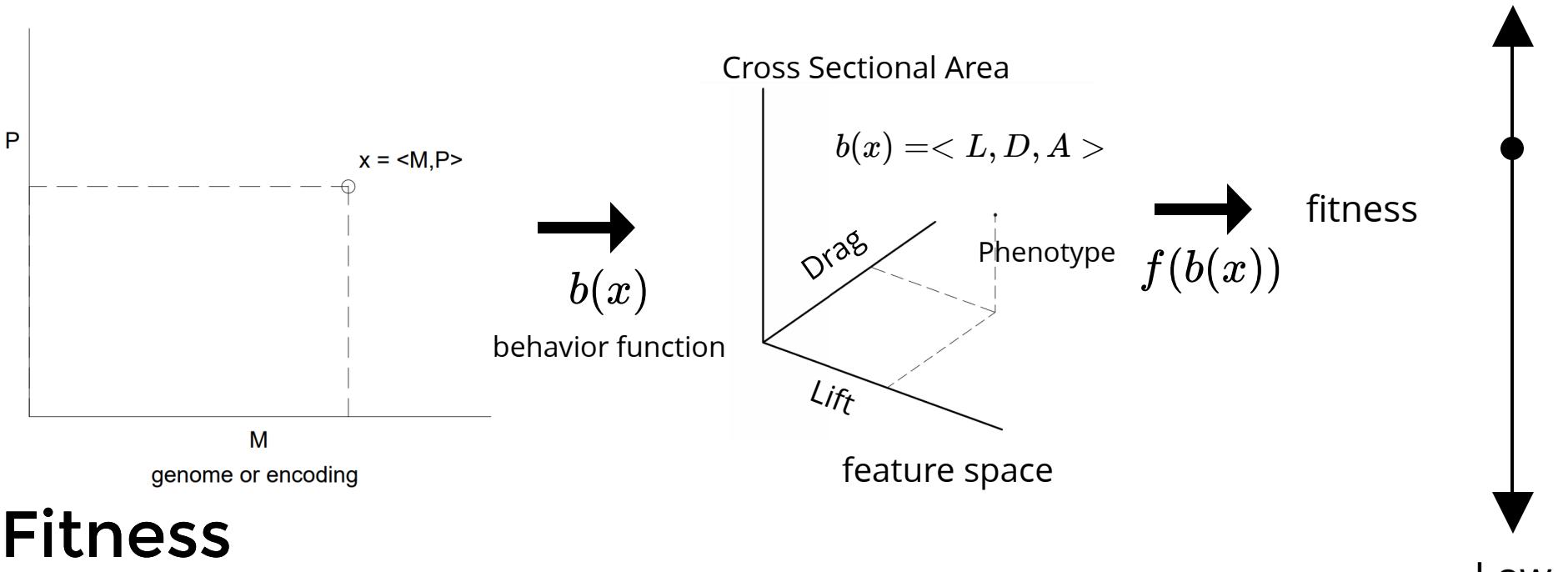
# MAP-Elites Illumination



## Phenotypes

The characteristics of an individual

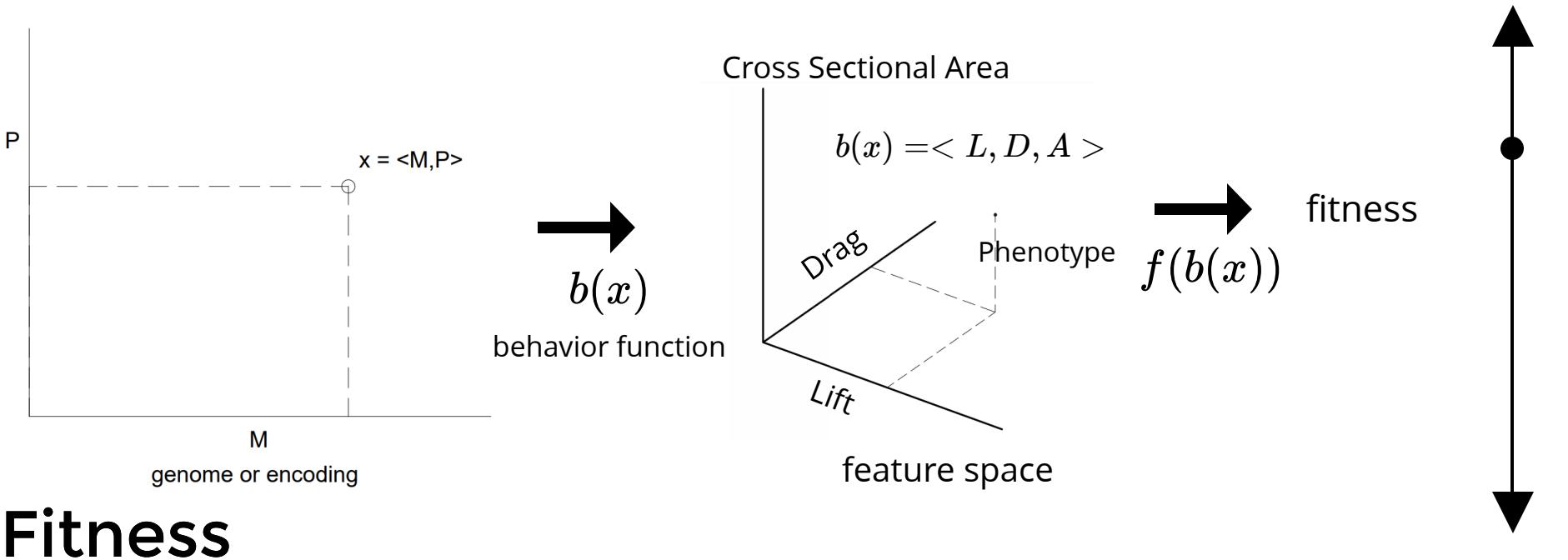
# MAP-Elites Illumination



## Fitness

An assessment of "Goodness"

# MAP-Elites Illumination

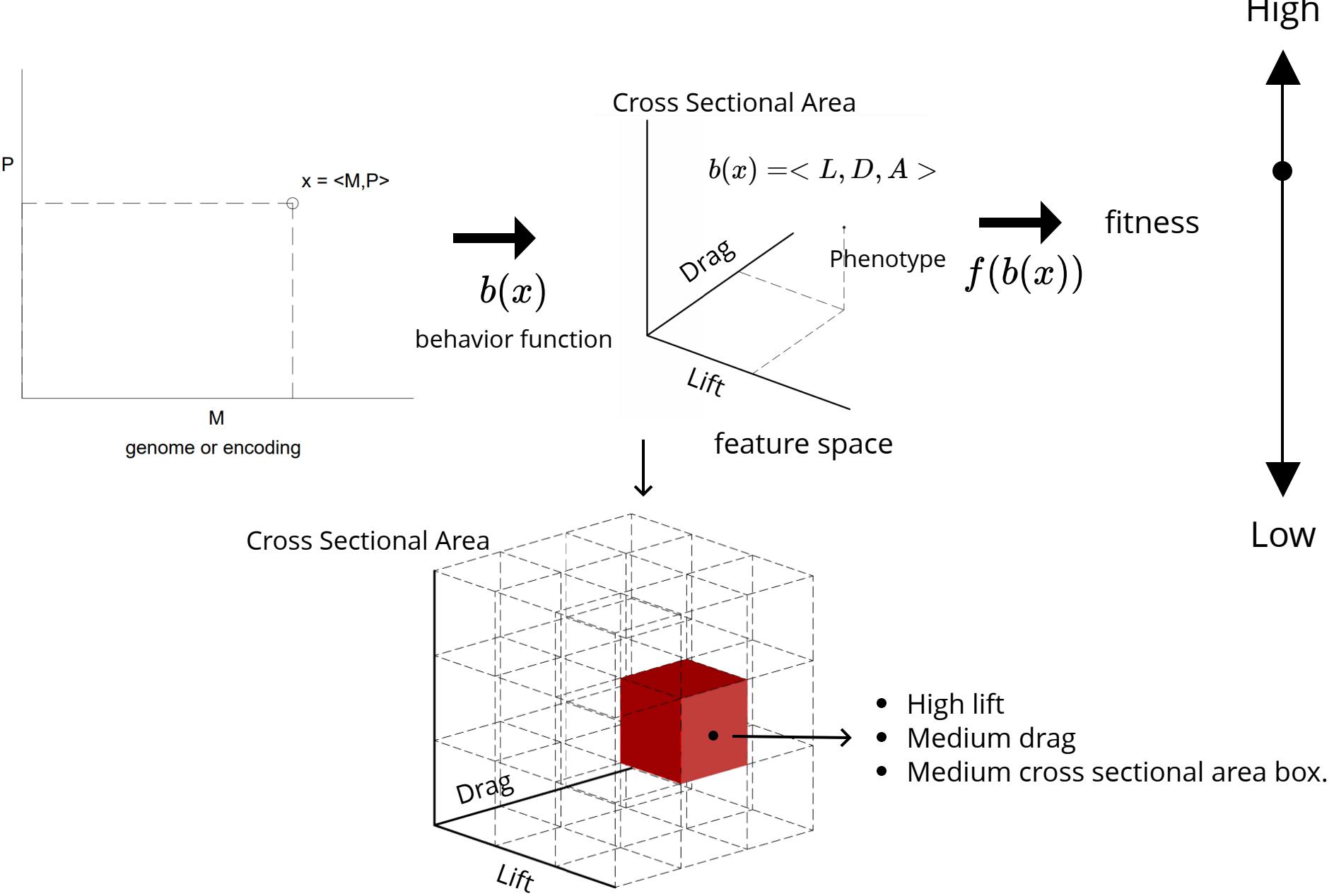


## Fitness

An assessment of "Goodness"

$$f(x) = Lift(x) - b * Drag(x) + c * Area(x)$$

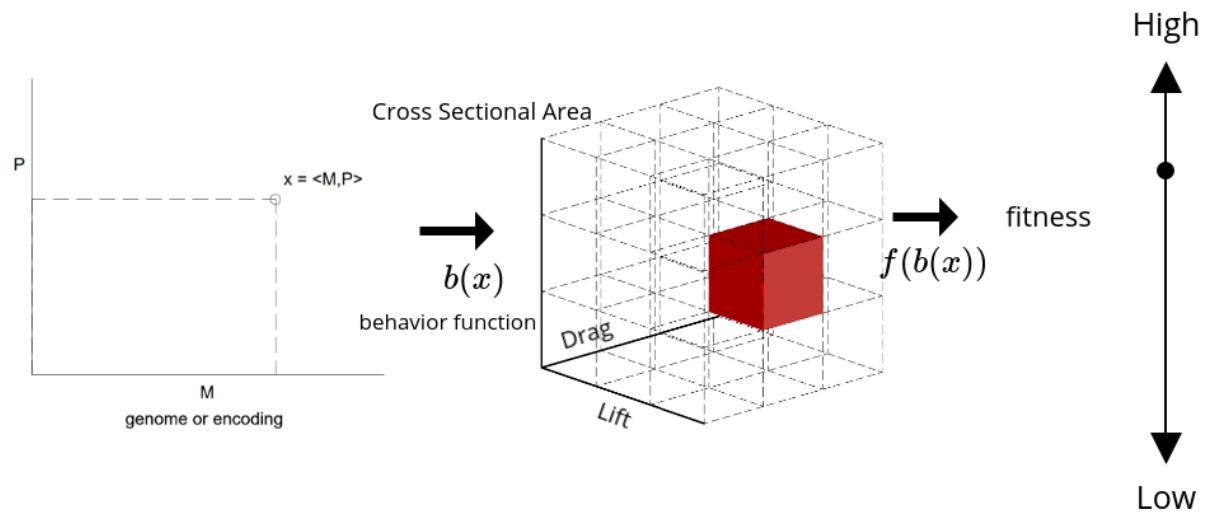
# MAP-Elites Illumination



# MAP-Elites Illumination

# MAP-Elites Illumination

## Setup



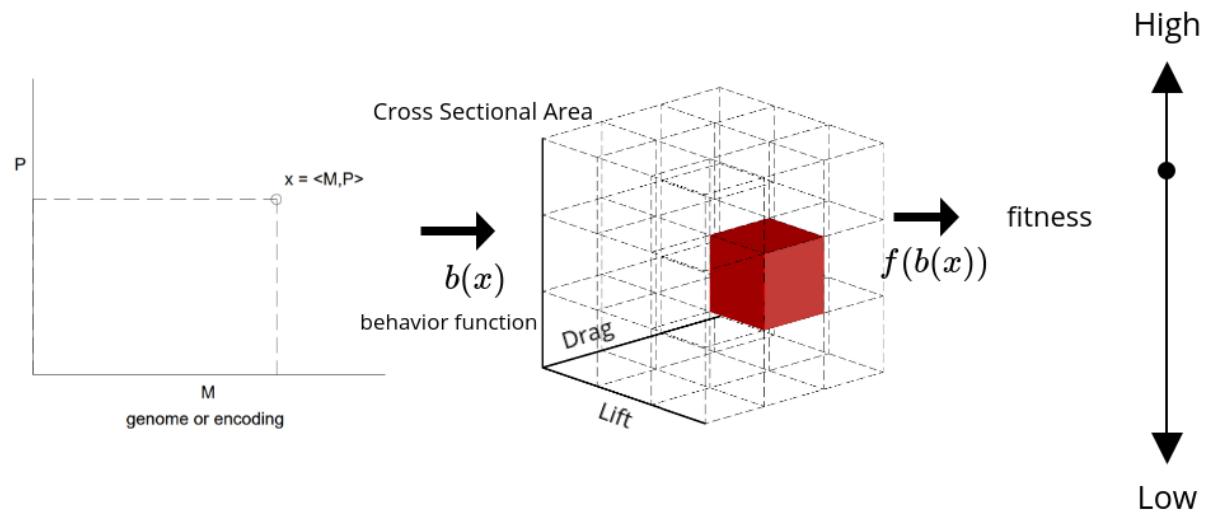
## Evolution

## Termination

# MAP-Elites Illumination

## Setup

- Create random set of solutions,  $X$



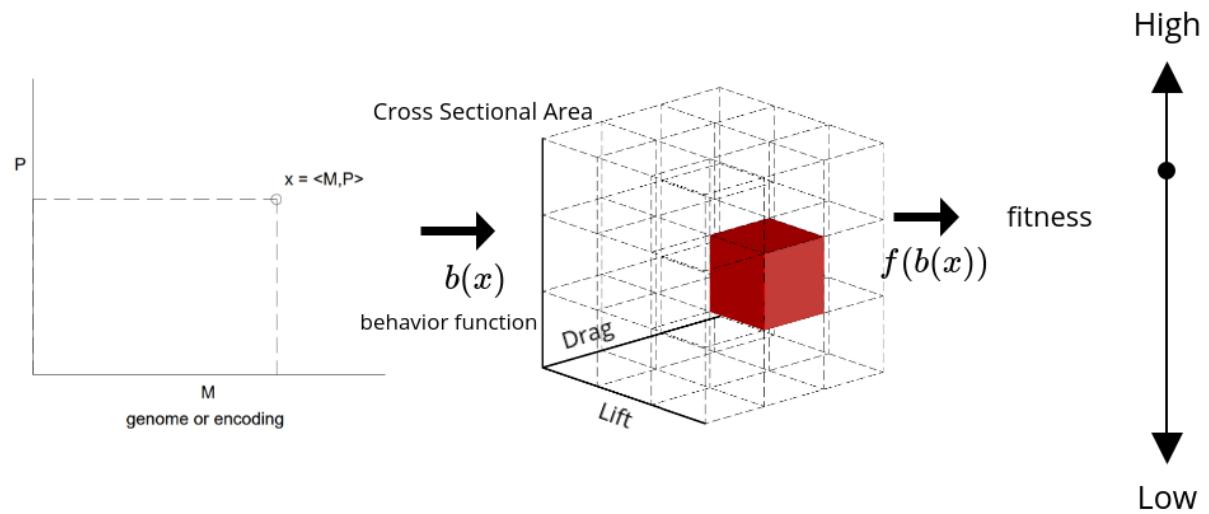
## Evolution

## Termination

# MAP-Elites Illumination

## Setup

- Create random set of solutions,  $X$
- Add them to the feature space



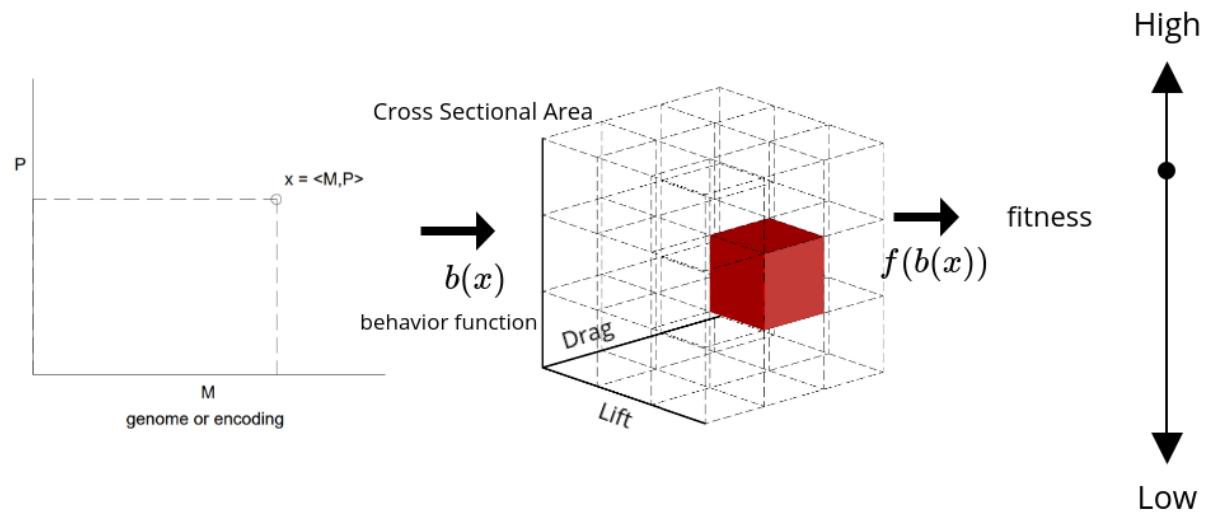
## Evolution

## Termination

# MAP-Elites Illumination

## Setup

- Create random set of solutions,  $X$
- Add them to the feature space



## Evolution

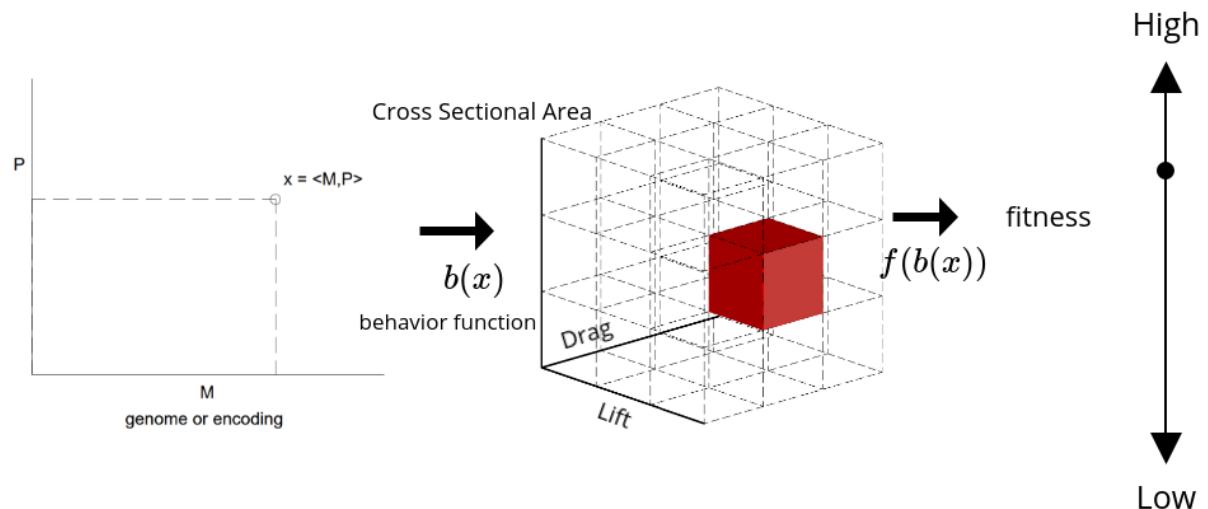
## Feature Space Outcomes

## Termination

# MAP-Elites Illumination

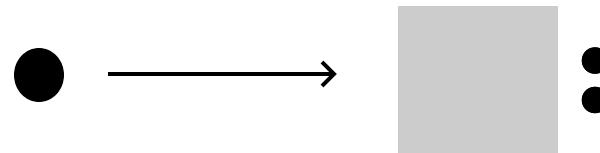
## Setup

- Create random set of solutions,  $X$
- Add them to the feature space



## Evolution

## Feature Space Outcomes

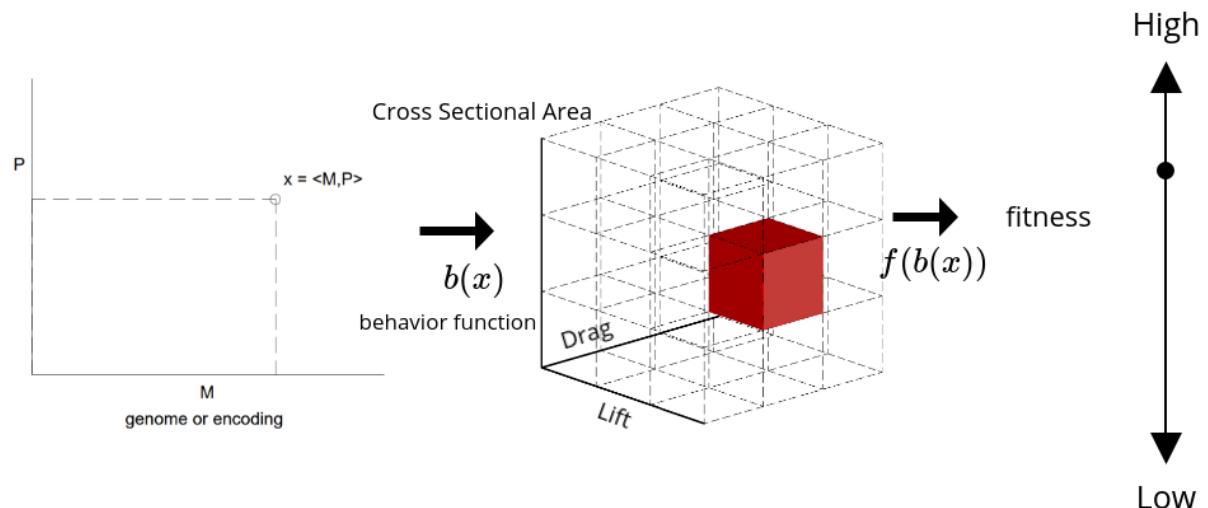


## Termination

# MAP-Elites Illumination

## Setup

- Create random set of solutions,  $X$
- Add them to the feature space



## Evolution

## Feature Space Outcomes

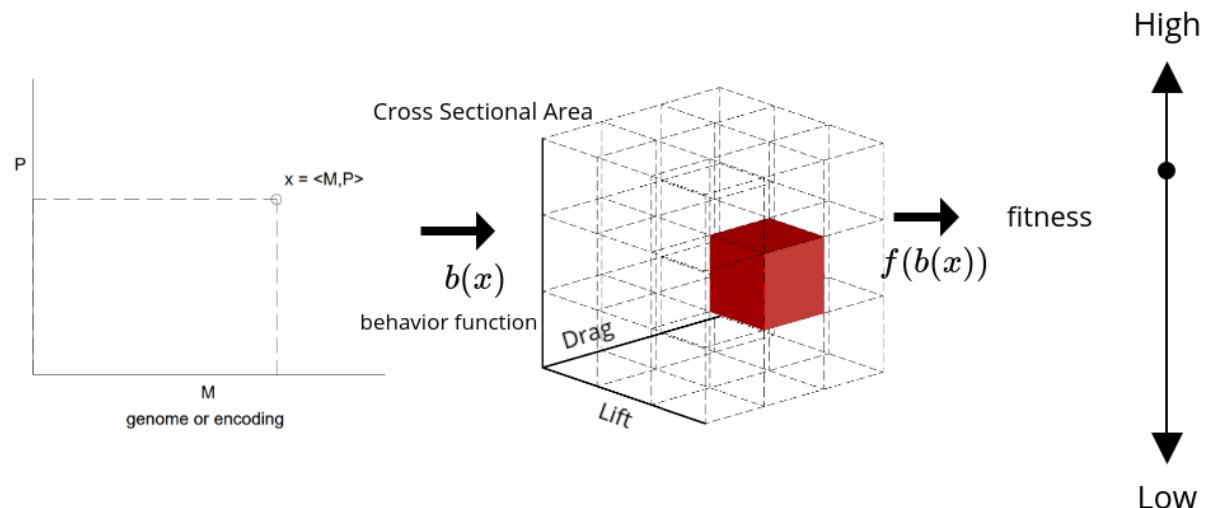


## Termination

# MAP-Elites Illumination

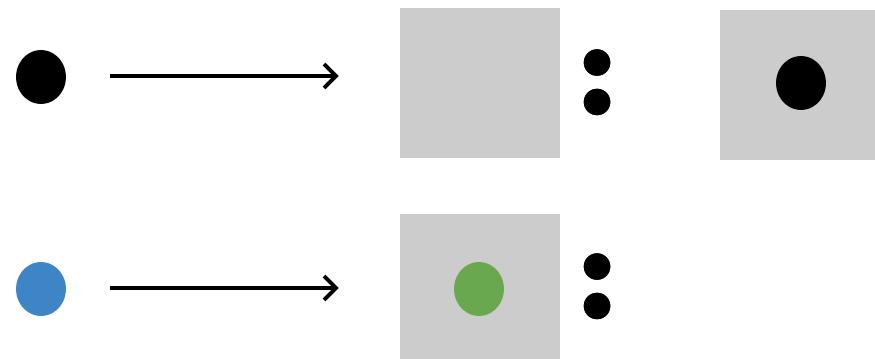
## Setup

- Create random set of solutions,  $X$
- Add them to the feature space



## Evolution

## Feature Space Outcomes



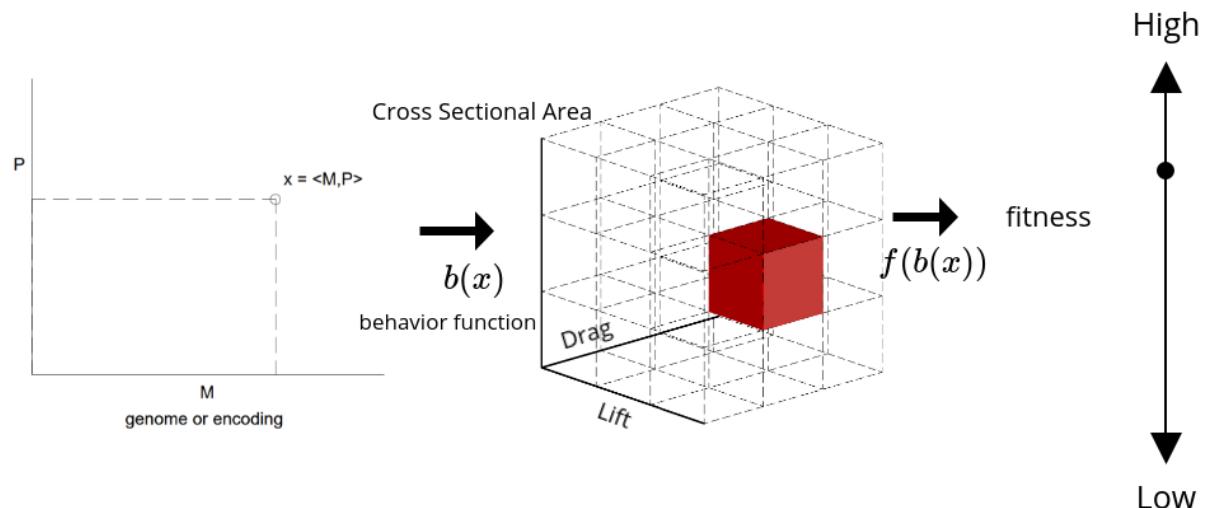
$$F(\text{Blue Circle}) \leq F(\text{Green Circle})$$

## Termination

# MAP-Elites Illumination

## Setup

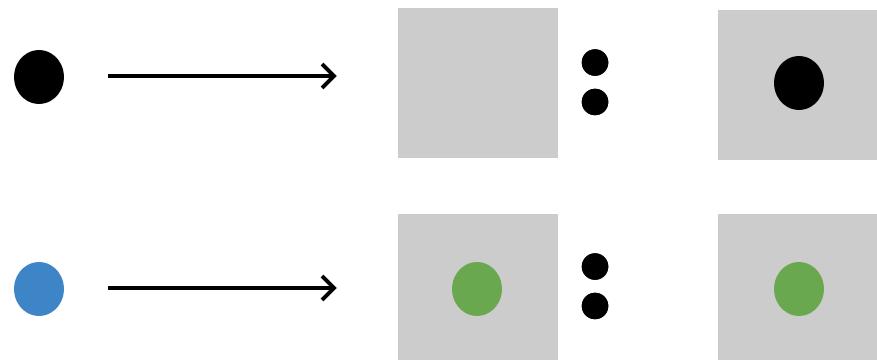
- Create random set of solutions,  $X$
- Add them to the feature space



## Evolution

## Feature Space Outcomes

## Termination

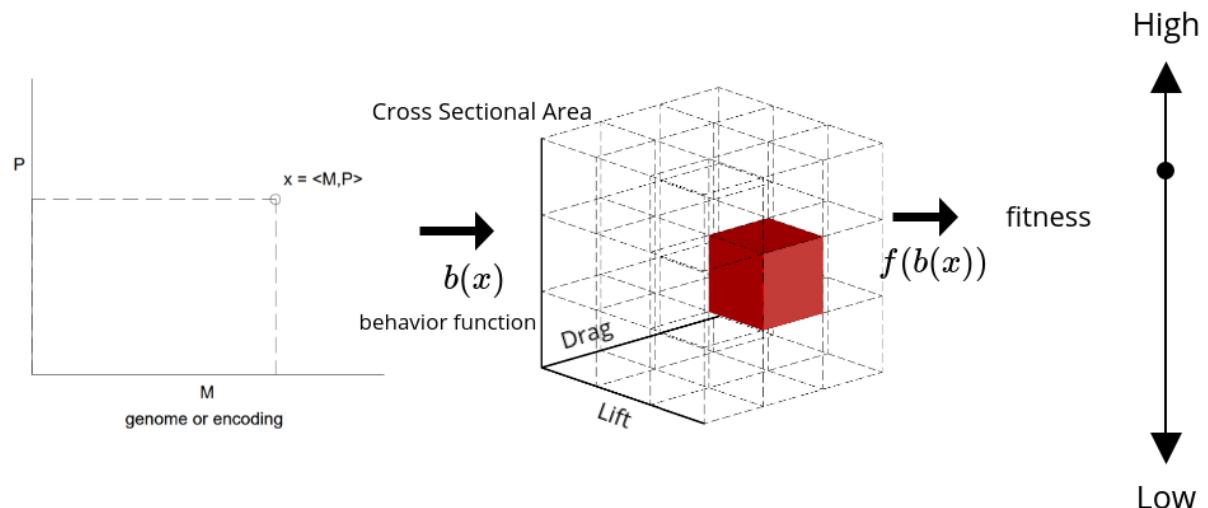


$$F(\text{Blue}) \leq F(\text{Green})$$

# MAP-Elites Illumination

## Setup

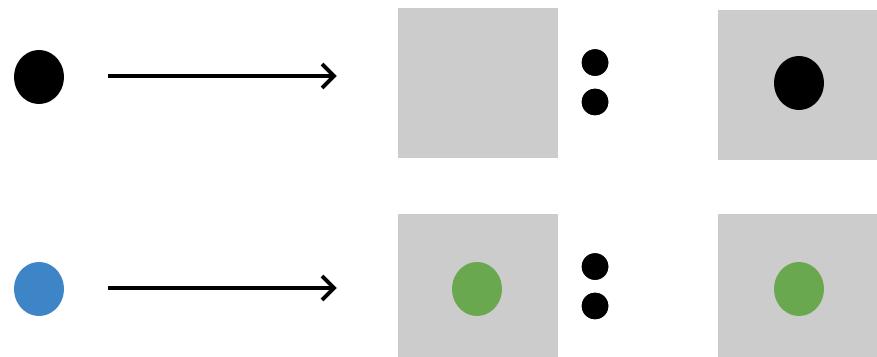
- Create random set of solutions,  $X$
- Add them to the feature space



## Evolution

```
1 for generation in I
2     x <- random_individual(X)
3     x' <- random_variation(x)
4     b' <- behavior_function(x')
5     feature_space_insert(x', b')
```

## Feature Space Outcomes



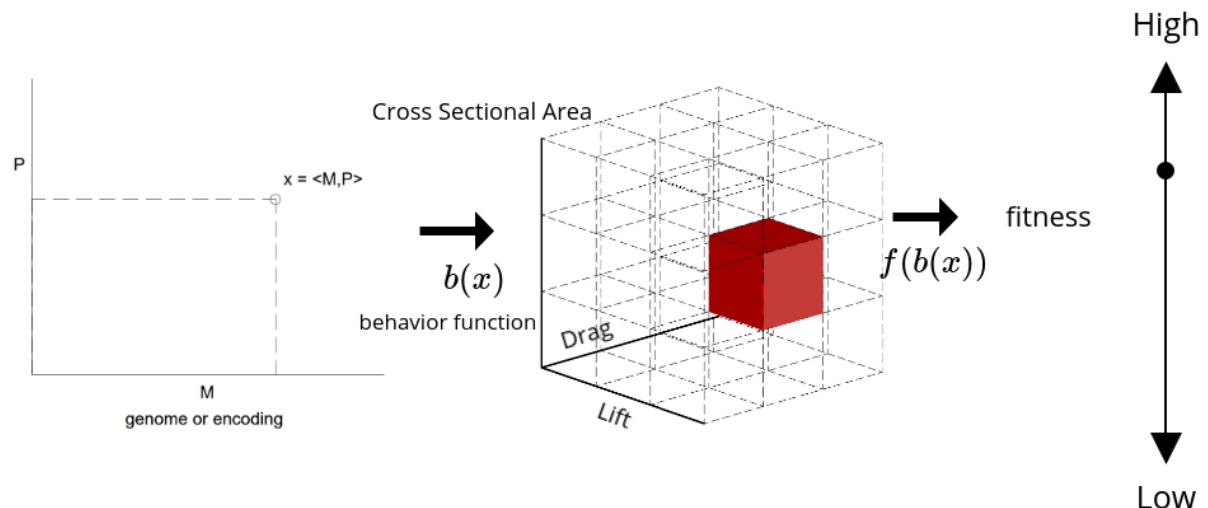
$$F(\text{Blue Circle}) \leq F(\text{Green Circle})$$

## Termination

# MAP-Elites Illumination

## Setup

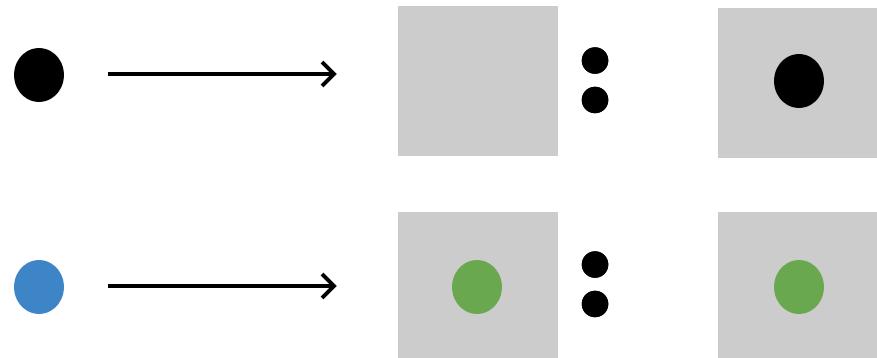
- Create random set of solutions,  $X$
- Add them to the feature space



## Evolution

```
1 for generation in I
2     x <- random_individual(X)
3     x' <- random_variation(x)
4     b' <- behavior_function(x')
5     feature_space_insert(x', b')
```

## Feature Space Outcomes



$$F(\bullet) \leq F(\circ)$$

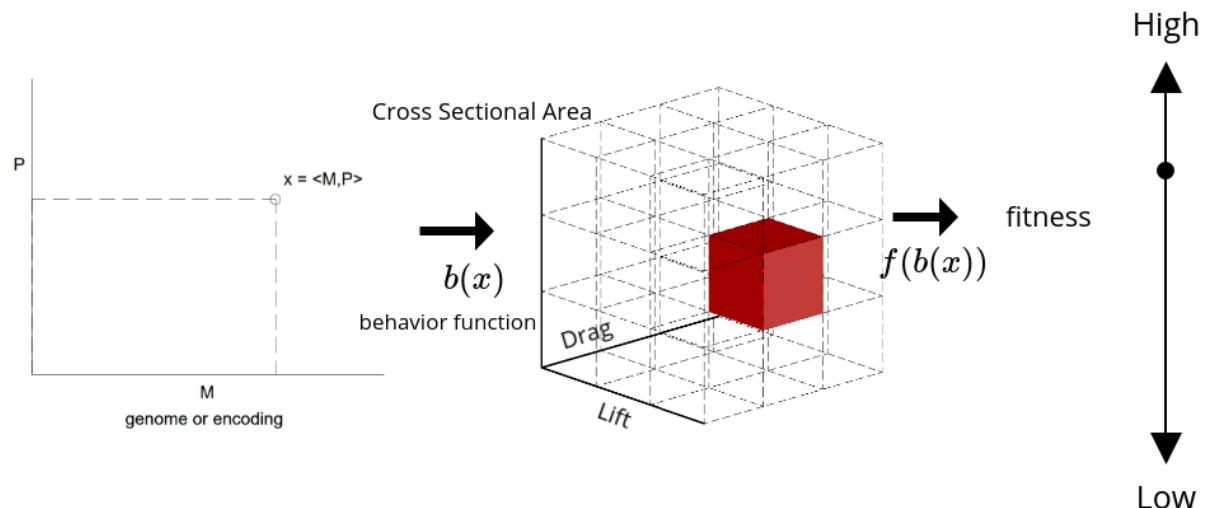
## Termination

- Termination condition is reached

# MAP-Elites Illumination

## Setup

- Create random set of solutions,  $X$
- Add them to the feature space



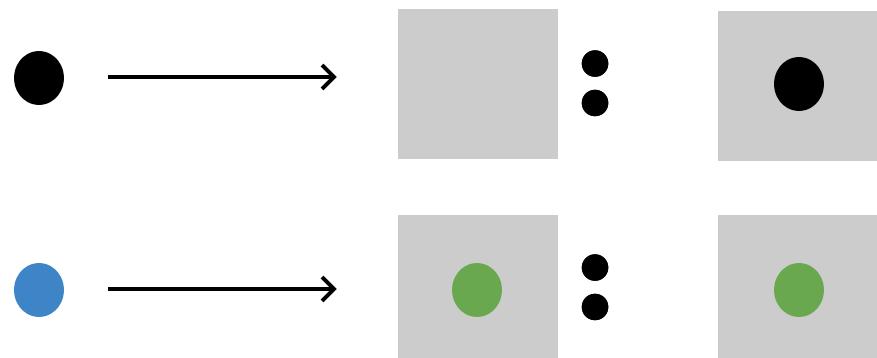
## Evolution

```
1 for generation in I
2     x <- random_individual(X)
3     x' <- random_variation(x)
4     b' <- behavior_function(x')
5     feature_space_insert(x', b')
```

## Termination

- Termination condition is reached
- Return individuals and behaviors

## Feature Space Outcomes



$$F(\text{Blue}) \leq F(\text{Green})$$

# Why not just use MAP-Elites?

# Why not just use MAP-Elites?

# Why not just use MAP-Elites?

It runs the model a lot

# Gaussian Processes

# Gaussian Processes

- Can make extrapolations with little data

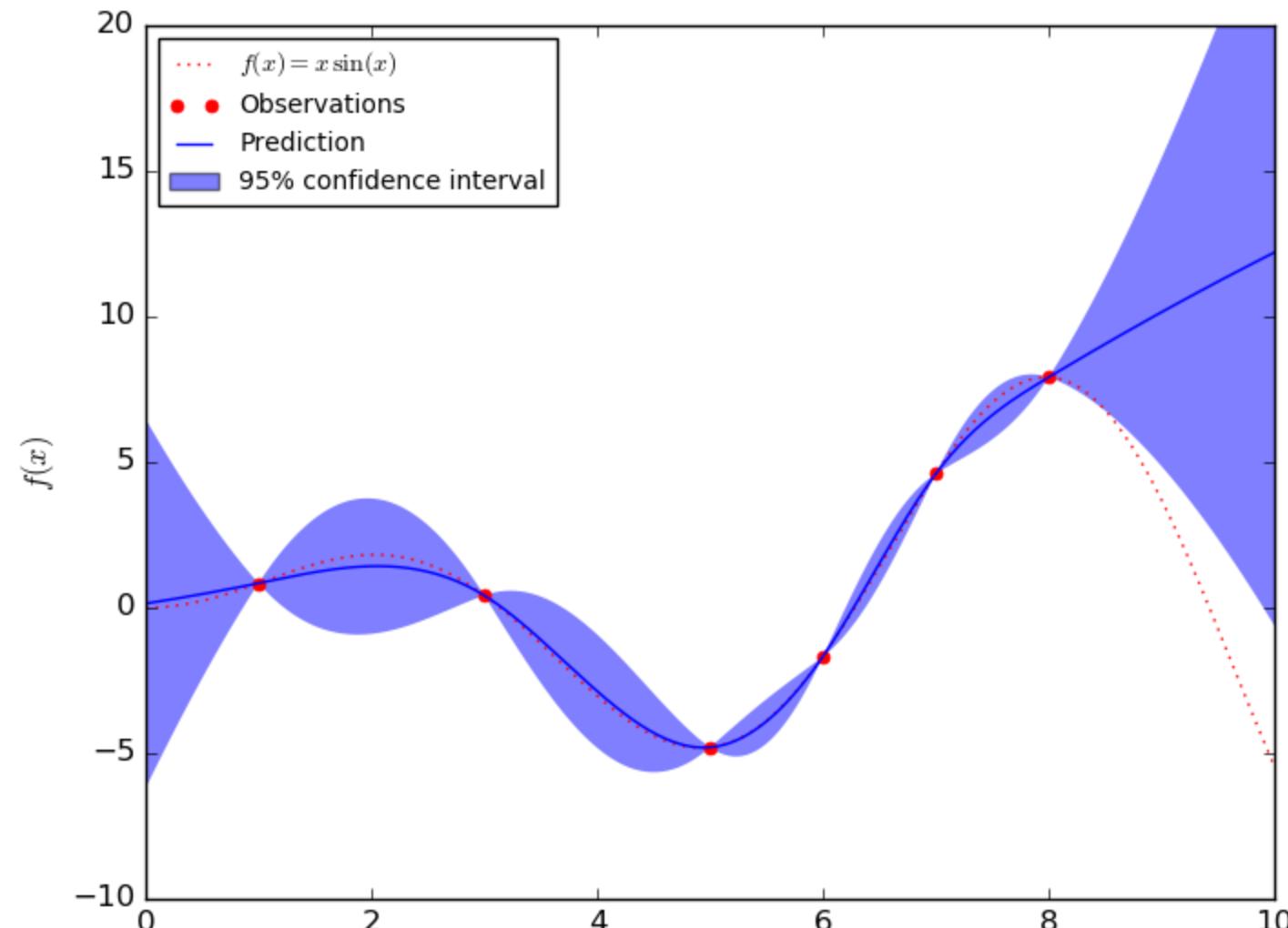
# Gaussian Processes

- Can make extrapolations with little data
- Extrapolating points is computationally cheap

# Gaussian Processes

- Can make extrapolations with little data
- Extrapolating points is computationally cheap
- Includes a confidence with each extrapolation

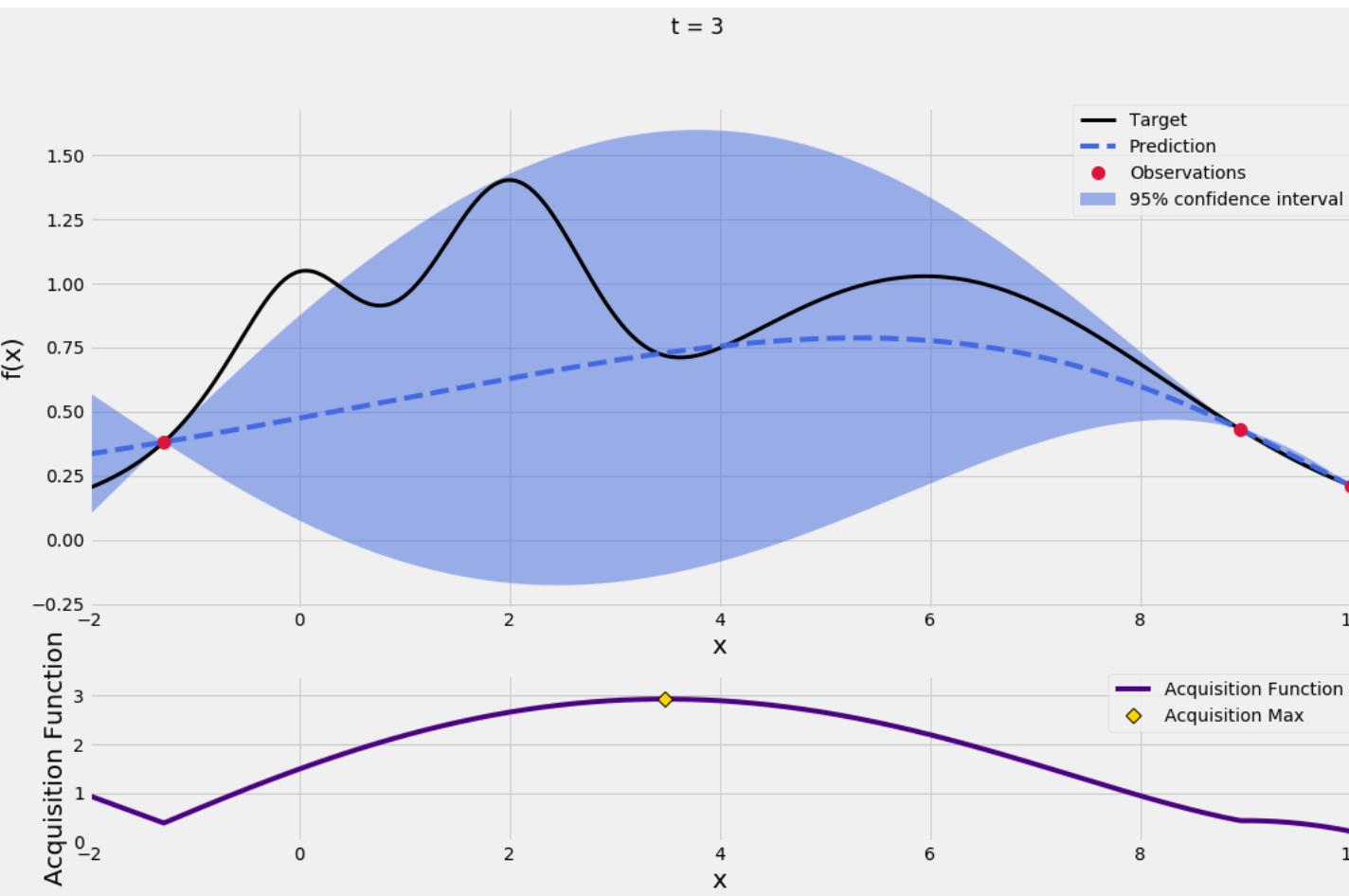
# Gaussian Processes



# Bayesian Optimization

Upper Confidence Bound:

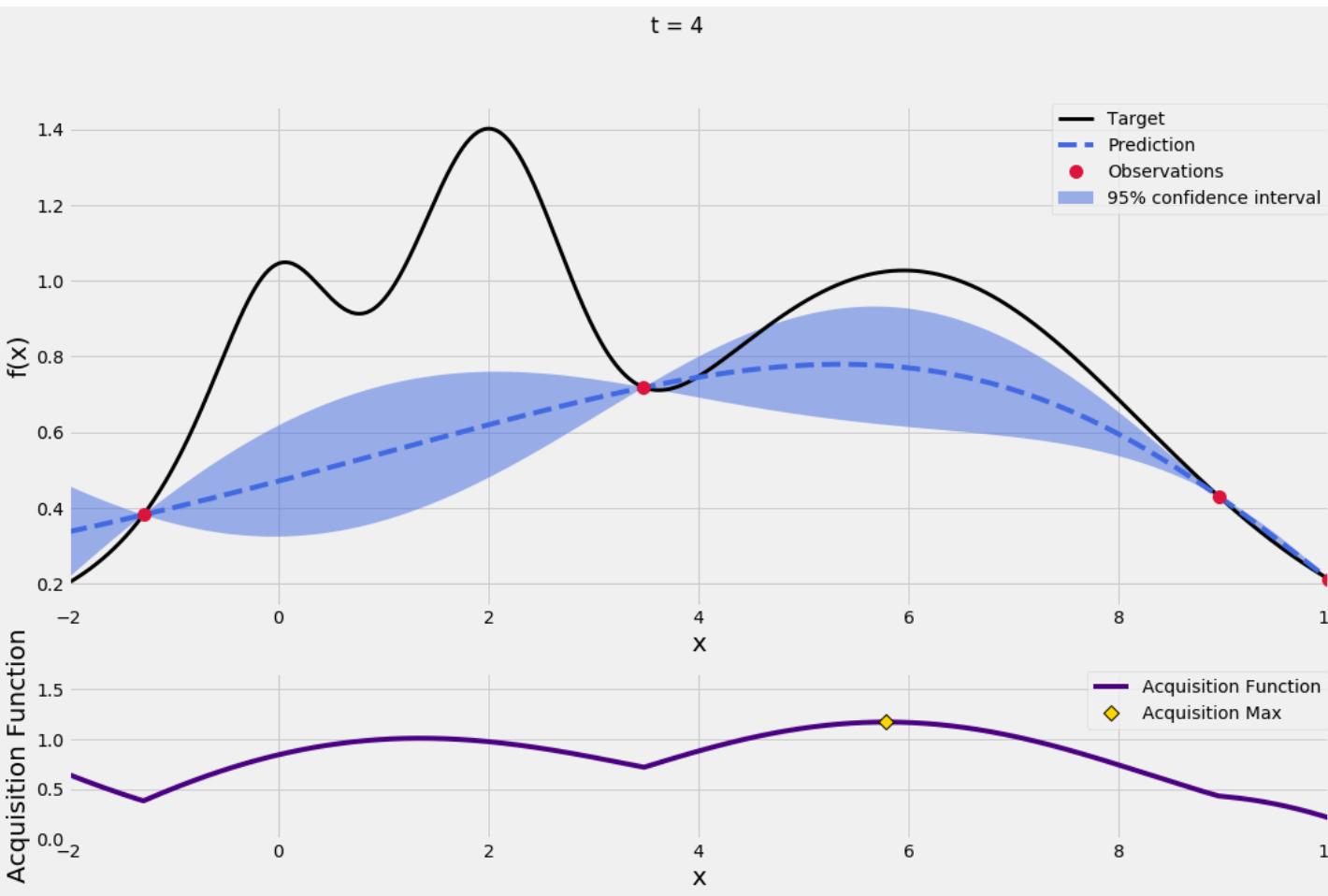
$$UCB(x) = \mu(x) + k\sigma(x)$$



# Bayesian Optimization

Upper Confidence Bound:

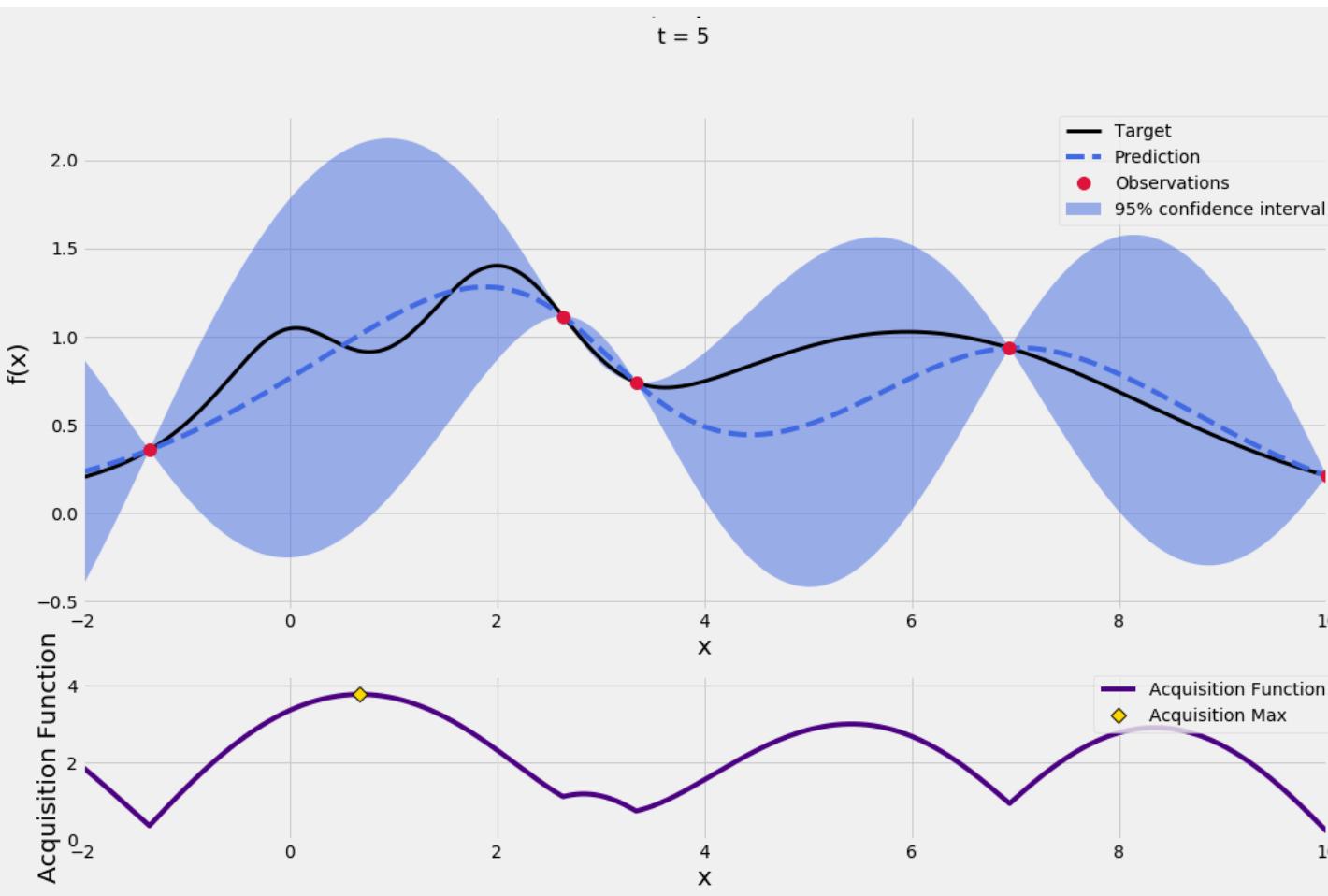
$$UCB(x) = \mu(x) + k\sigma(x)$$



# Bayesian Optimization

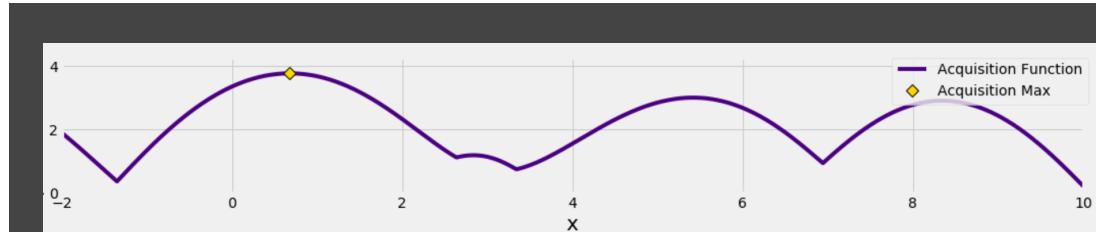
Upper Confidence Bound:

$$UCB(x) = \mu(x) + k\sigma(x)$$



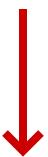
# Model Recap

## Acquisition Function

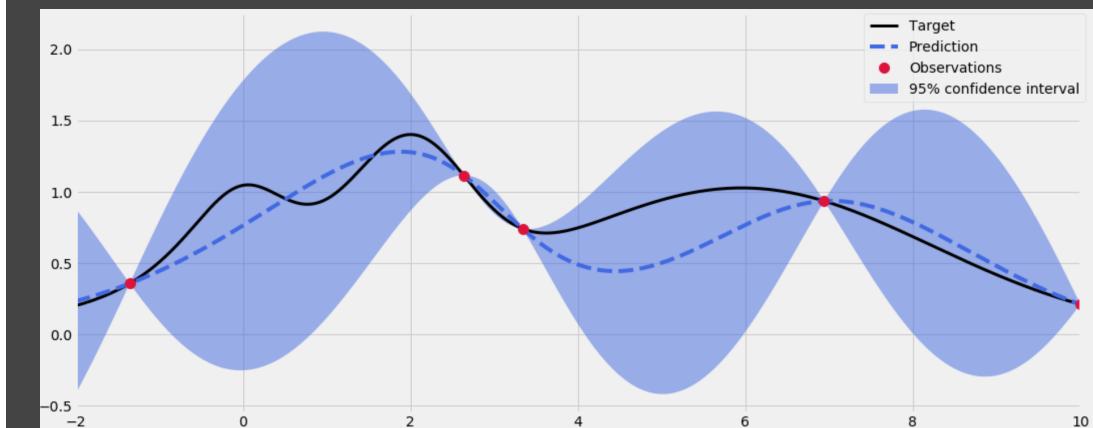
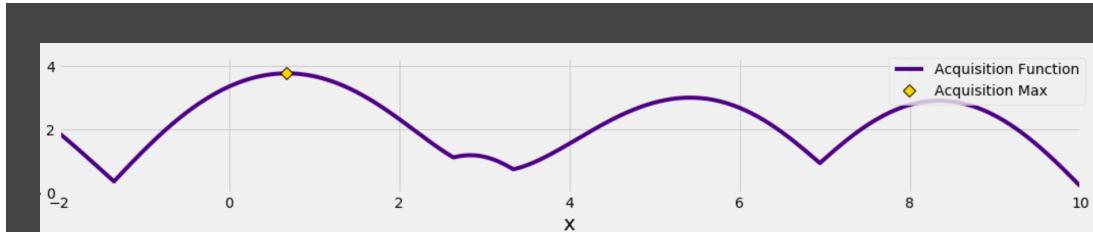


# Model Recap

## Acquisition Function

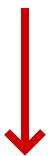


## Gaussian Process



# Model Recap

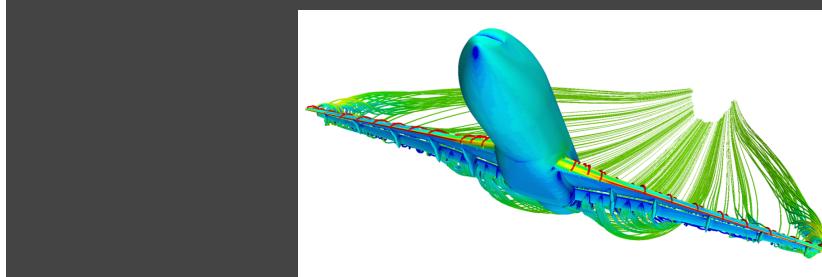
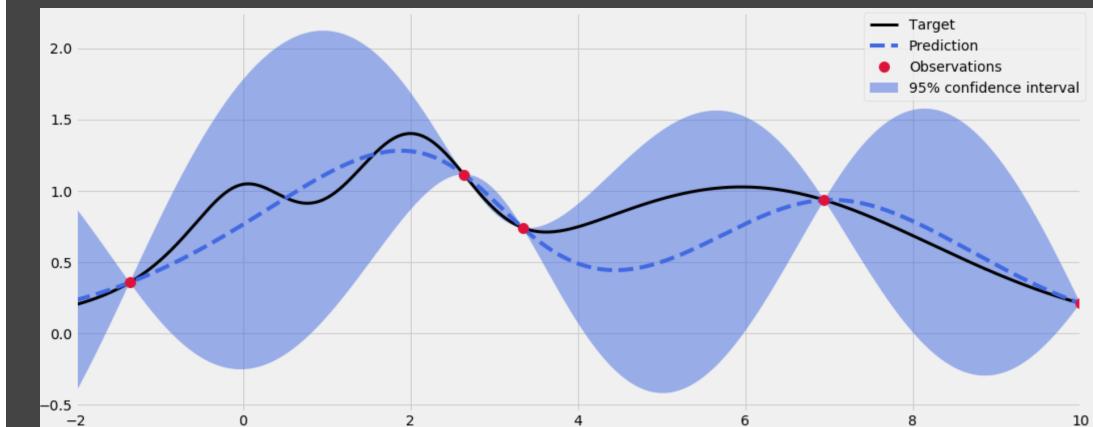
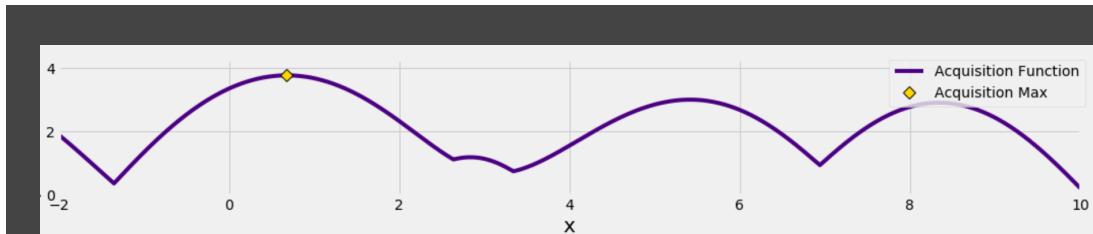
## Acquisition Function



## Gaussian Process



## CFD



# Model Recap

Acquisition  
Function



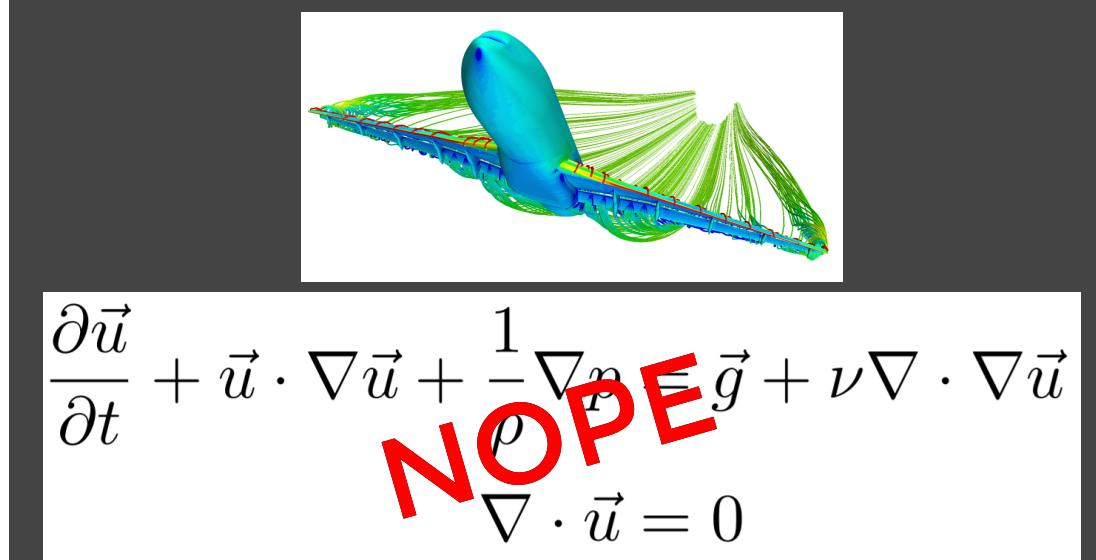
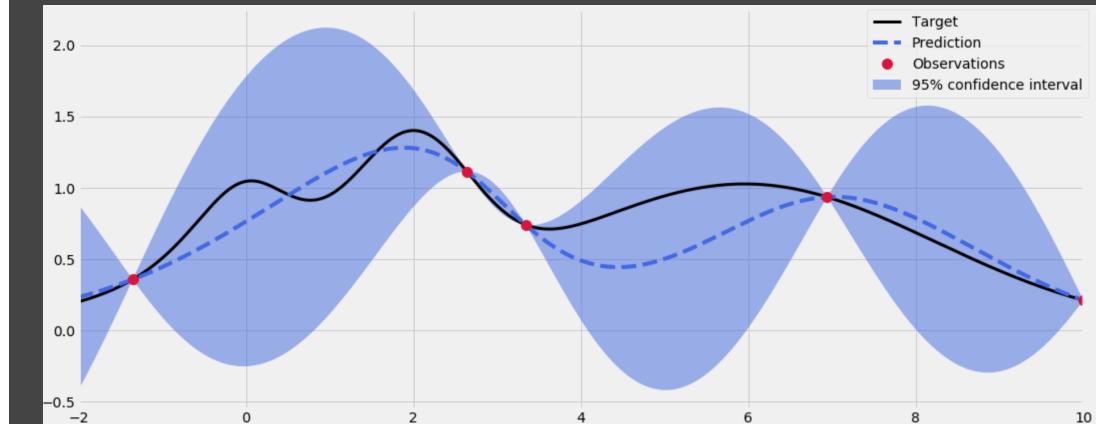
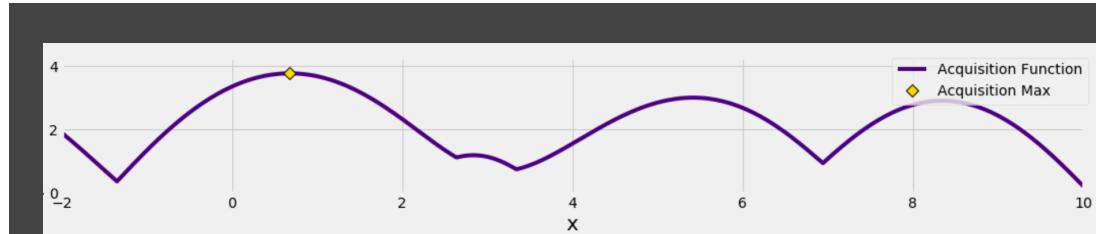
Gaussian Process



CFD



Naiver Stokes



# SAIL Algorithm

**Setup:** Create Gaussian  
Process

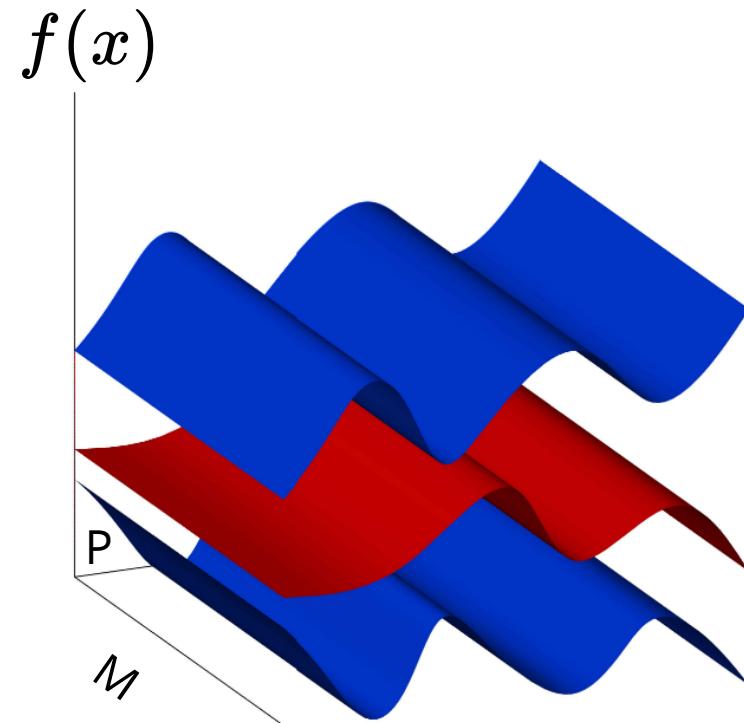
**Evolution:** Produce  
Acquisition Map

**Termination:** Produce  
Prediction Map

# SAIL Algorithm

## Setup: Create Gaussian Process

- Select random individuals
- Create Gaussian process



# SAIL Algorithm

## Evolution: Produce Acquisition Map

Within computational budget:

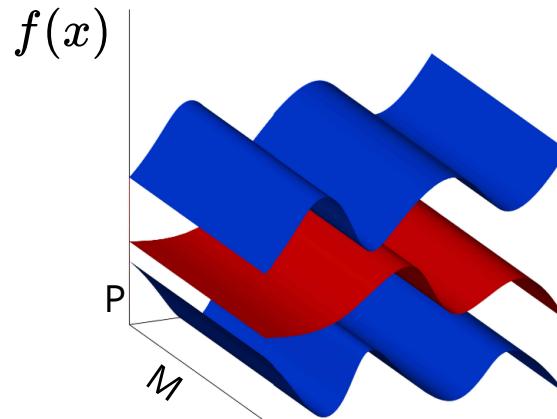
- Create acquisition function
- Illuminate with MAP-Elites

# SAIL Algorithm

## Evolution: Produce Acquisition Map

Within computational budget:

- Create acquisition function
- Illuminate with MAP-Elites



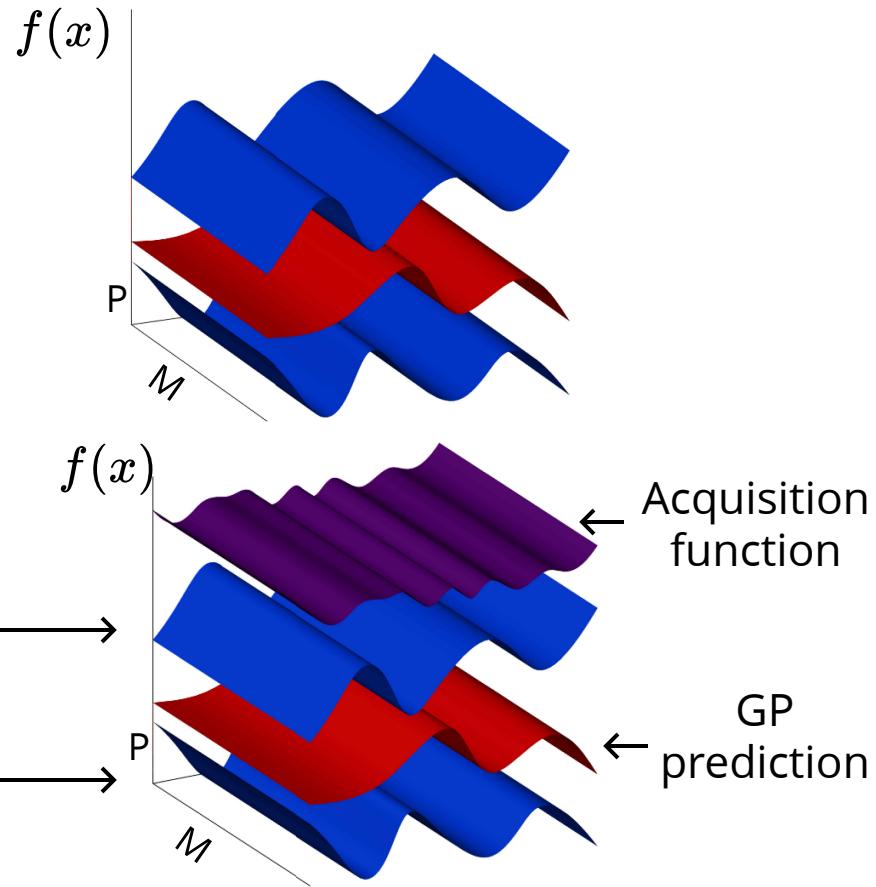
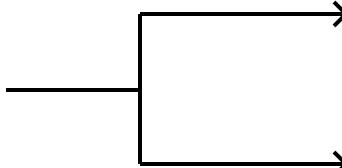
# SAIL Algorithm

## Evolution: Produce Acquisition Map

Within computational budget:

- Create acquisition function
- Illuminate with MAP-Elites

GP  
confidence



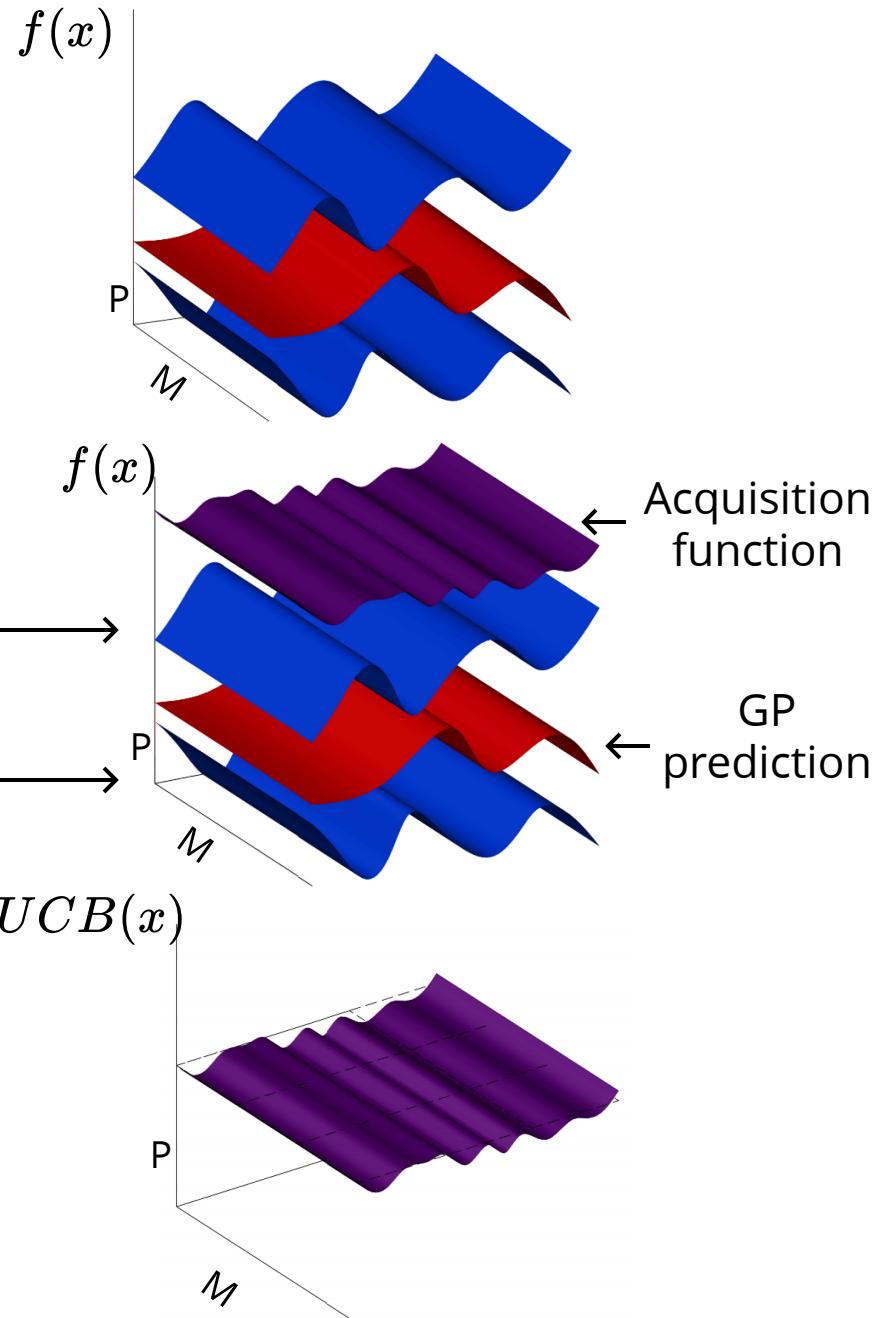
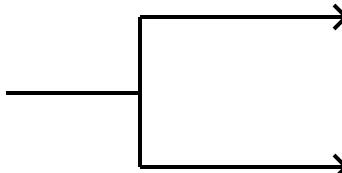
# SAIL Algorithm

## Evolution: Produce Acquisition Map

Within computational budget:

- Create acquisition function
- Illuminate with MAP-Elites

GP  
confidence



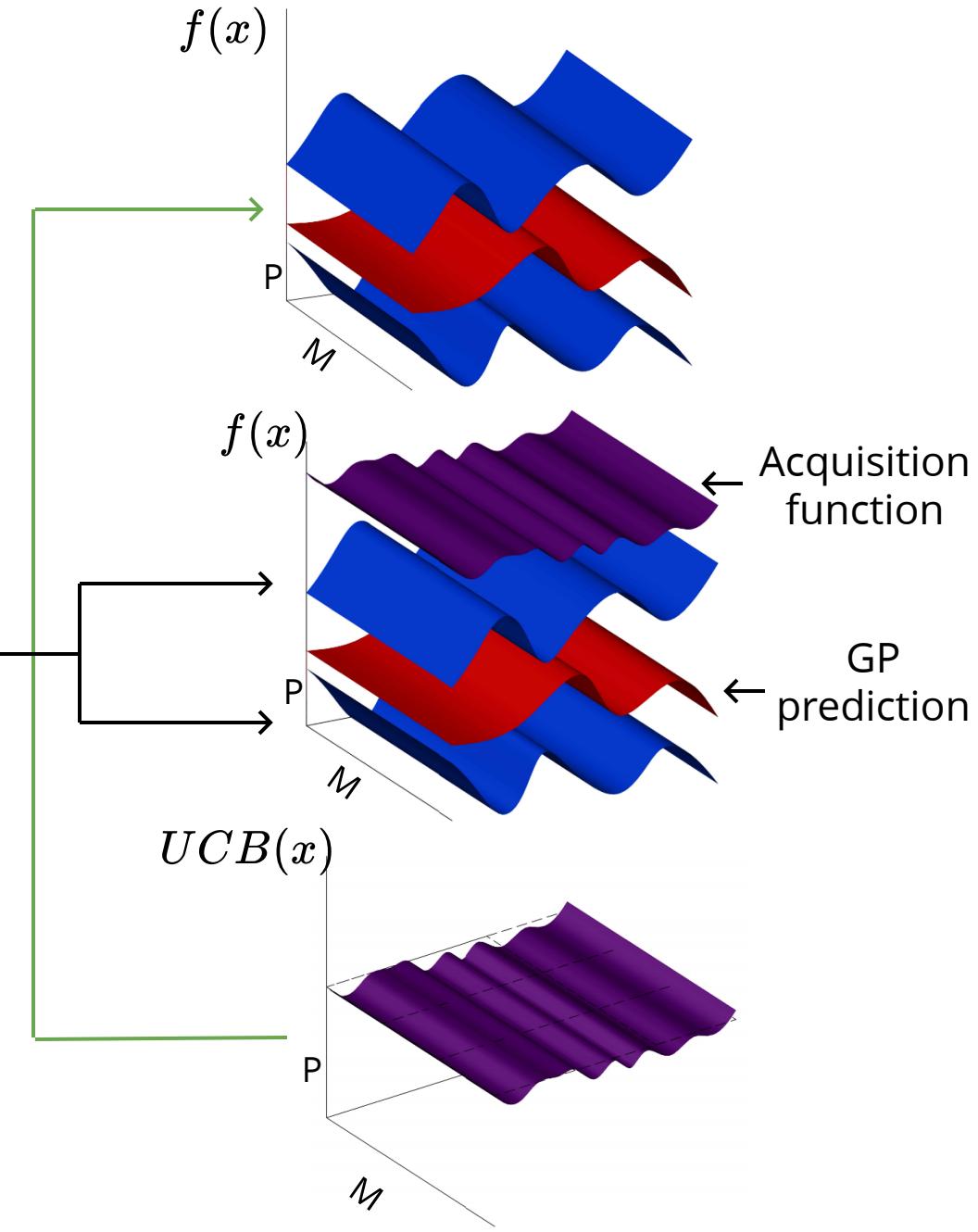
# SAIL Algorithm

## Evolution: Produce Acquisition Map

Within computational budget:

- Create acquisition function
- Illuminate with MAP-Elites

GP  
confidence



# SAIL Algorithm

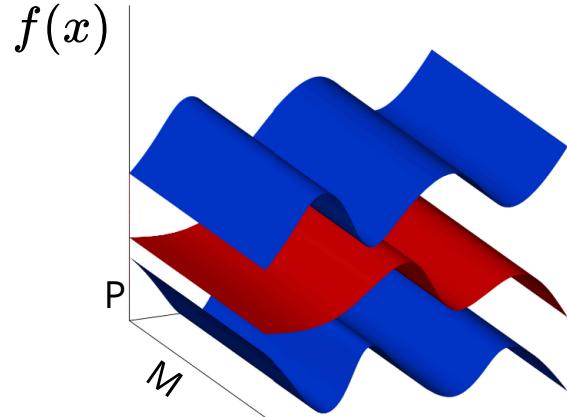
## Termination: Produce Prediction Map

- Grab only the GP-Prediction
- Illuminate with MAP Elites

# SAIL Algorithm

## Termination: Produce Prediction Map

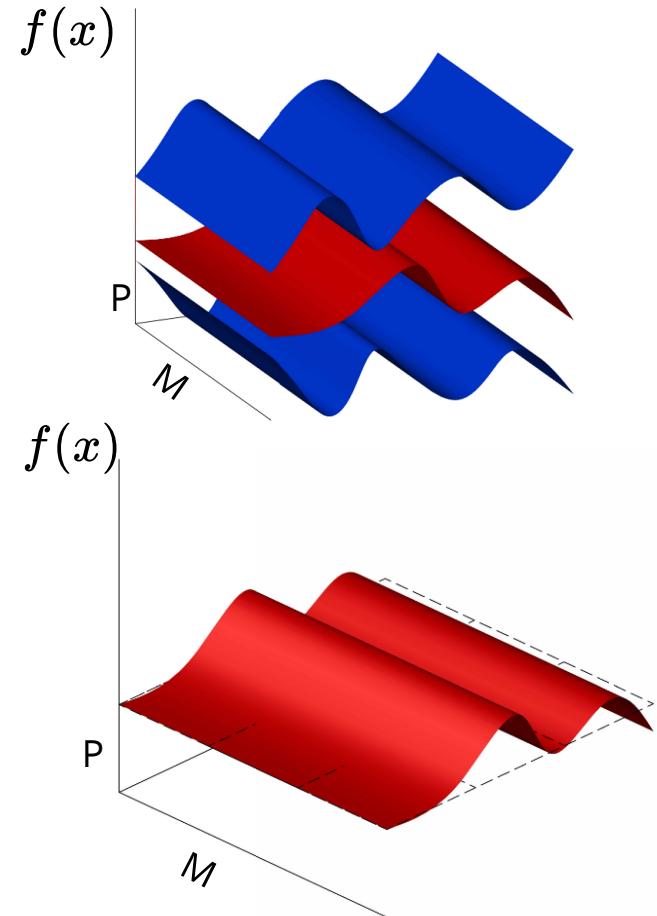
- Grab only the GP-Prediction
- Illuminate with MAP Elites



# SAIL Algorithm

## Termination: Produce Prediction Map

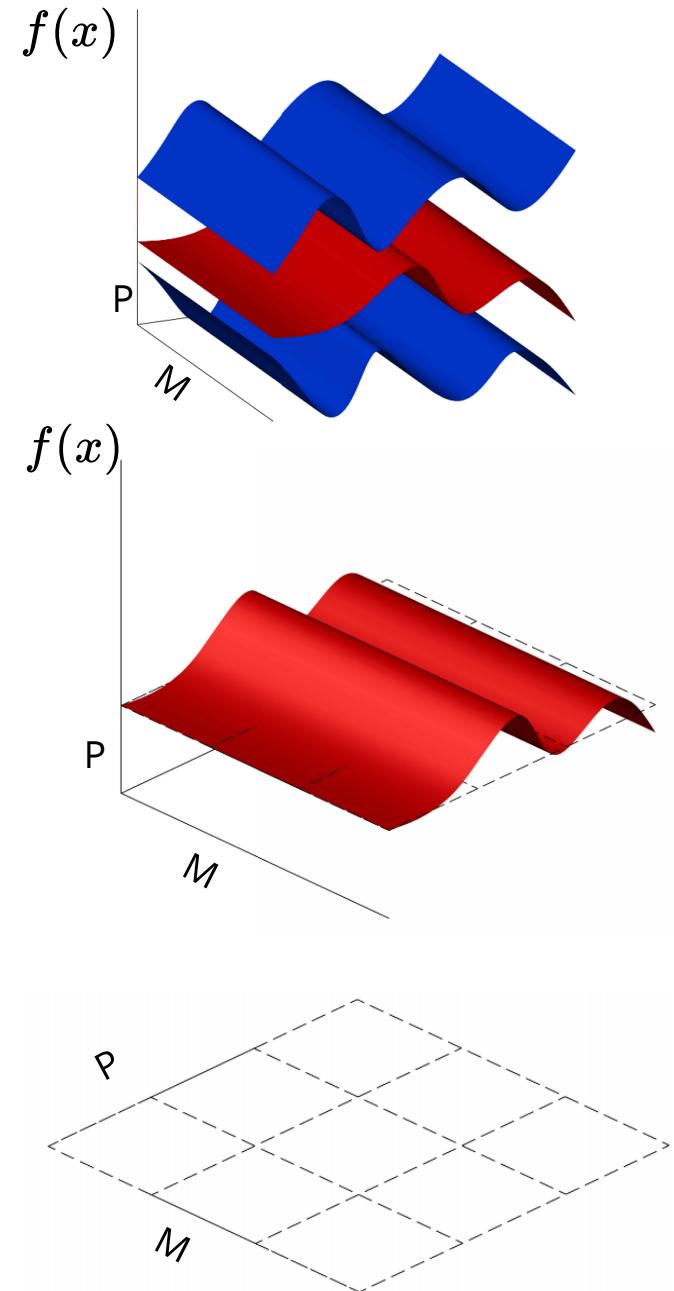
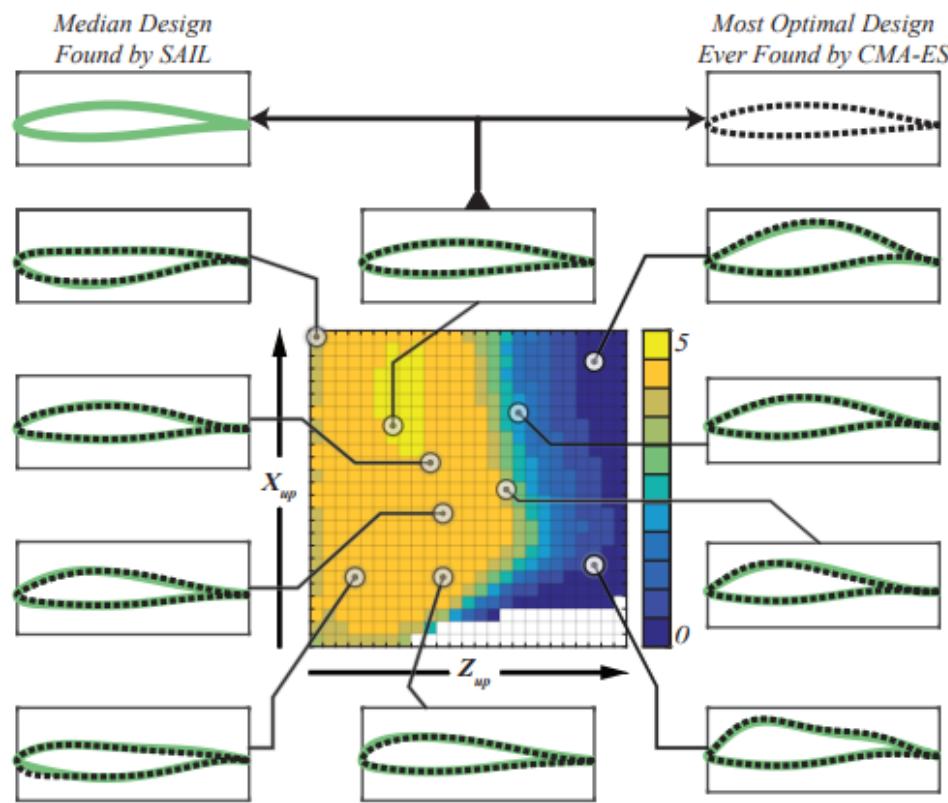
- Grab only the GP-Prediction
- Illuminate with MAP Elites



# SAIL Algorithm

## Termination: Produce Prediction Map

- Grab only the GP-Prediction
- Illuminate with MAP Elites



# Velomobile Experiment



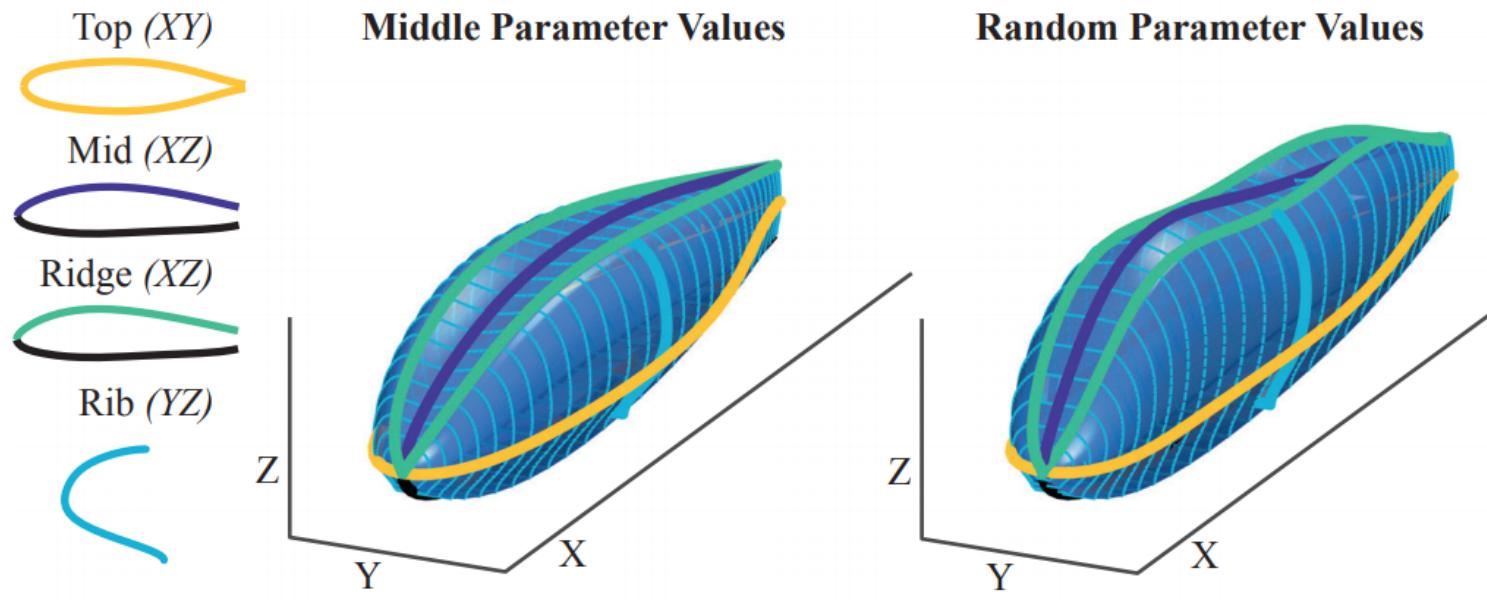
Taken from [1]

## Challenges

- CFD Simulations for drag calculation are expensive
- How do you represent a velomobile shape?
- How do you run a traditional optimizer here?

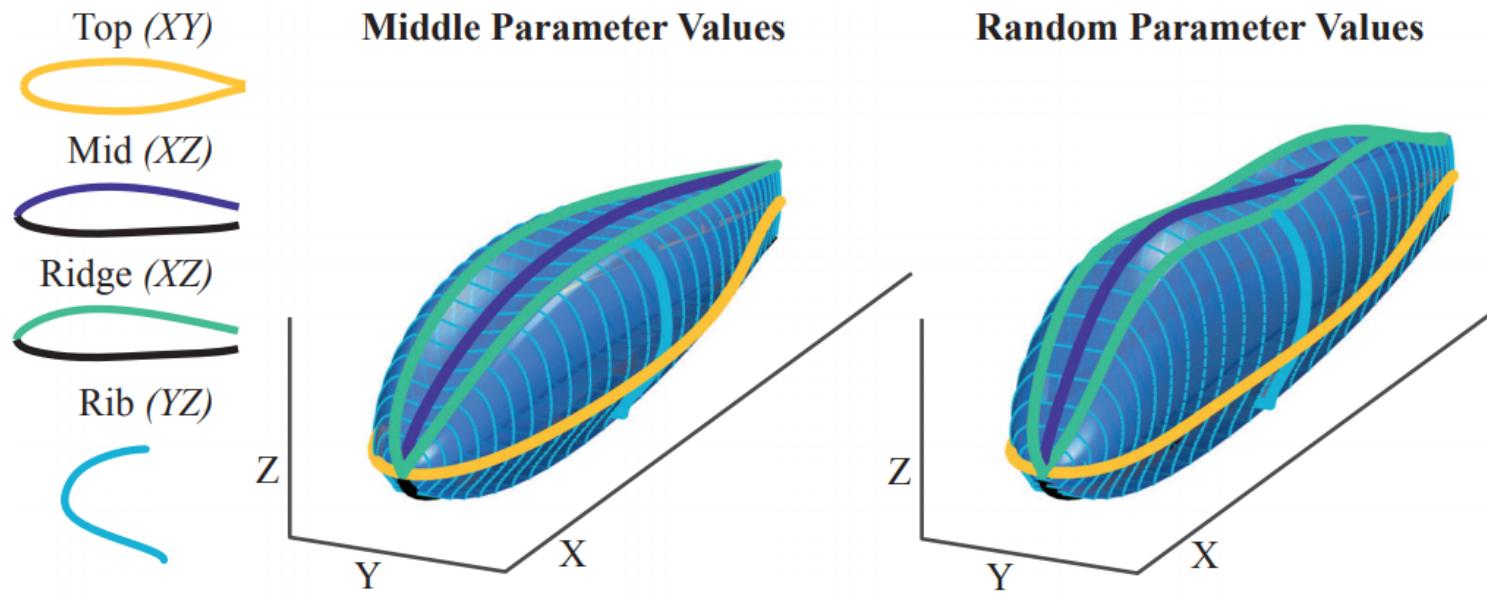
# Velomobile Experiment Representations

## Parameterization



# Velomobile Experiment Representations

## Parameterization

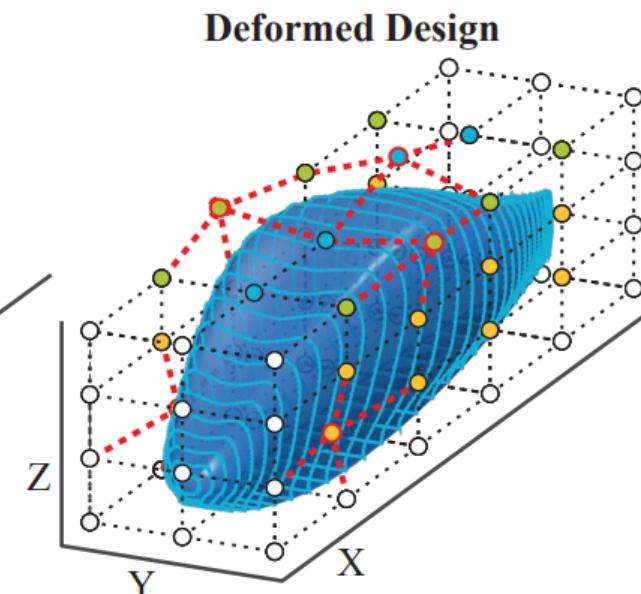
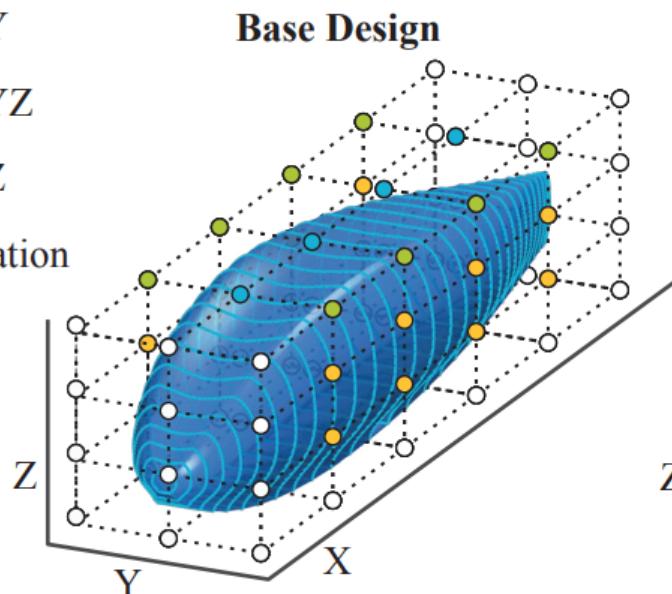


16 parameters

# Velomobile Experiment Representations

## Deformation

- Deform in Y
- Deform in YZ
- Deform in Z
- No Deformation

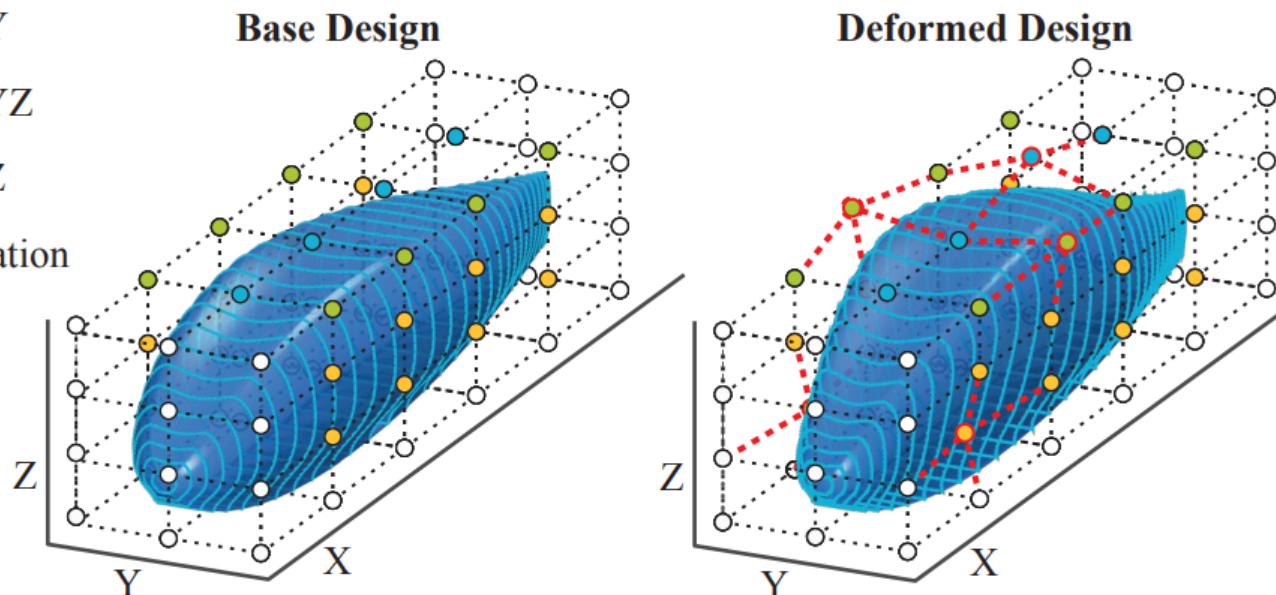


Taken from [1]

# Velomobile Experiment Representations

## Deformation

- Deform in Y
- Deform in YZ
- Deform in Z
- No Deformation



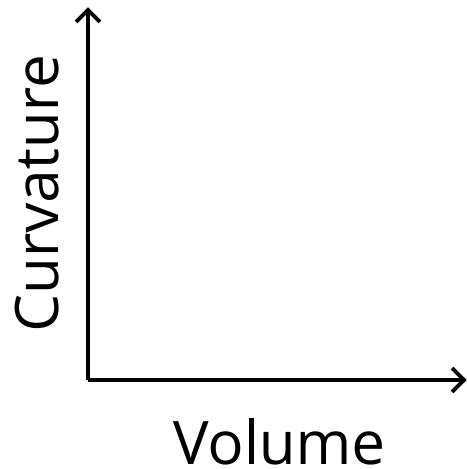
16 parameters

Taken from [1]

# **Velomobile Experiment**

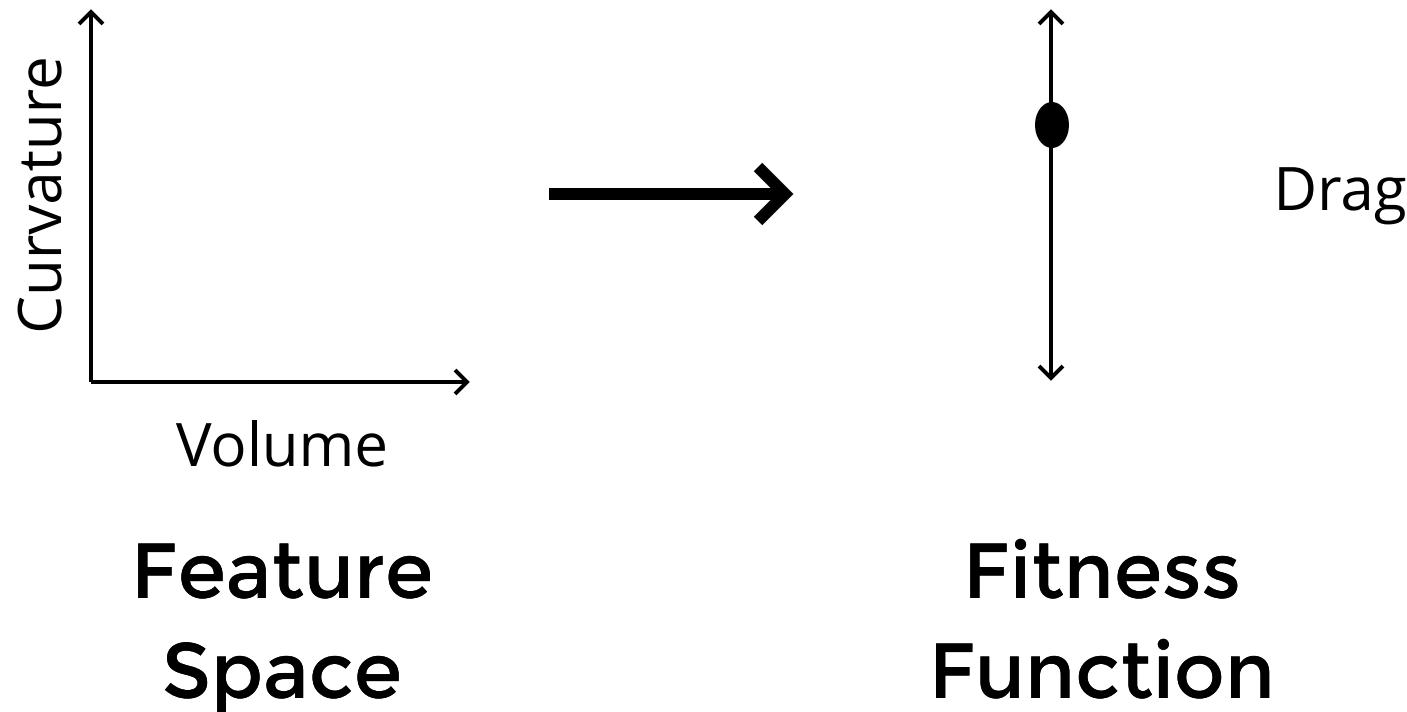
## **Setup**

# Velomobile Experiment Setup

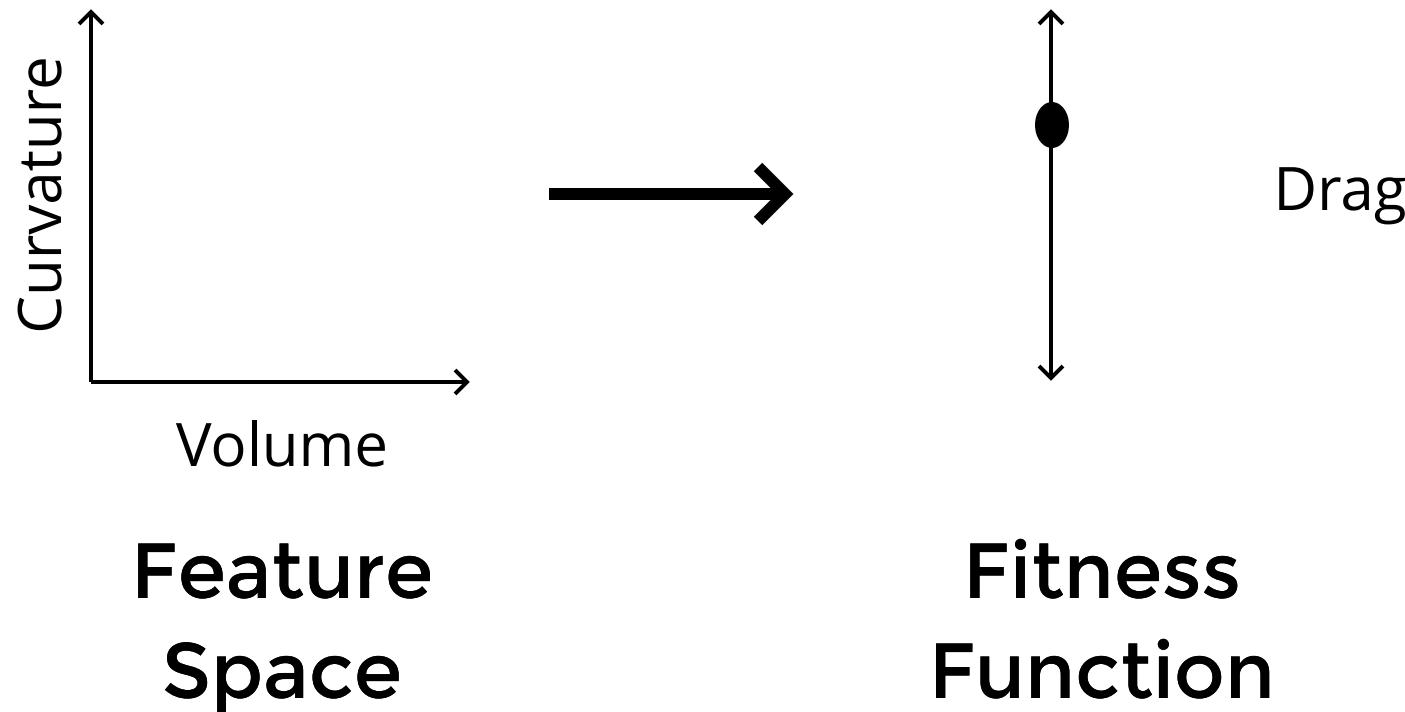


**Feature  
Space**

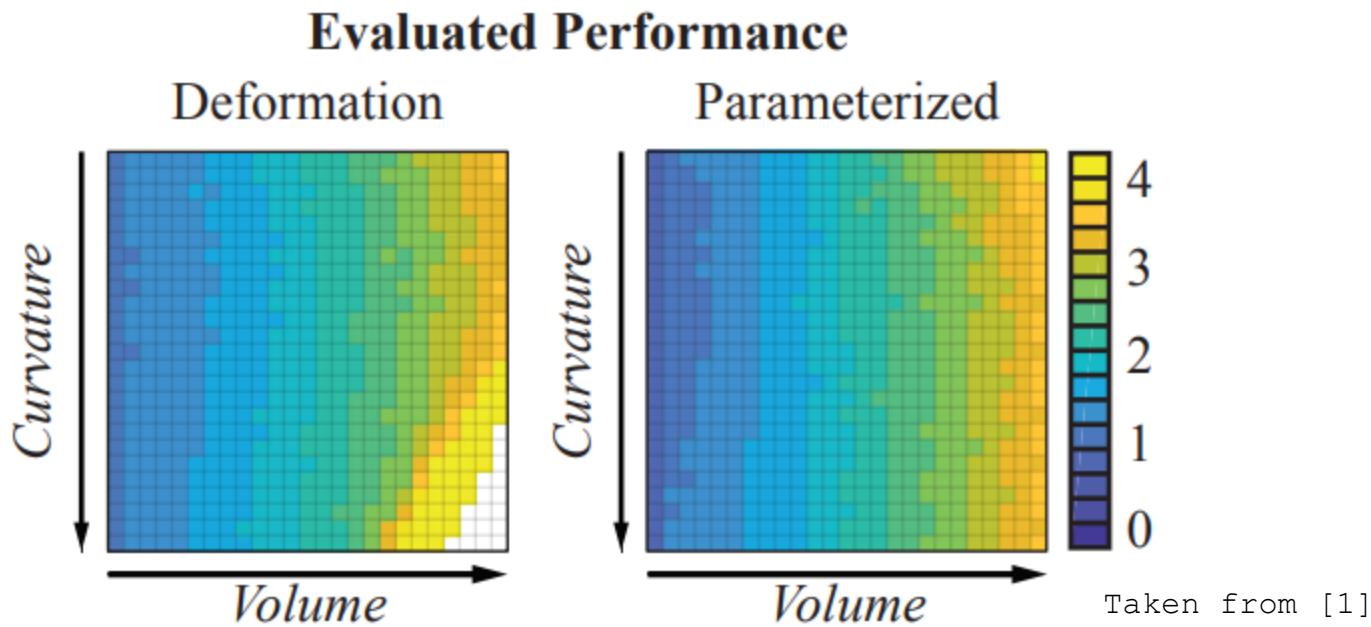
# Velomobile Experiment Setup



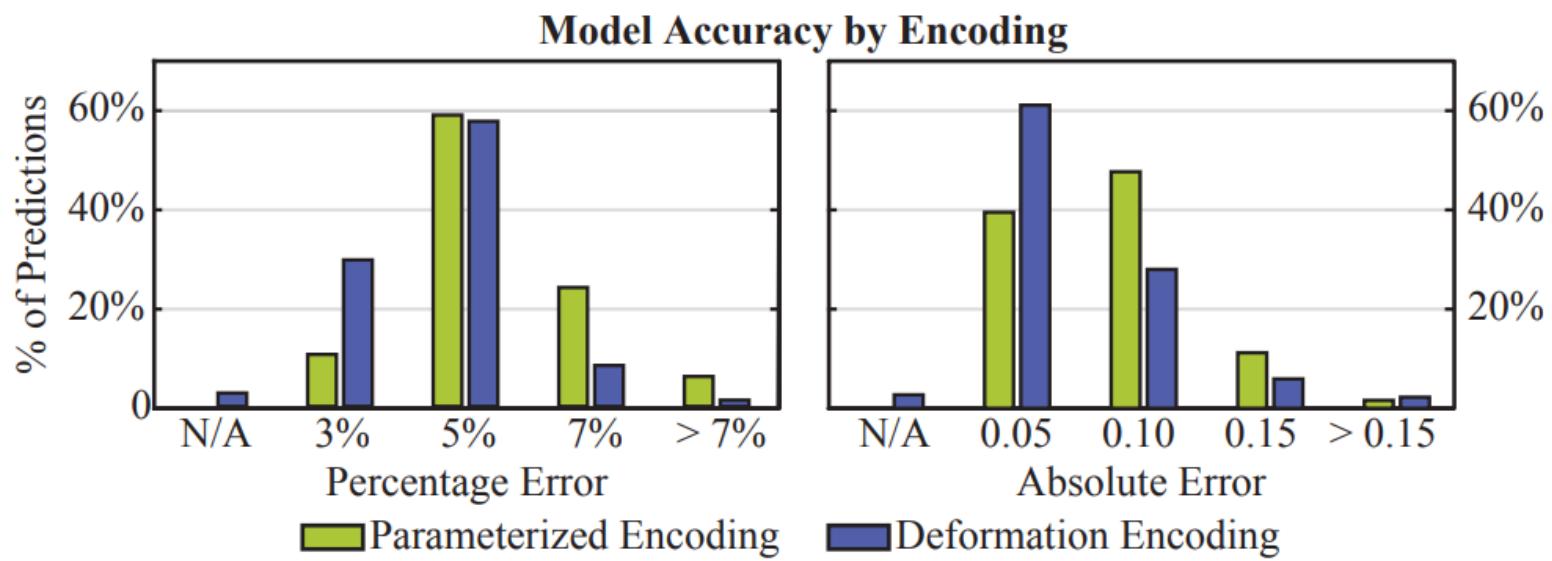
# Velomobile Experiment Setup



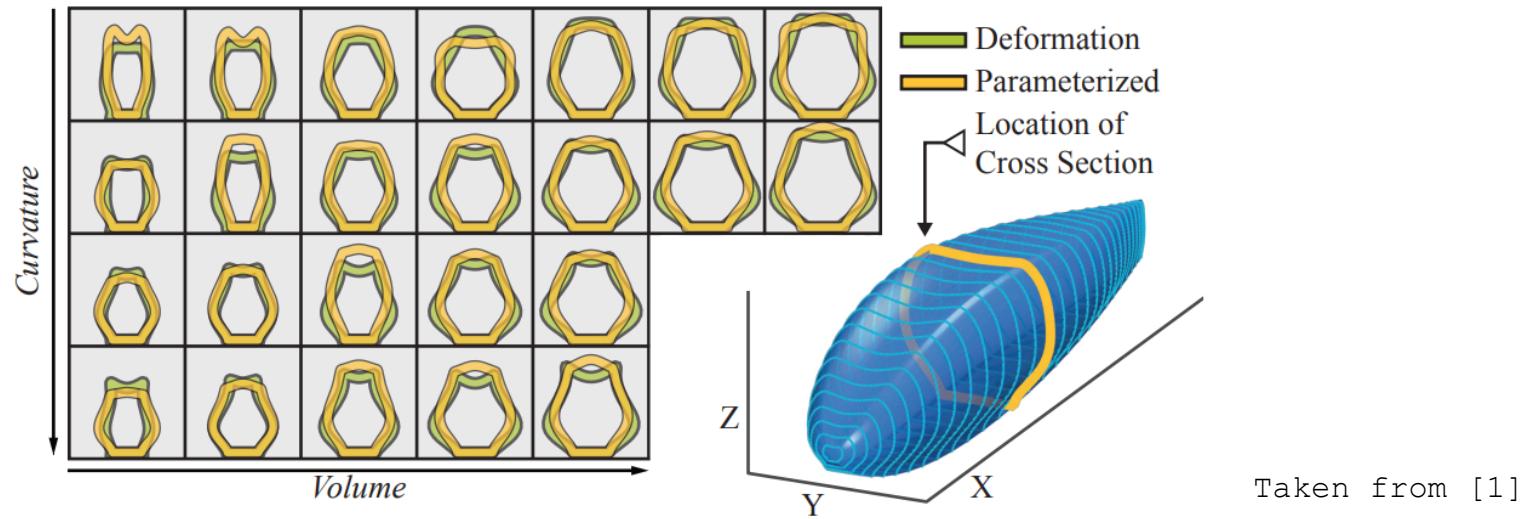
# Velomobile Experiment Results



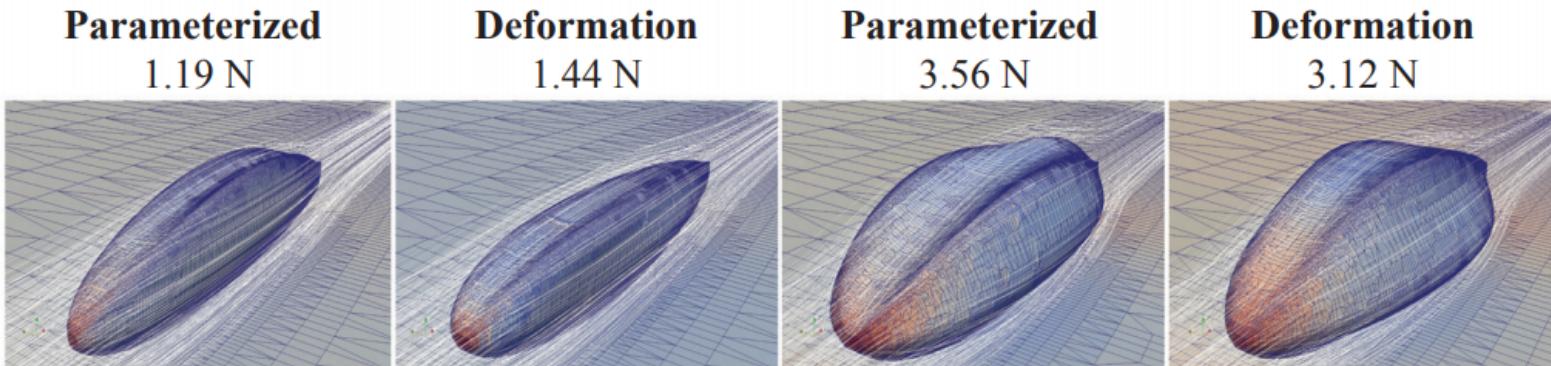
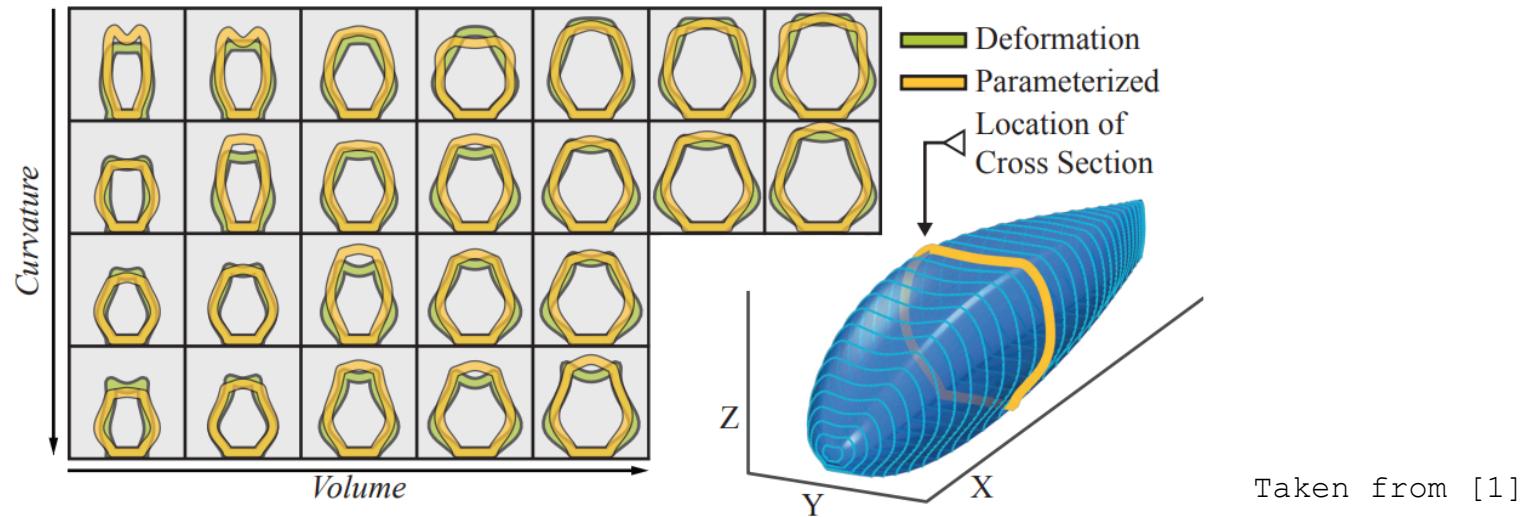
# Velomobile Experiment Results



# Velomobile Experiment Results



# Velomobile Experiment Results



# Acknowledgments

# Acknowledgments

- Andrew Kroska and Shawn Saliyev

# Acknowledgments

- Andrew Kroska and Shawn Saliyev
- Lexi Barta

# Acknowledgments

- Andrew Kroska and Shawn Saliyev
- Lexi Barta
- Nic McPhee

# Acknowledgments

- Andrew Kroska and Shawn Saliyev
- Lexi Barta
- Nic McPhee
- Computer Science faculty

# Acknowledgments

- Andrew Kroska and Shawn Saliyev
- Lexi Barta
- Nic McPhee
- Computer Science faculty
- All of you!

# Acknowledgments

- Andrew Kroska and Shawn Saliyev
- Lexi Barta
- Nic McPhee
- Computer Science faculty
- All of you!

# Citations

[1] Adam Gaier, Alexander Asteroth, Jean-Baptiste Mouret.  
Data-Efficient Design Exploration through Surrogate-Assisted  
Illumination

# Acknowledgments

- Andrew Kroska and Shawn Saliyev
- Lexi Barta
- Nic McPhee
- Computer Science faculty
- All of you!

# Citations

[1] Adam Gaier, Alexander Asteroth, Jean-Baptiste Mouret.  
Data-Efficient Design Exploration through Surrogate-Assisted  
Illumination

# Questions?

Get in touch!

[scot0530@morris.umn.edu](mailto:scot0530@morris.umn.edu)

[github.com/morrislenny](https://github.com/morrislenny)

[linkedin.com/in/leonid-scott](https://www.linkedin.com/in/leonid-scott)