Computational Assemblies: Analysis, Design, and Fabrication

Peng Song
Ziqi Wang
Marco Livesu

SINGAPORE UNIVERSITY OF TECHNOLOGY AND DESIGN
ETH zürich
imati

Timetable

		Peng	Ziqi	Marco
Introduction	~20 mins	X		
Computational analysis of assemblies	~ 50 mins	X		
Computational design of assemblies	~50 mins		X	
Computational fabrication of assemblies	~ 50 mins			X
Q & A	~ 10 mins	X	X	X

Computational Design of Assemblies

- Our goal is to design assemblies to achieve users' required objectives with the help of computational methods.

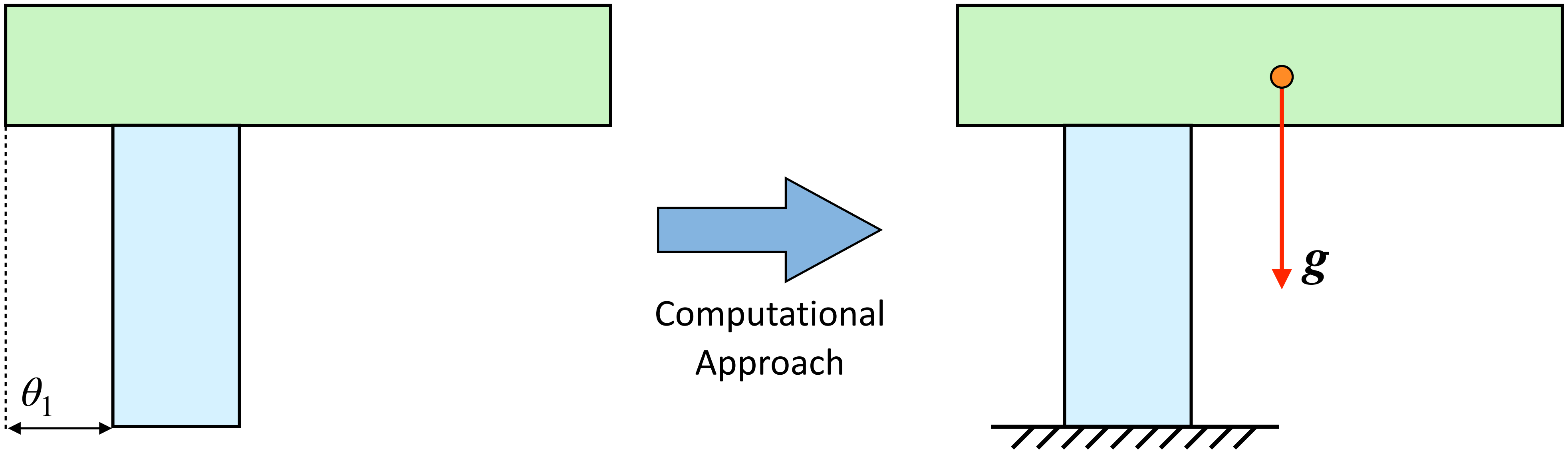
Puzzle

Furniture

Architecture

Case Study

- **Geometry**: The assembly's geometry is determined by several design parameters.
- **Objectives**: The goal is to make the assembly equilibrium under gravity.

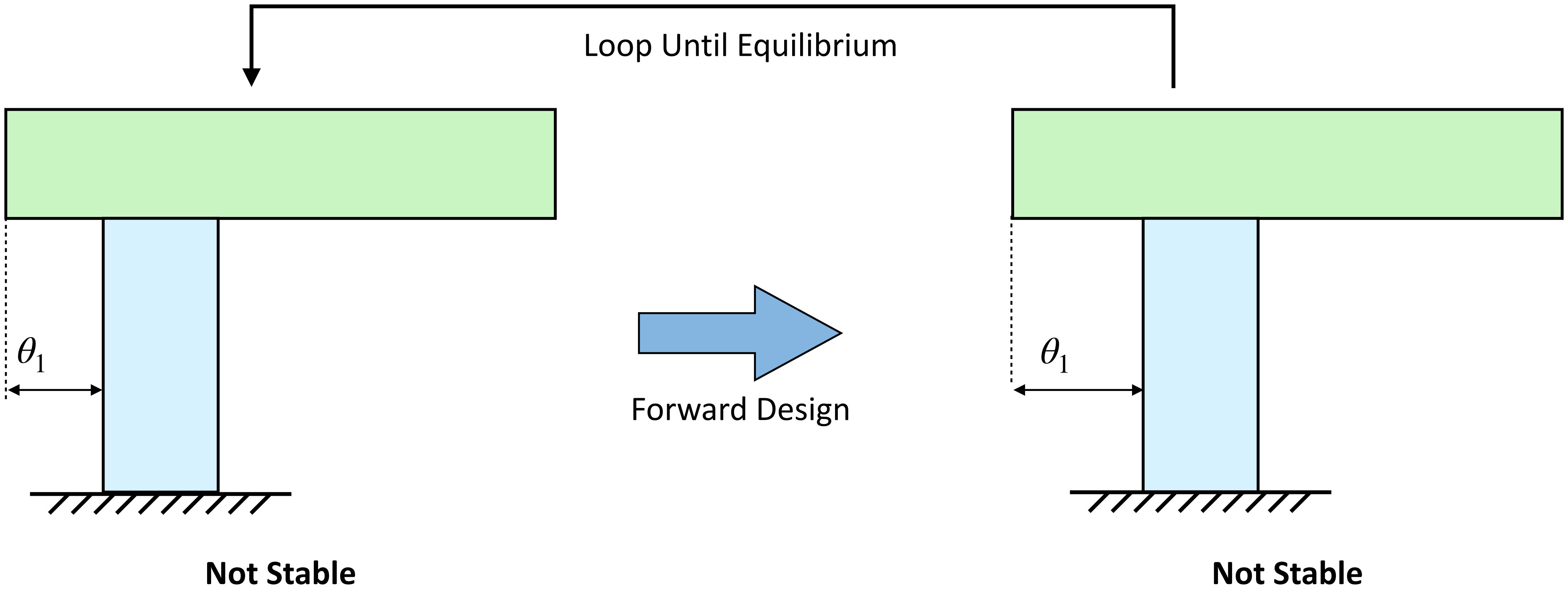


Geometry
Parametric Model (t)

Objectives
Equilibrium under Gravity

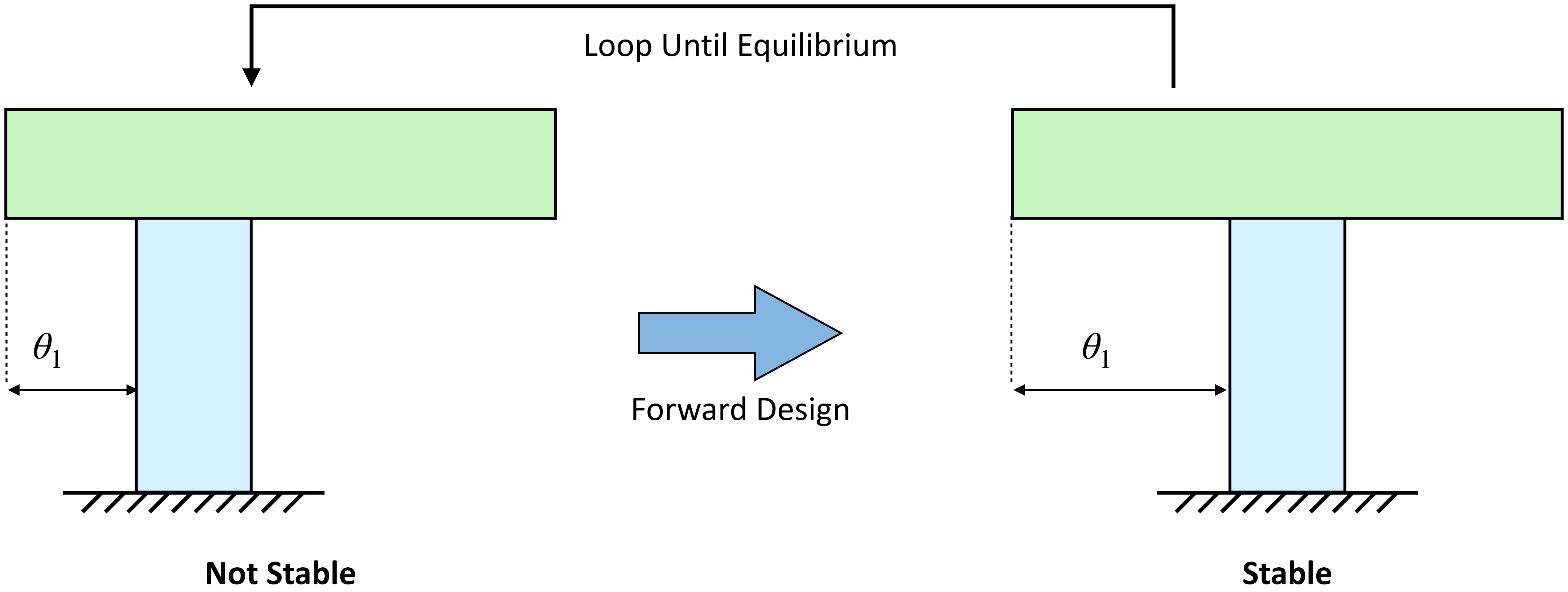
Forward Design Framework

- Manually tune the design parameters until the assembly can stay in equilibrium under gravity.
- More design iterations are required if the current design is not stable.



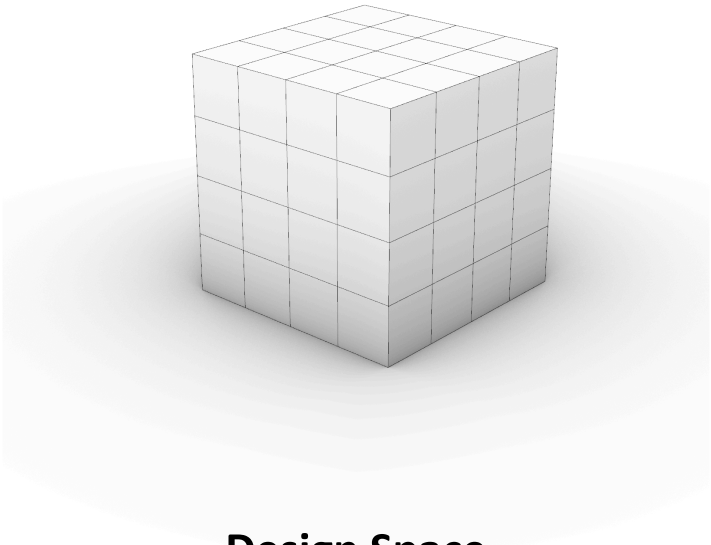
Forward Design Framework

- The forward design framework is challenging and demands expertises.



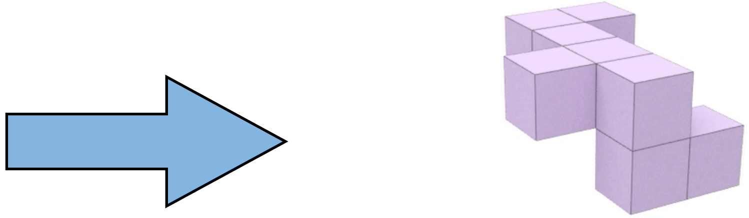
Challenges of Forward Design

- The problem can have an enormous design space but finding one feasible solution is non-trivial.



Design Space

6^{64}

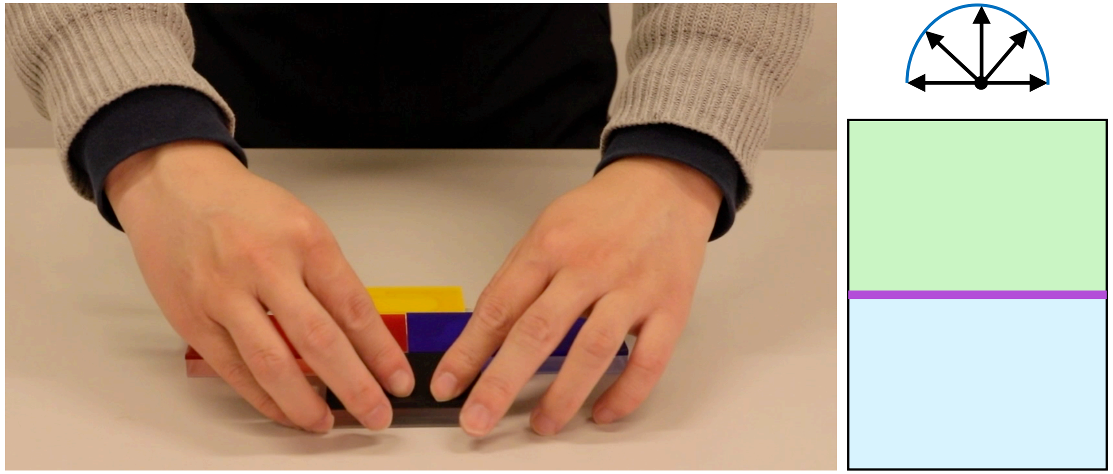


Objectives

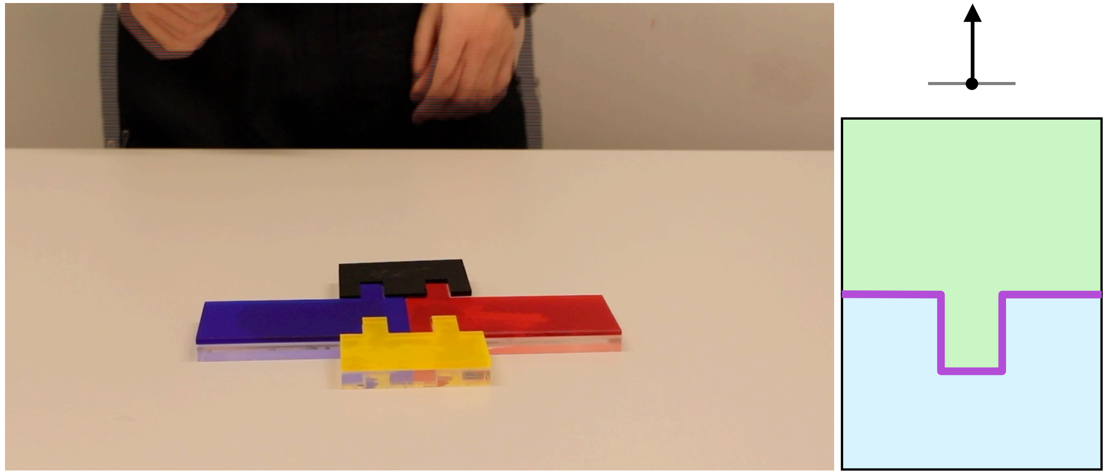
Stable under arbitrary forces

Challenges of Forward Design

- The problem can have multiple design objectives.
- They might be contradicting.



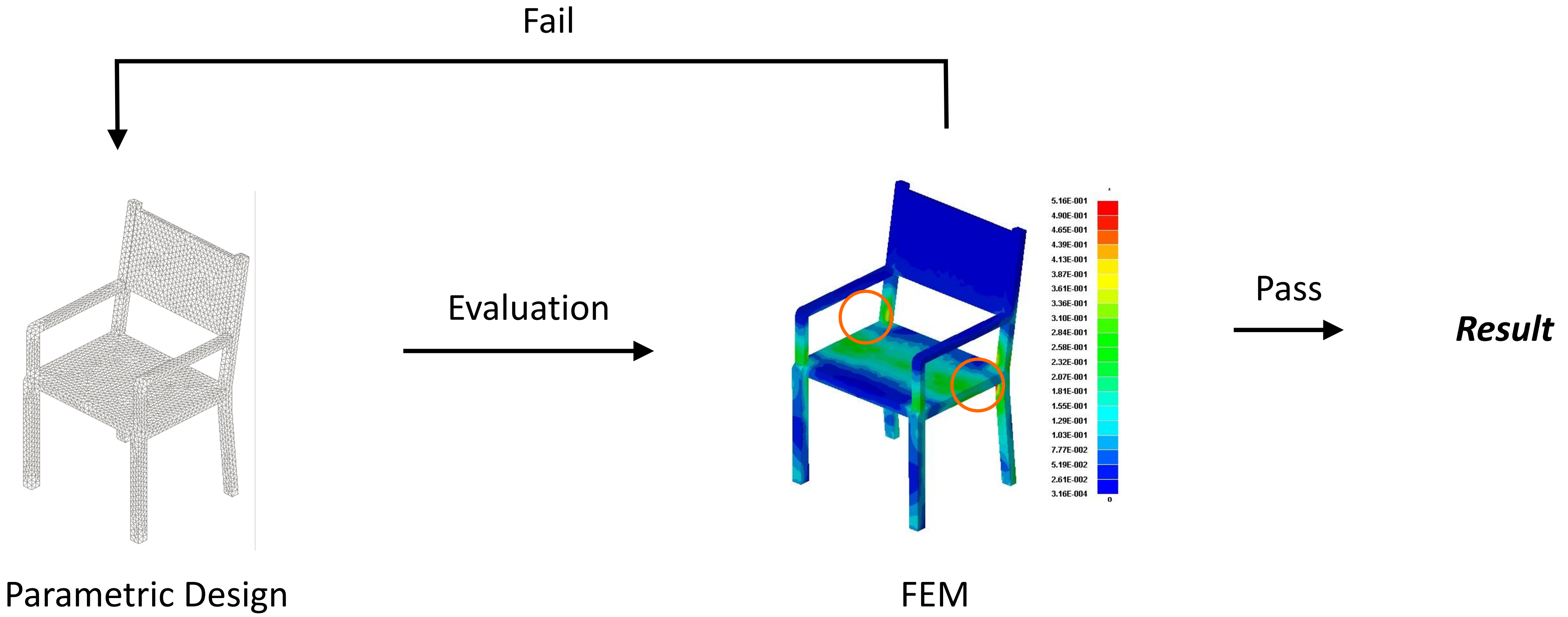
Easy-to-Assemble
Non-stable



Hard-to-Assemble
Stable

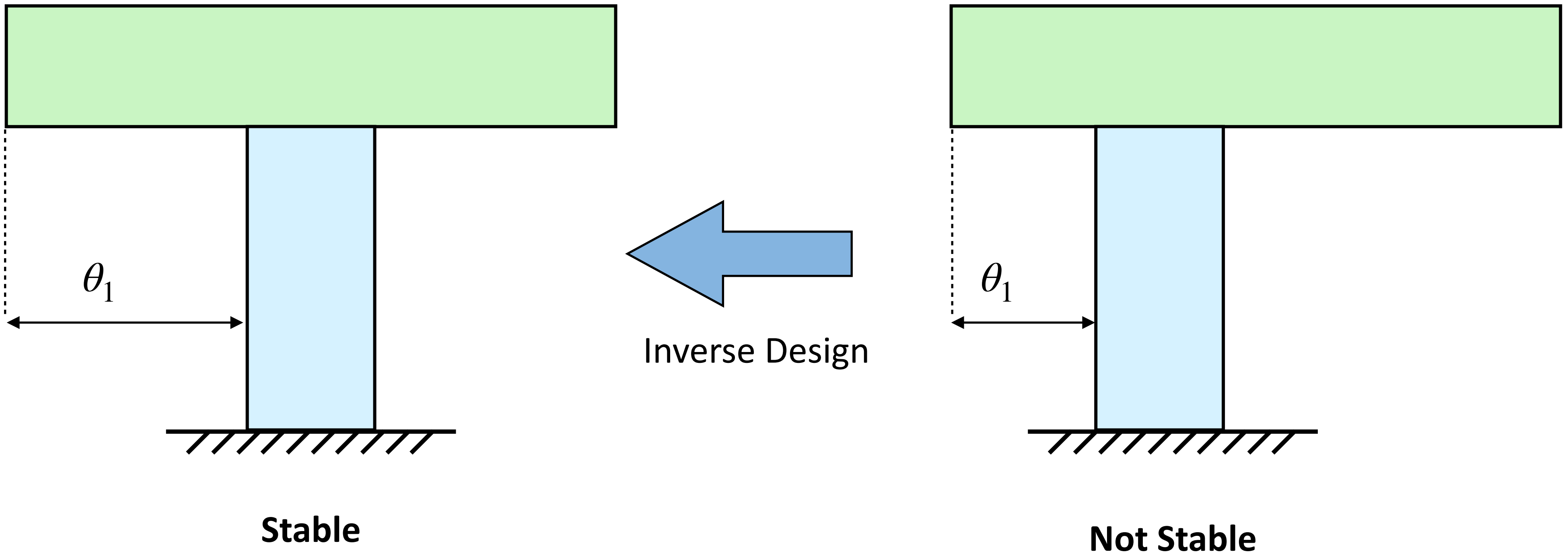
Challenges of Forward Design

- Some evaluation processes are not time-efficient.



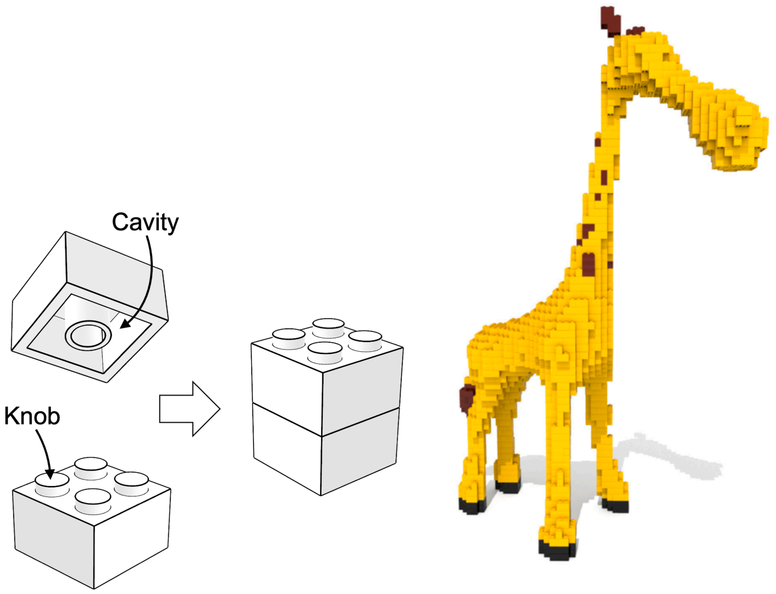
Inverse Design Framework

- Design algorithms to generate assemblies that satisfy users' specified objectives.



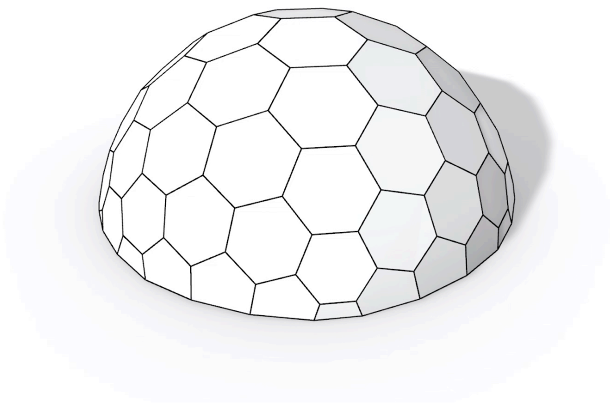
Part Geometry

- Discrete Geometry: searching algorithm.
- Continuous Geometry: gradient-based algorithm.



Discrete Geometry

[Luo et al. 2015]

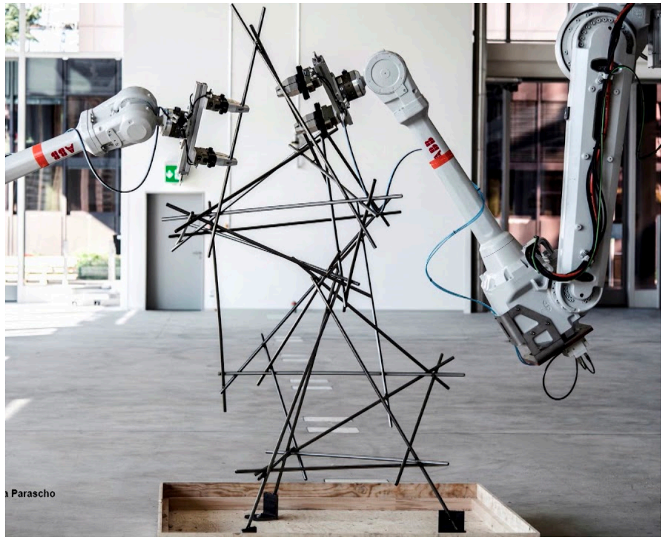


Continous Geometry

[Wang et al. 2019]

Objectives

Assemblability

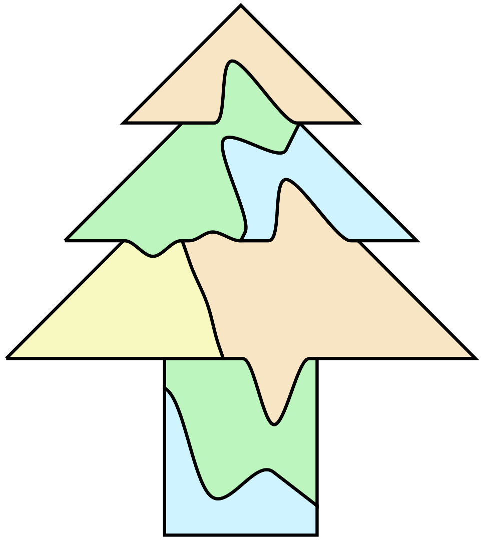


[Parascho et al. 2017]

Fabricatility

[Cignoni et al. 2014]

Stability



[Wang et al. 2021]

Functionality

[Song et al. 2017]

Stability Optimization

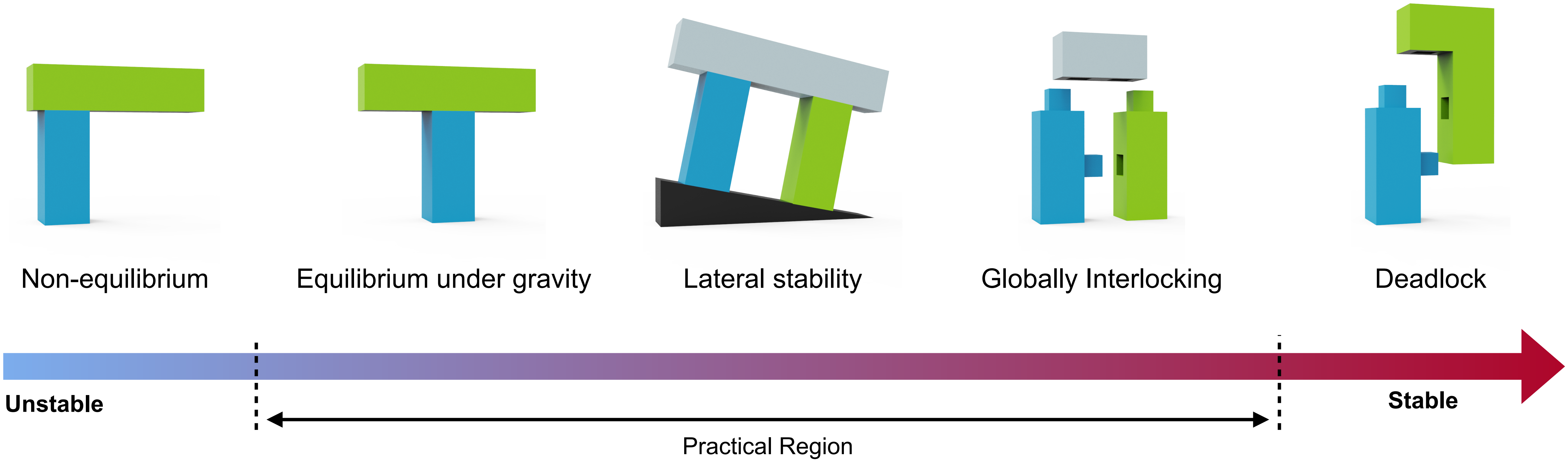
- We mainly focus on designing structurally stable assemblies.
- Stability is the most fundamental requirement for all buildings.
- Designing stable structures that are glue/mortar free is very challenging.

[Nara Todaiji]

[MIT Sean Collier Memorial]

Stability Spectrum

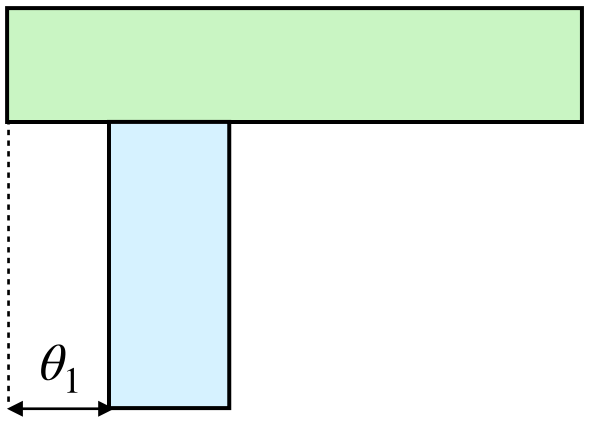
- Besides gravitational equilibrium, we will also cover other types of structural stability.



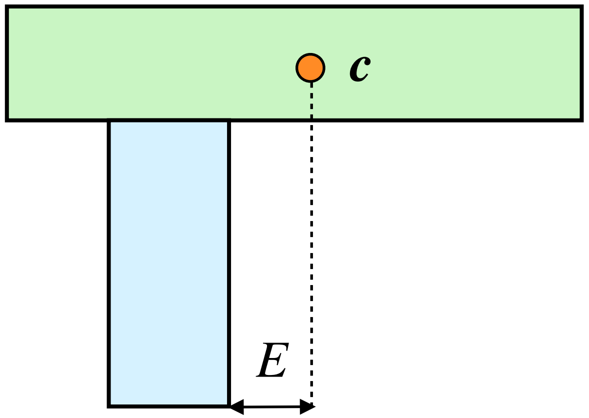
ETH zürich imati
Peng Song Ziqi Wang Marco Livesu

Overview

- Part 1: General stability optimization framework using the gradient information.



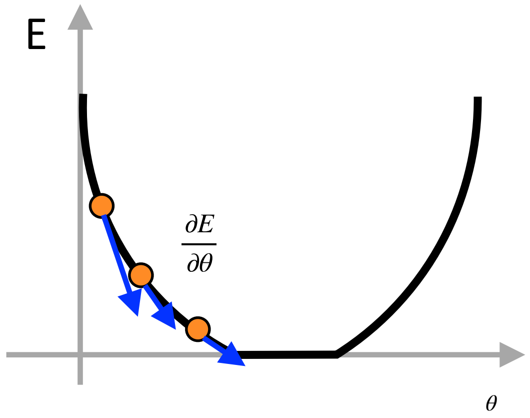
Step 1
Geometrical Property



Step 2
Infeasibility Energy

$$\boldsymbol{\theta} \xrightarrow{\partial} \boldsymbol{c} \xrightarrow{\partial} E$$

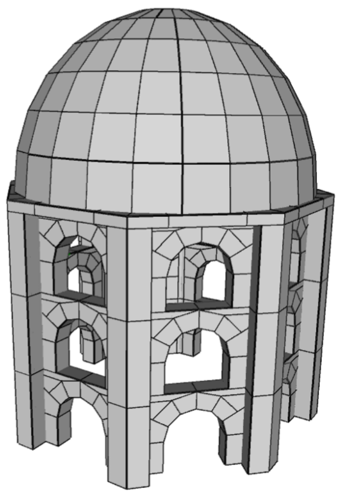
Step 3
Sensitivity Analysis



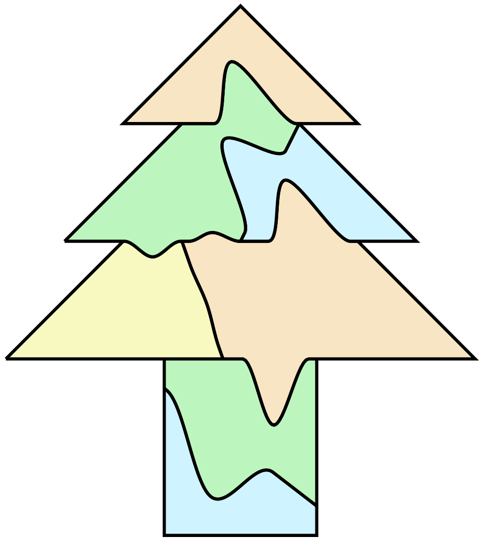
Step 4
Numerical Optimization

Overview

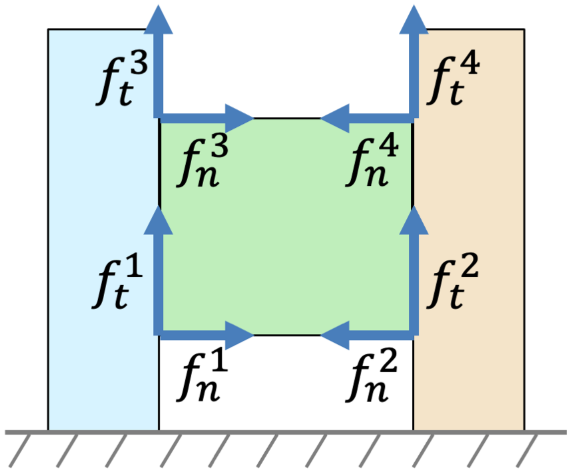
- Part 2: Stability optimization for gravitational equilibrium.
  - Force-based equilibrium method
  - Kinematic-based equilibrium method
  - Friction



[Whiting et al 2009, 2012]



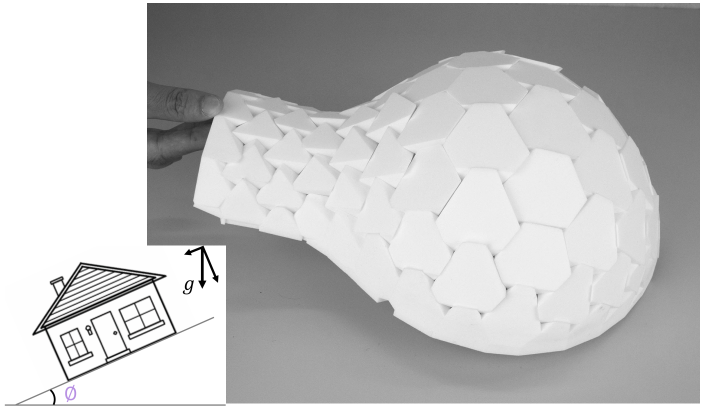
[Wang et al. 2021]



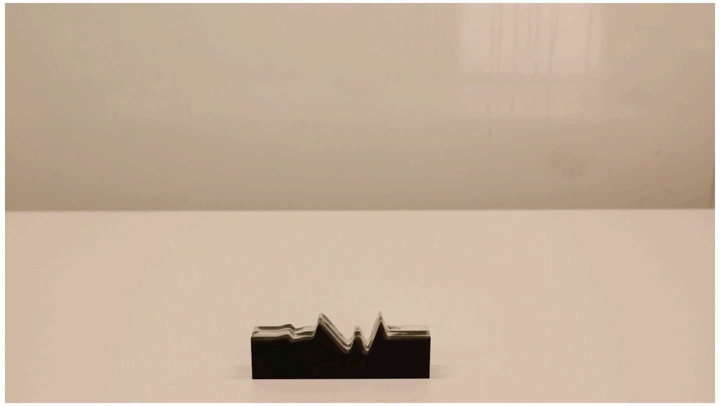
[Yao et al. 2017]

Overview

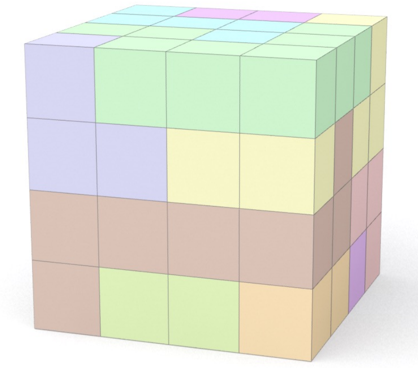
- Part 3: Design for stability under other types of forces
  - Lateral stability
  - Scaffolding-free assembly
  - Globally interlocking



[Wang et al. 2019]



[Wang et al. 2021]

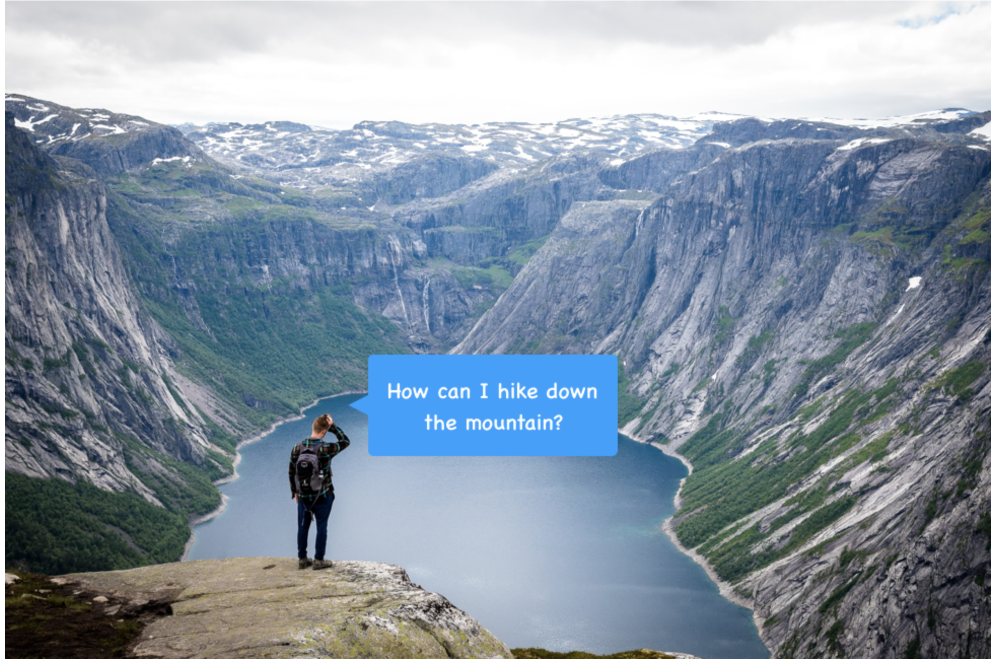


[Wang et al. 2018]

Part 1: General Stability Optimization Framework

Gradient-based Optimization

- Gradient-based optimization is the most common approach to solve the inverse design problem.

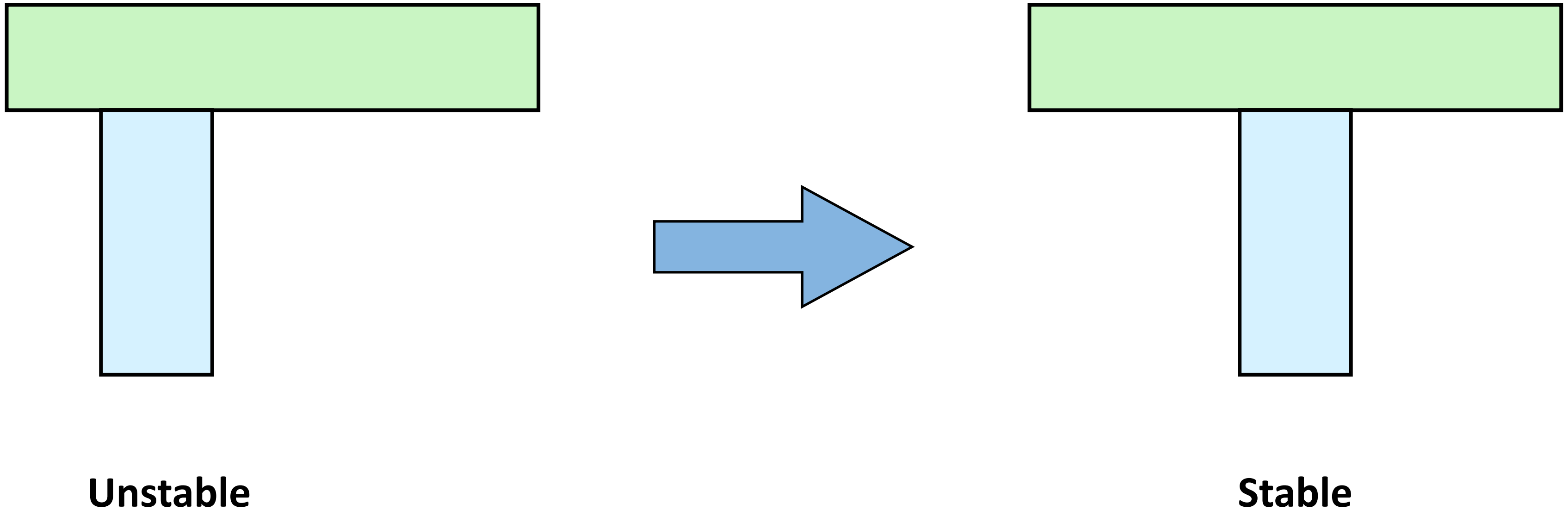


Strategy #1: Take a random downhill slope. **Slow**

Strategy #2: Take the *steepest slope*! **Fast**

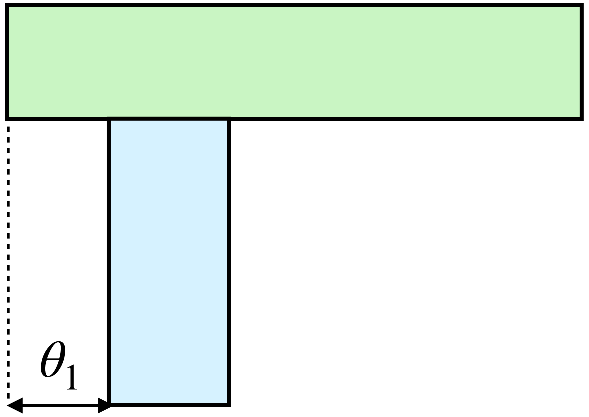
Stability Optimization

- By alternating the parts' geometry, making the assembly stable under certain loading conditions.

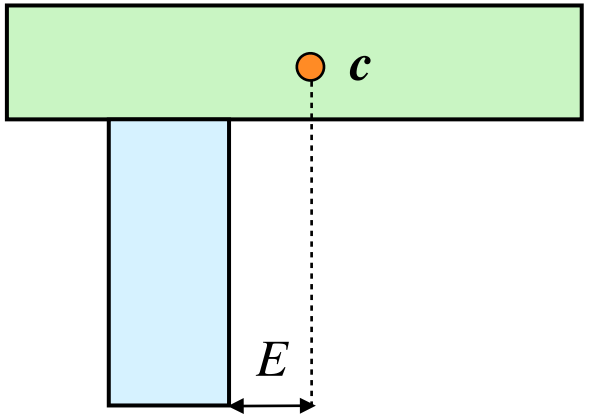


Gradient-based Stability Optimization

- Gradient-based stability optimization has four main components:



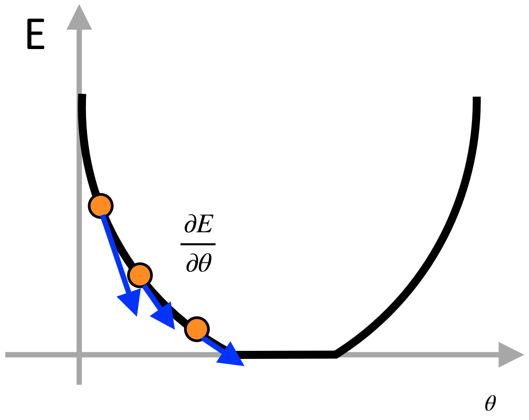
Step 1
Geometrical Property



Step 2
Infeasibility Energy

$$\boldsymbol{\theta} \xrightarrow{\partial} \boldsymbol{c} \xrightarrow{\partial} E$$

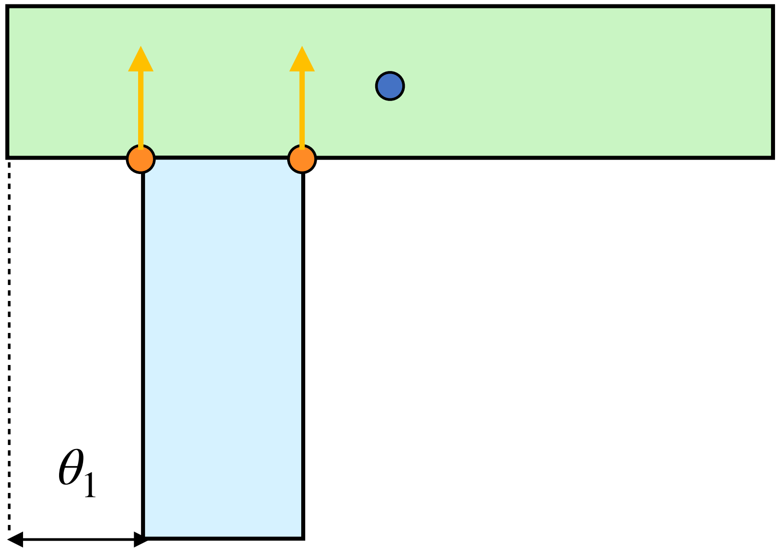
Step 3
Sensitivity Analysis



Step 4
Numerical Optimization

Step #1 Geometrical Property

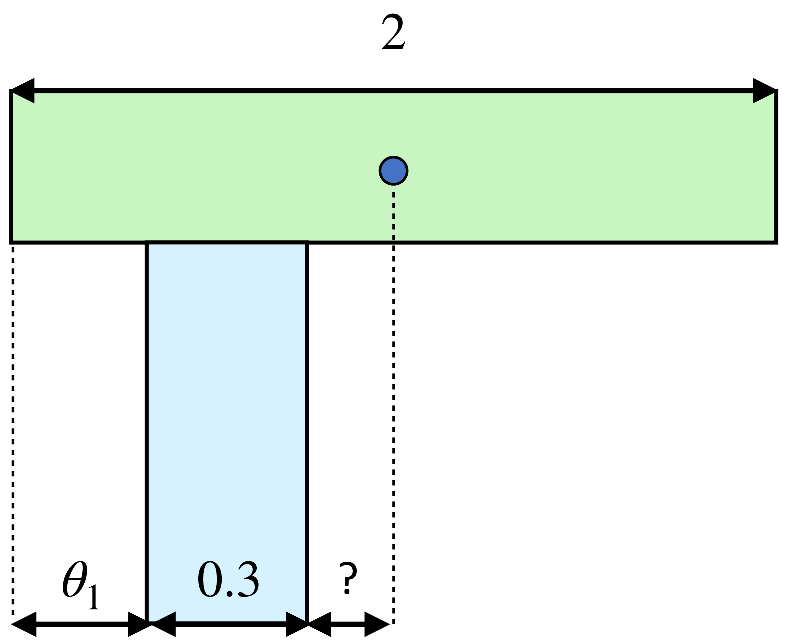
- Compute necessary geometrical properties for stability analysis.



1. Contact Points
2. Contact Normals
3. Parts' Centroids
4. Parts' Volumes

Step #2 Infeasibility Measurement

- Compute the infeasibility energy which measures how unstable the structure is.



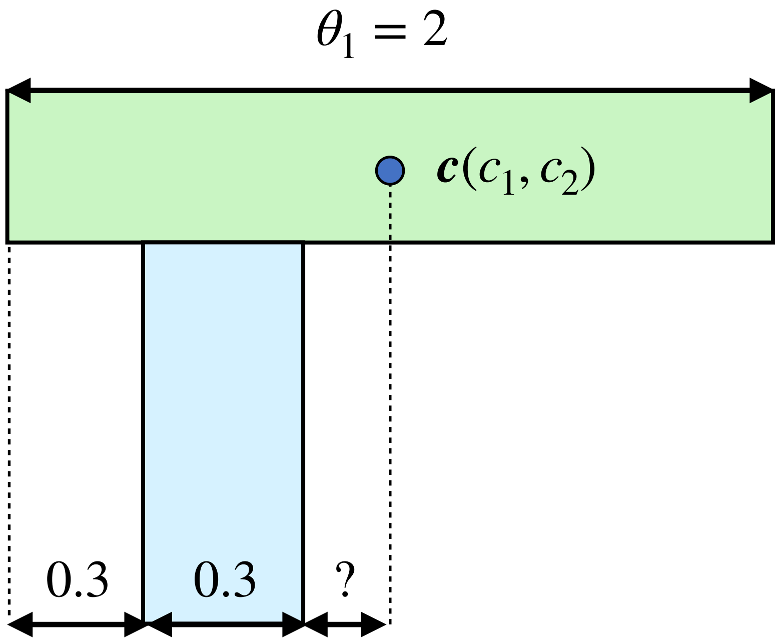
Infeasbility Energy

$$E = \left\| \; — \; \right\|^2$$

$$= (0.7 - \theta)^2$$

Step #3 Sensitivity Analysis

- Compute the infeasibility energy's gradient/hessian with respect to the design parameters.



Gradient: $\dfrac{\partial E}{\partial \theta_1}$

Chain Rule: $\dfrac{\partial E}{\partial \theta_1} = \dfrac{\partial E}{\partial c_1}\dfrac{\partial c_1}{\partial \theta_1}$

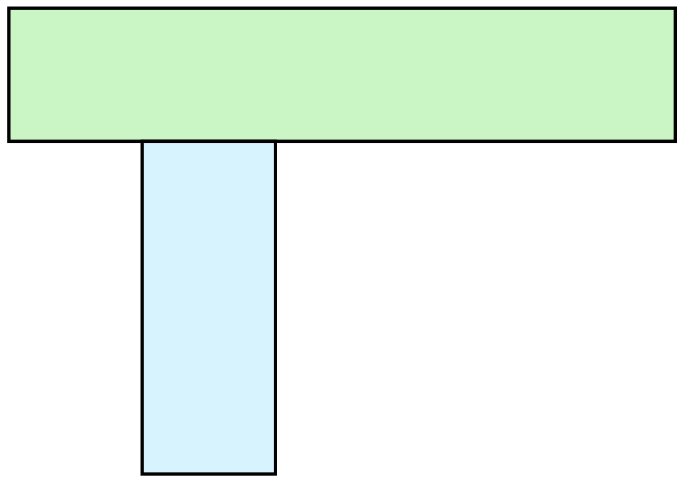
Step #4 Numerical Optimization

- Various numerical optimization tools can be used to solve the inverse design problem.

	Gradient Descent	Netwon Method	Quasi-Netwon Method
Data	Gradient	Hessian	Gradient
Speed	Slow	Fast	Medium
Code	Easy-to-implement	Hard-to-implement	Easy-to-implement

Part 2: Stability optimization for gravitational equilibrium

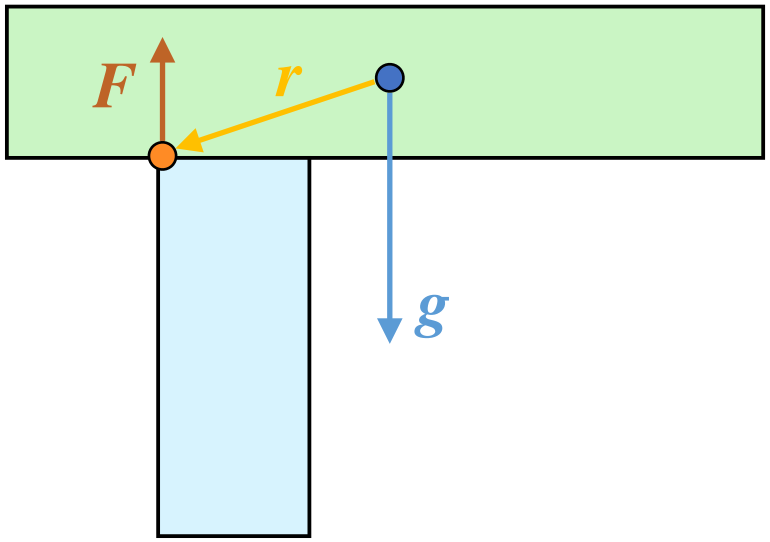
Assumptions



1. Parts are rigid body.
2. Friction is ignored.
3. The bottom part (blue) is fixed.

Recap: Rigid Body Equilibrium

- Rigid body equilibrium can check whether the internal and external forces/torque of a given structure are balanced.



Force Balance: $\sum F + g = 0$

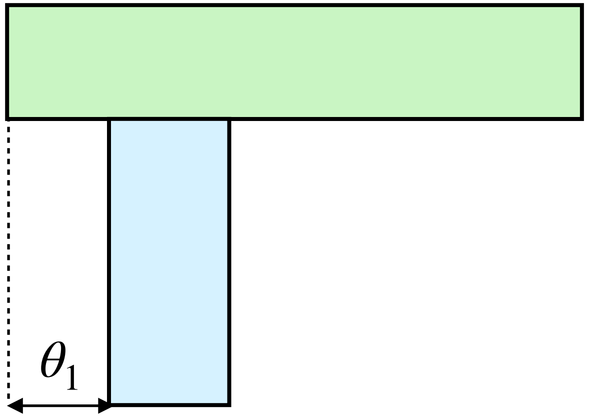
Torque Balance: $\sum r \times F = 0$

Non-negative: $F \geq 0$

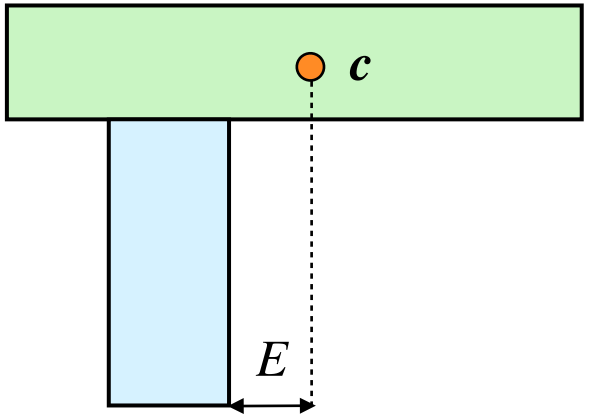
Problems: only provides a binary result (yes/no).

Recap: Gradient-based Stability Optimization

- The most critical step is to compute faithful infeasibility energy.



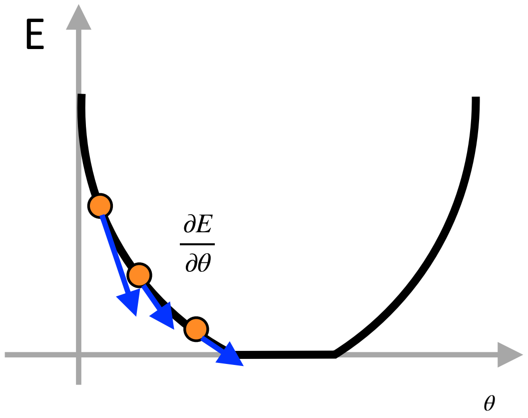
Step 1
Geometrical Property



Step 2
Infeasibility Energy

$$\boldsymbol{\theta} \xrightarrow{\partial} \boldsymbol{c} \xrightarrow{\partial} E$$

Step 3
Sensitivity Analysis

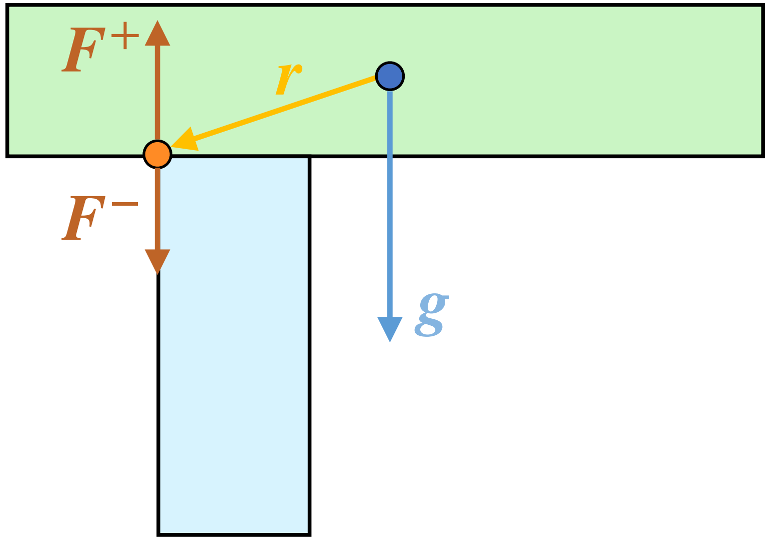


Step 4
Numerical Optimization

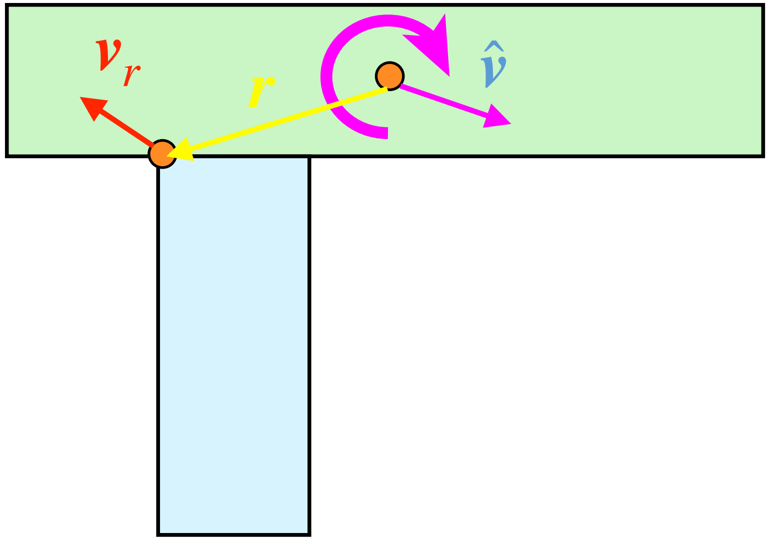
Equilibrium Infeasibility Energy

- Two ways of computing infeasibility energy for equilibrium problems.

Force-based Equilibrium Method

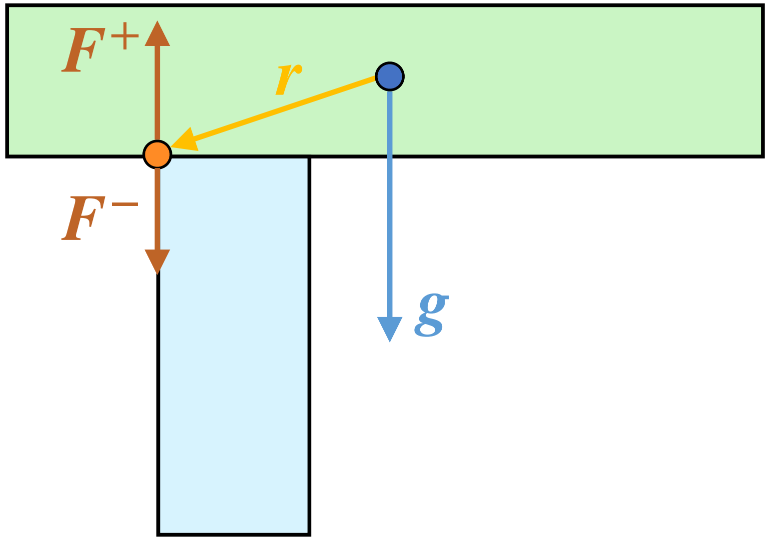


Kinematic-based Equilibrium Method



Force-based Infeasibility Measurement

- Split each contact force $\boldsymbol{F}$ into the positive and negative parts $\boldsymbol{F}^+, \boldsymbol{F}^-$.
- The norm of the negative contact force is used to compute the infeasibility energy.



Minimizing tension: $\min \sum ||F^-||^2$

Force Balance: $\sum \boldsymbol{F} + \boldsymbol{g} = 0$

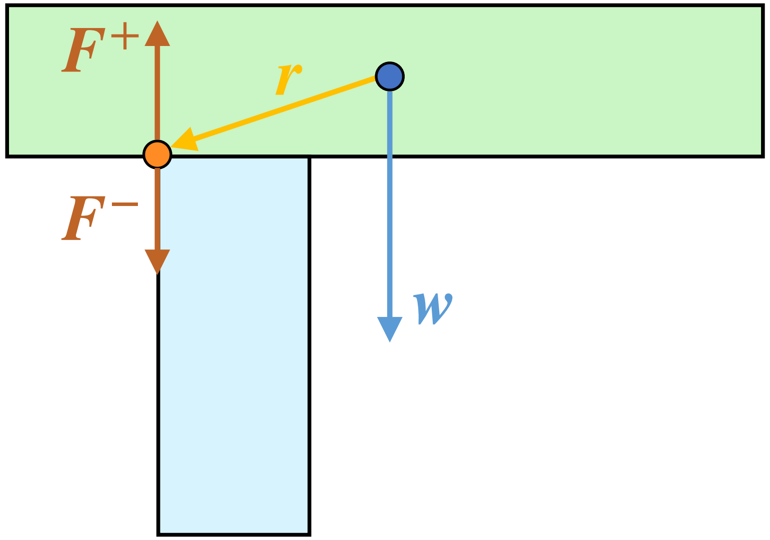
Torque Balance: $\sum \boldsymbol{r} \times \boldsymbol{F} = 0$

Non-negative: $\boldsymbol{F}^+, \boldsymbol{F}^- \geq 0$

$\boldsymbol{F} = \boldsymbol{F}^+ - \boldsymbol{F}^-$

Quadratic Programming

- The infeasibility energy can be computed by a quadratic programming solver.



Minimizing tension: $\min \sum ||F^-||^2$

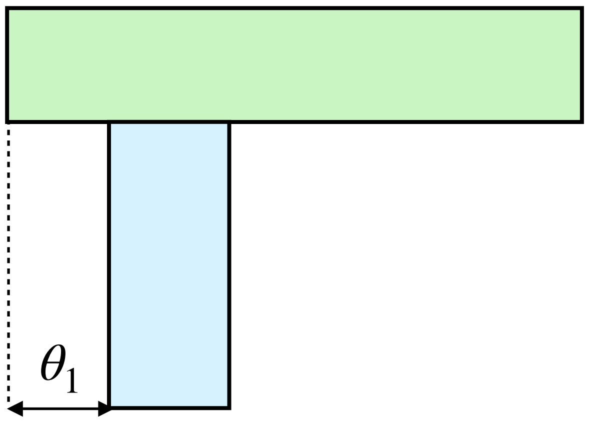
Force/Torque Balance: $A_{eq}F + w = 0$

Non-negative: $F^+, F^- \geq 0$

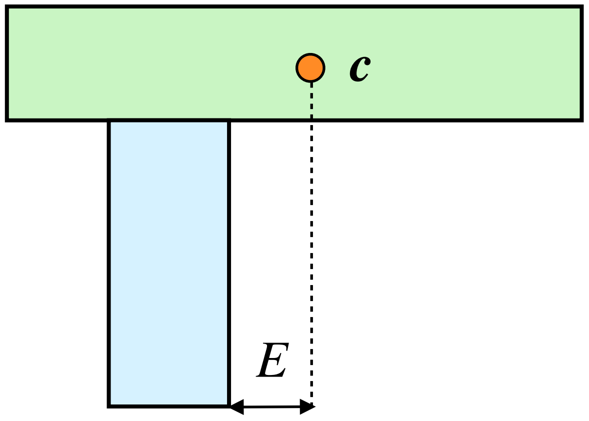
$F = F^+ - F^-$

Gradient-based Stability Optimization

- The next challenging step is to compute gradient using sensitivity analysis.



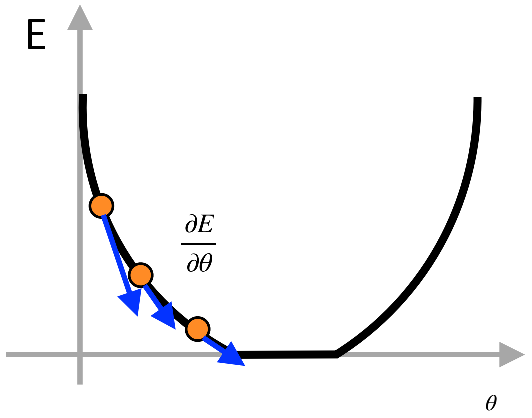
Step 1
Geometry Computing



Step 2
Infeasibility Energy

$$\boldsymbol{\theta} \xrightarrow{\partial} \boldsymbol{c} \xrightarrow{\partial} E$$

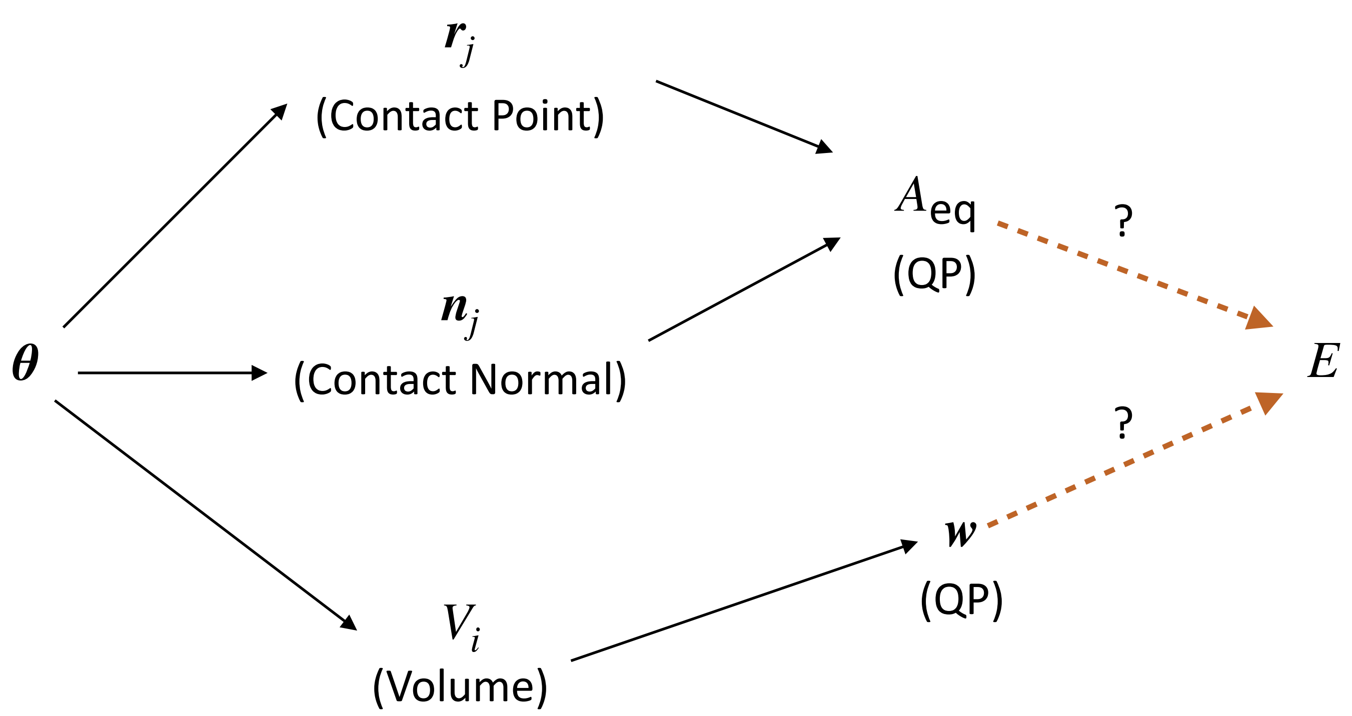
Step 3
Sensitivity Analysis



Step 4
Numerical Optimization

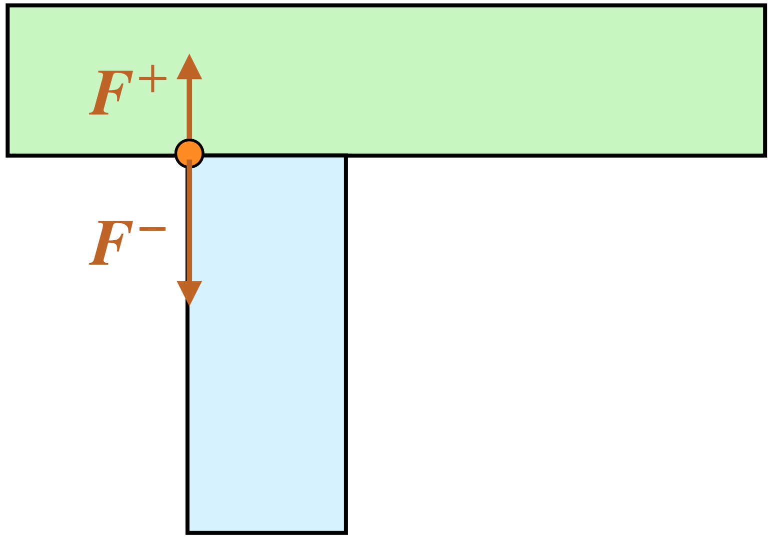
Chain Rule

- The chain rule help compute the gradient.
- However, the infeasibility energy's gradient with respect to the QP's coefficients are missing.



Sensitivtiy Analysis of QP

- Local perturbation of the geometry will only change the resulting force slightly.

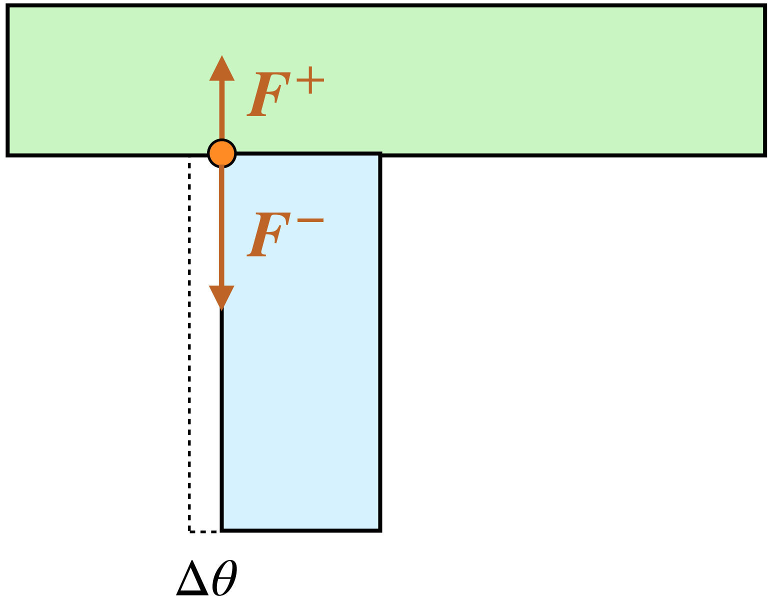


$F^+ = 1.0$

$F^- = 1.5$

Sensitivtiy Analysis of QP

- Local perturbation of the geometry will only change the resulting force slightly.



$$F^+ = 1.01$$

$$F^- = 1.49$$

Sensitivtiy Analysis of QP

- Applying region trust algorithm to replace inequalities with equalities.
- A closed-form solution is available for the QP problem with only equality constraints.

$$E(A_{\text{eq}}, w) = \min \sum ||\boldsymbol{F}^-||^2$$

$$A_{\text{eq}}\boldsymbol{F} + \boldsymbol{w} = 0$$

~~$\boldsymbol{F}^+, \boldsymbol{F}^- \geq 0$~~

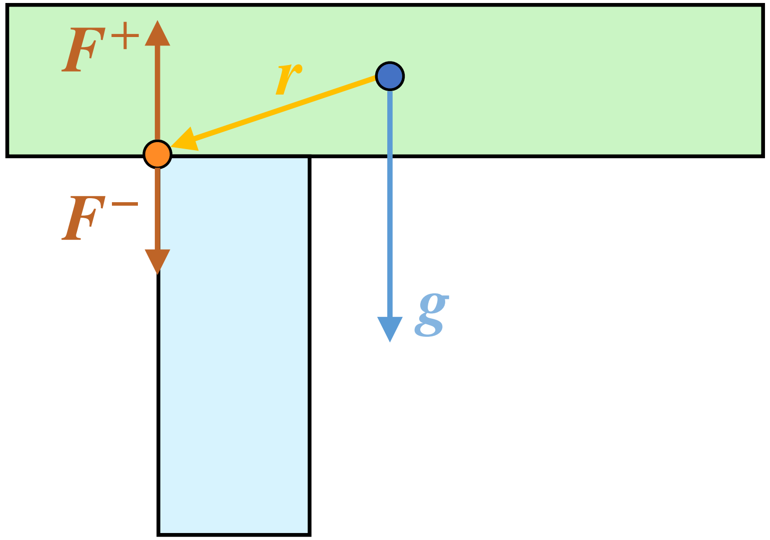
$$\boldsymbol{F}_i^+ = 0, \quad \boldsymbol{F}_j^- = 0$$

Closed-Form Solution

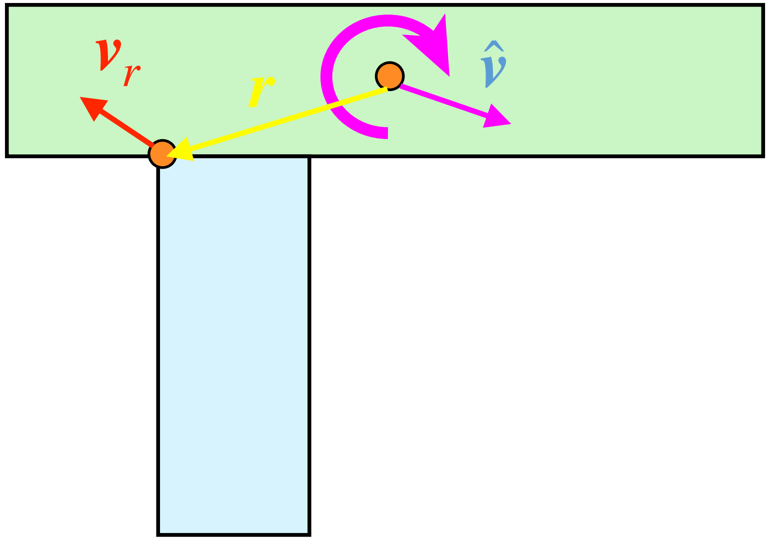
Kinematic-based Equilibrium Method

- Kinematic-based method measures infeasibility in the motion space.

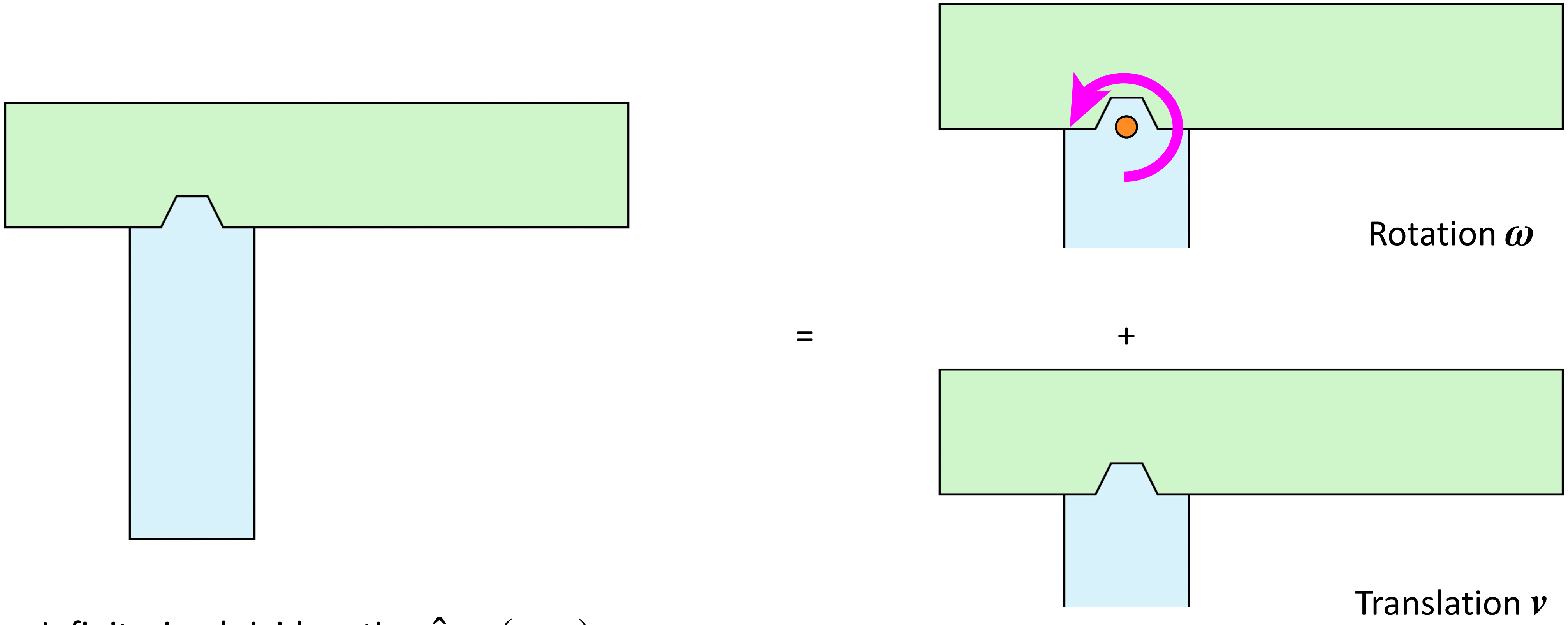
Force-based Equilibrium Method



Kinematic-based Equilibrium Method

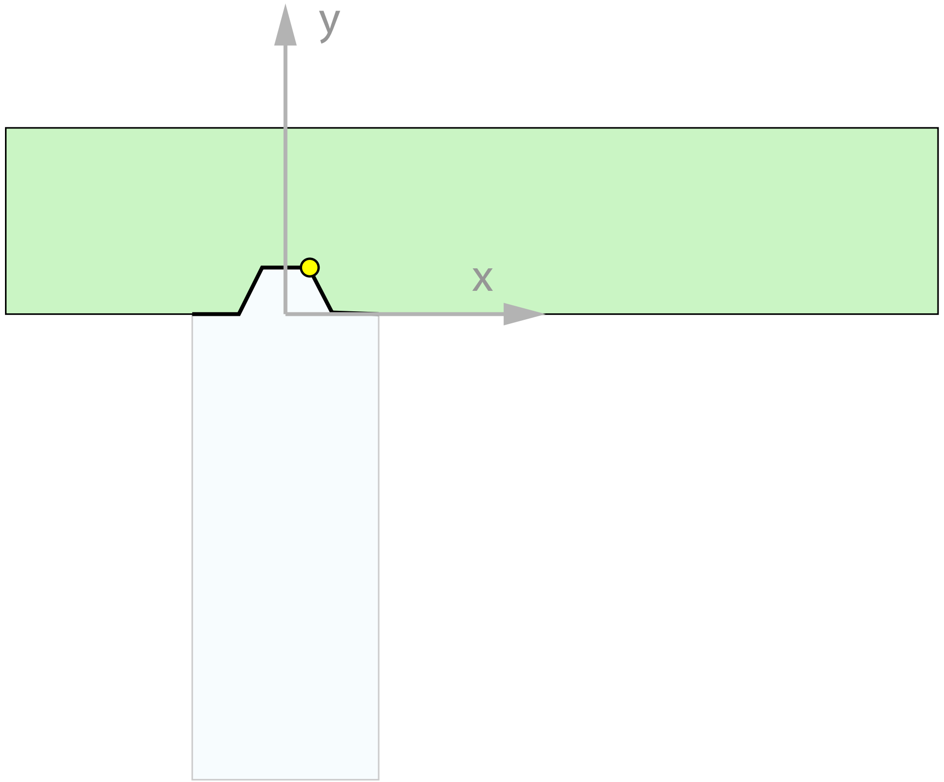


Infinitesimal Rigid Motion



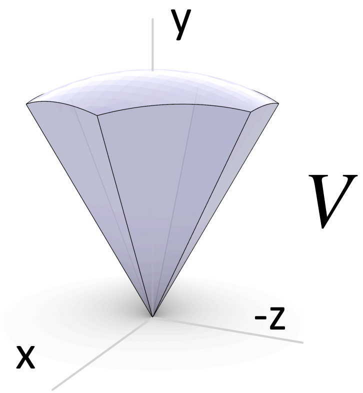
Infinitesimal rigid motion $\hat{\boldsymbol{v}} = (\boldsymbol{v}, \boldsymbol{\omega})$

Motion Space



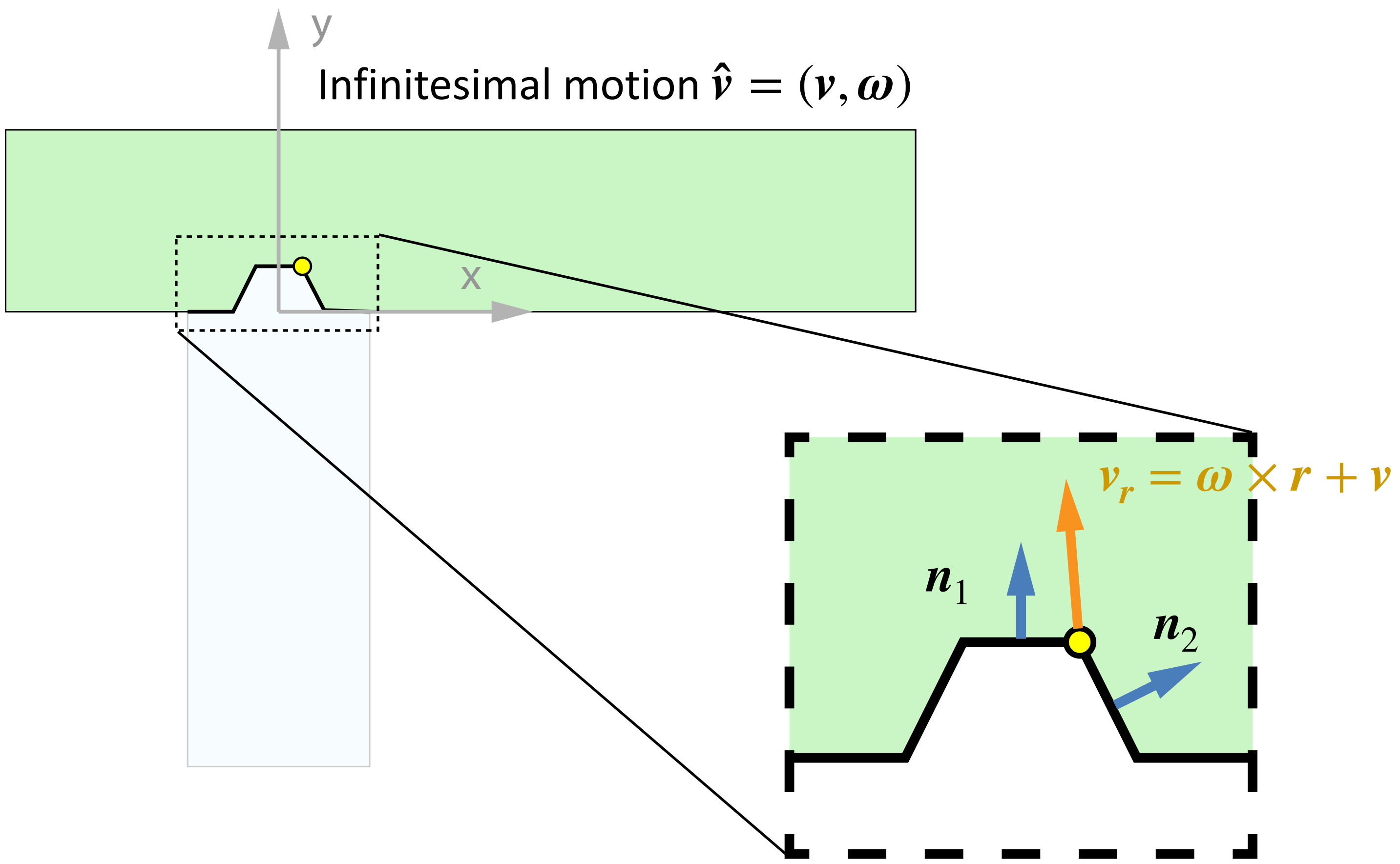
The motion space V of green part

=



{collision-free infinitesimal rigid motions $\hat{\boldsymbol{v}}$}

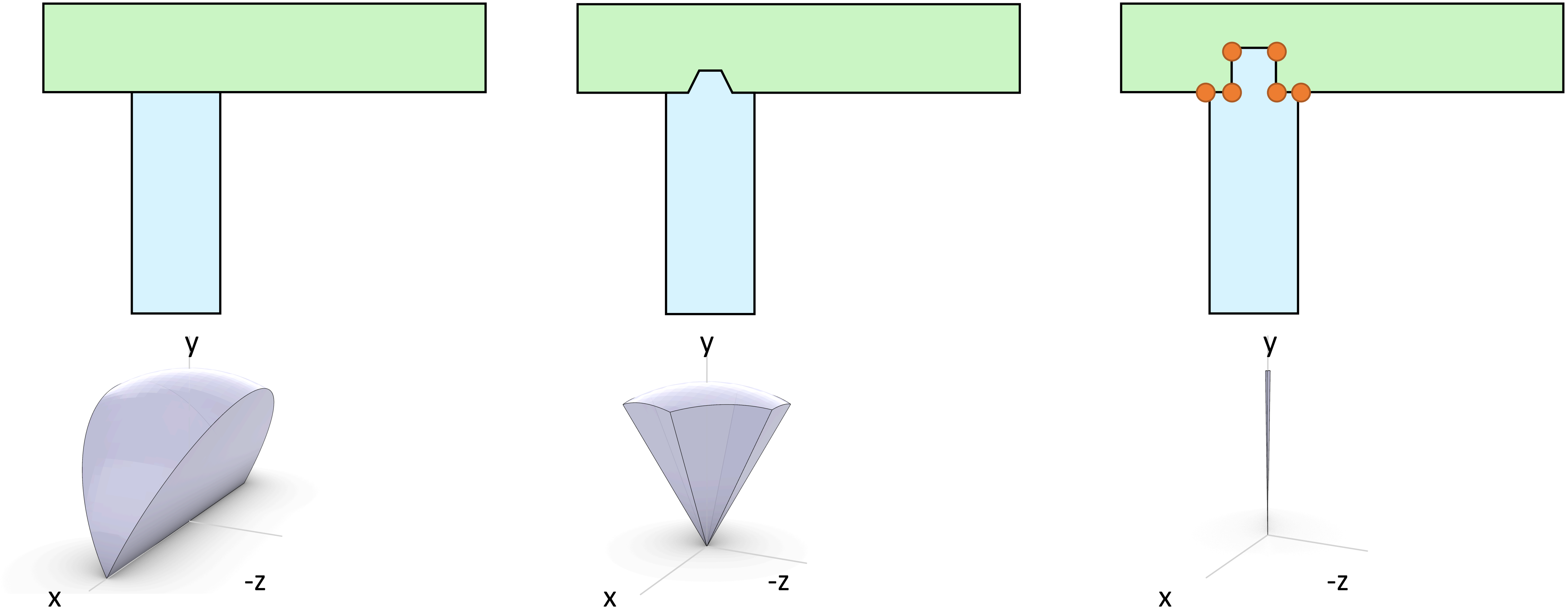
Non-collision constraints



$$\boldsymbol{v}_r \cdot \boldsymbol{n} \geq 0$$

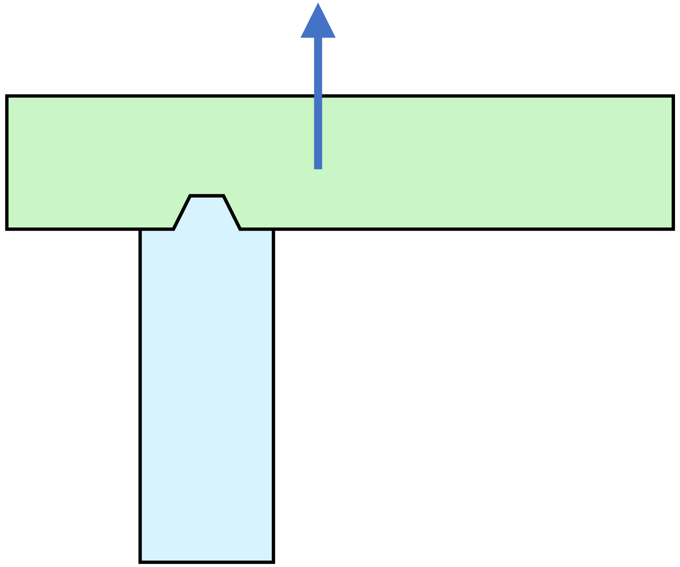
Non collision constraints

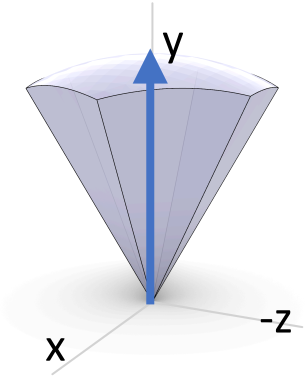
Motion Cone of Contacts



Physically Feasible Motion

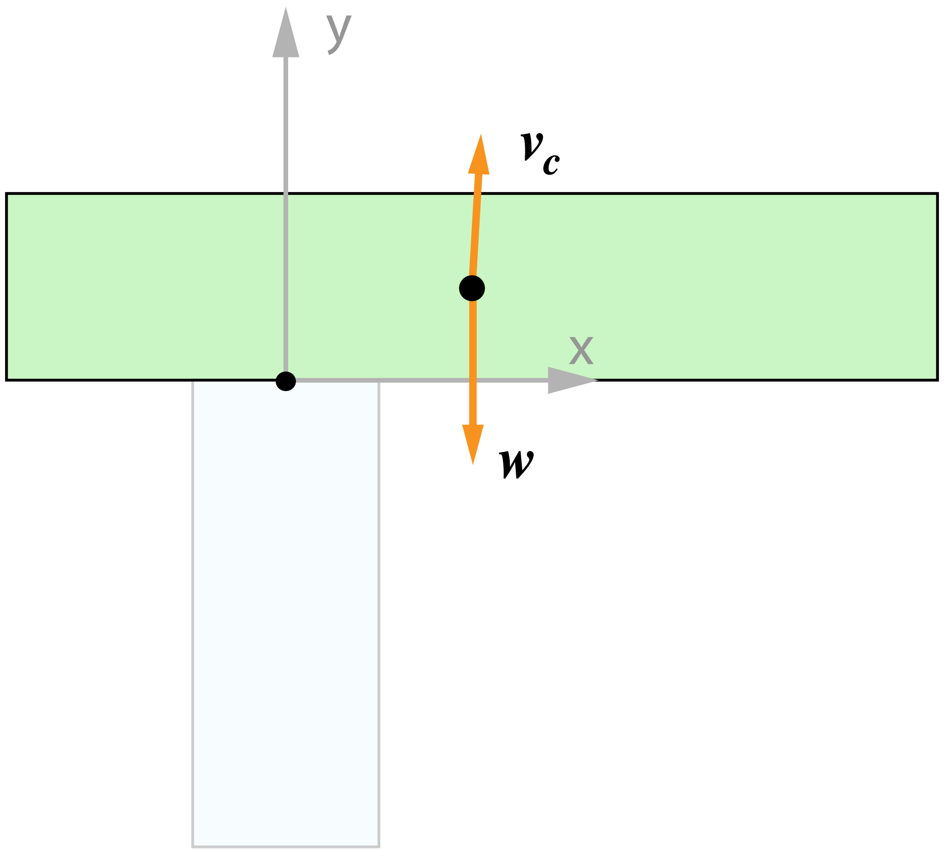
- Not every motion in the motion cone is physically plausible.





The translation along +y direction is not physically achievable.

Feasible Motion Space



Infinitesimal motion $\hat{v} = (v, \omega)$

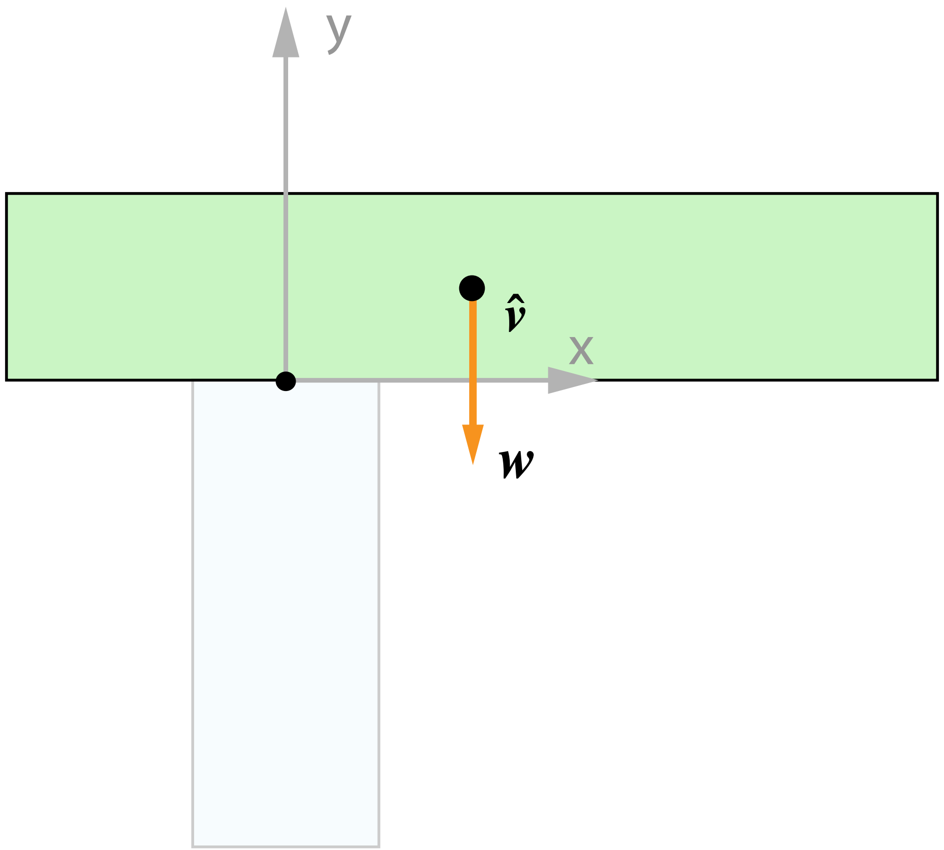
Velocity v_c at part's centre of mass

decreases its gravitational potential

$$v_c \cdot g > 0$$

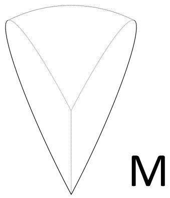
$$\hat{v} \cdot w > 0$$

Feasible Motions



Assembly is in equilibrium when

$$\begin{cases} \hat{v} \cdot w > 0 \\ \hat{v} \in \text{Motion Cone} \end{cases}$$

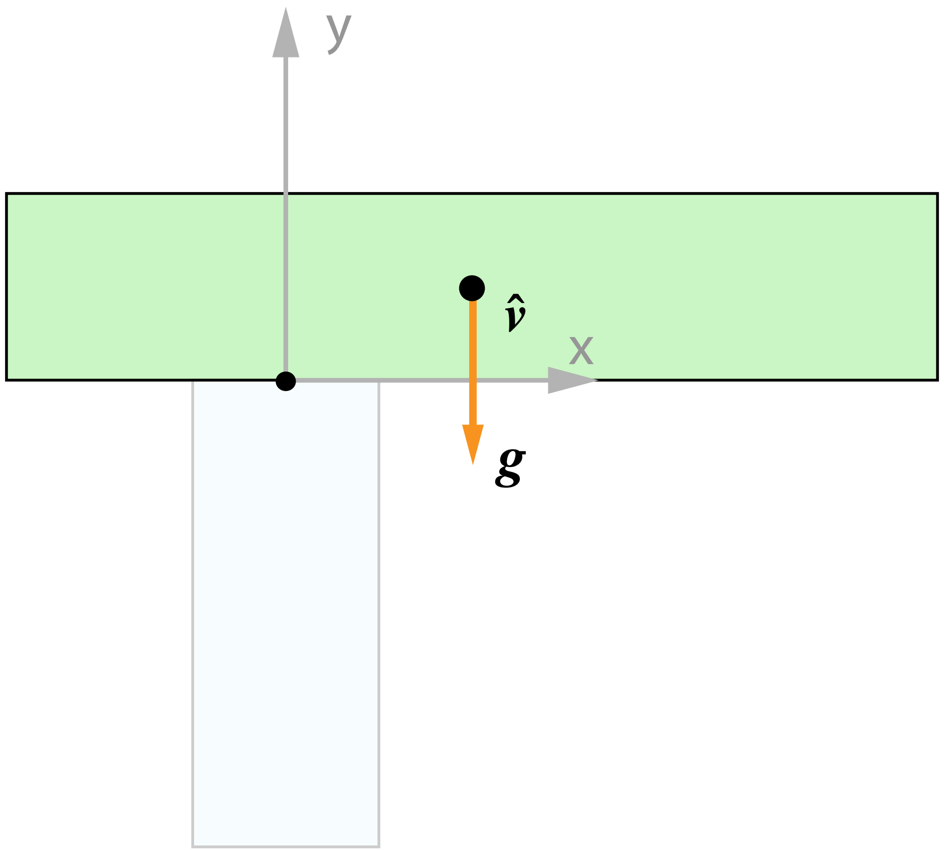


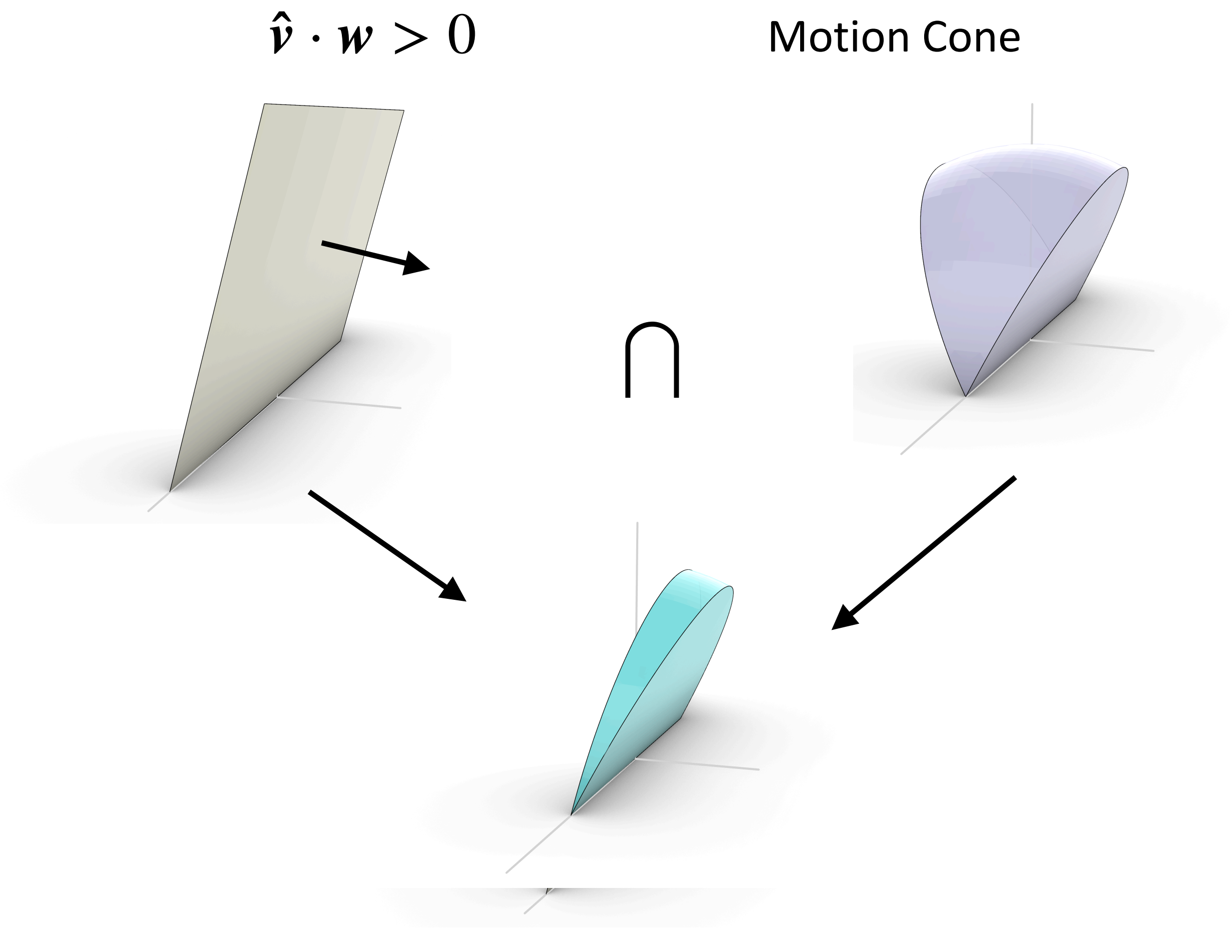
Motion Cone

does not have solutions.

Infinitesimal motion $\hat{v} = (v, \omega)$

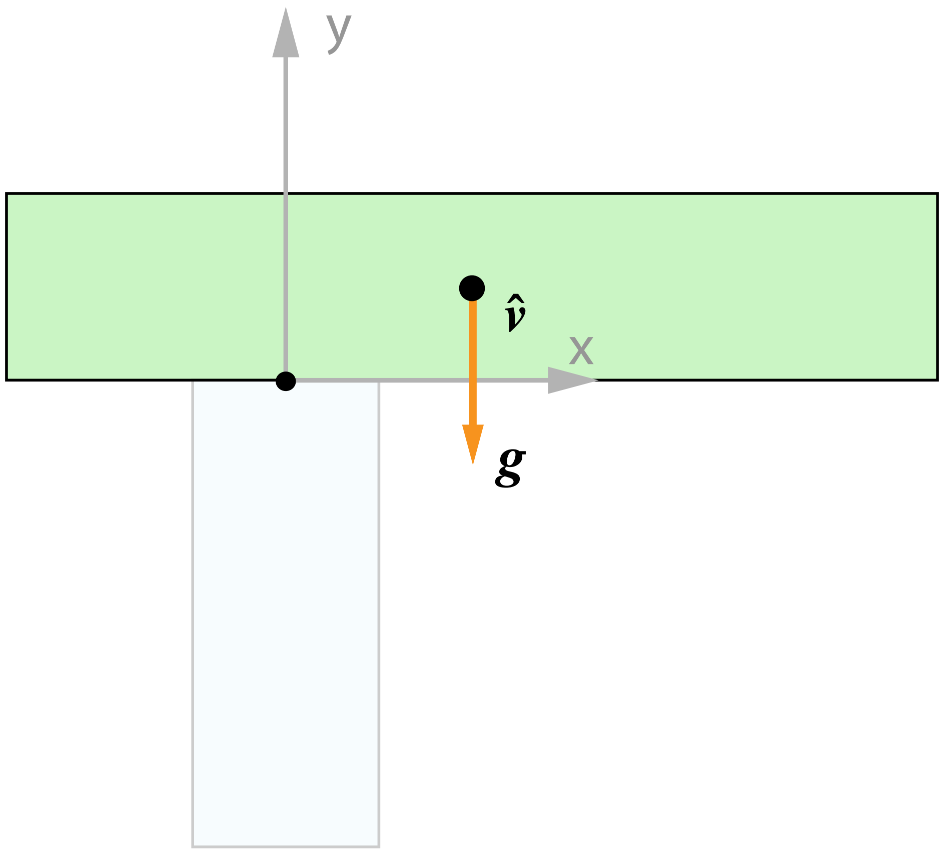
Feasible Motion Space



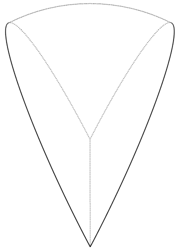


Infinitesimal motion $\hat{v} = (v, \omega)$

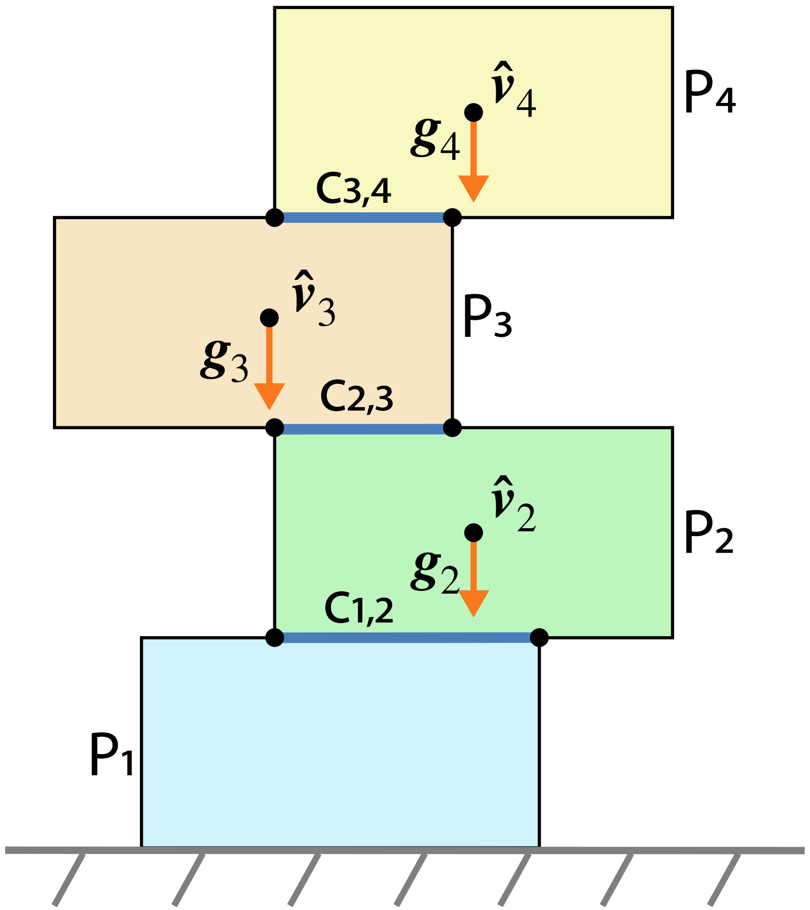
Infeasibility Measurement



$$\max \quad \boldsymbol{w} \cdot \hat{\boldsymbol{v}} - \frac{1}{2}\hat{\boldsymbol{v}} \cdot \hat{\boldsymbol{v}}$$

$\hat{\boldsymbol{v}} \in$ 

Infeasibility Measurement for Assembly



$$\hat{v} = \begin{bmatrix} \hat{v}_2 \\ \hat{v}_3 \\ \hat{v}_4 \end{bmatrix}$$

$$\hat{g} = \begin{bmatrix} \hat{g}_2 \\ \hat{g}_3 \\ \hat{g}_4 \end{bmatrix}$$

Infeasibility Measurement

$$\max \boldsymbol{w} \cdot \hat{v} - \frac{1}{2}\hat{v} \cdot \hat{v}$$

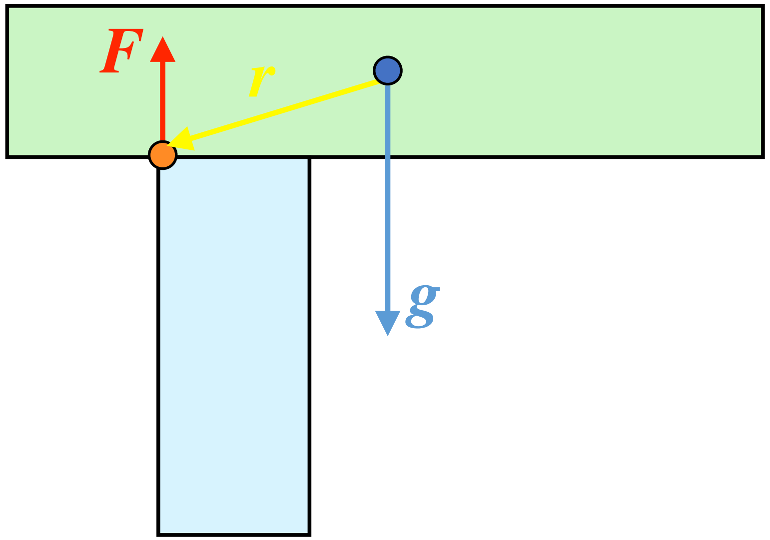
$$\hat{v}_2 \in V(C_{1,2})$$

$$\hat{v}_3 - \hat{v}_2 \in V(C_{2,3})$$

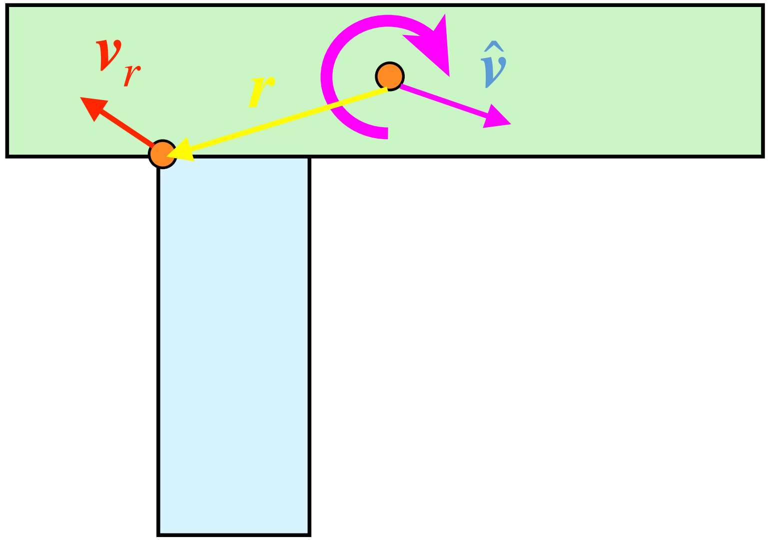
$$\hat{v}_4 - \hat{v}_3 \in V(C_{3,4})$$

Static-Kinematic Duality

- The correctness of the kinematic-based method is due to the static-kinematic duality.



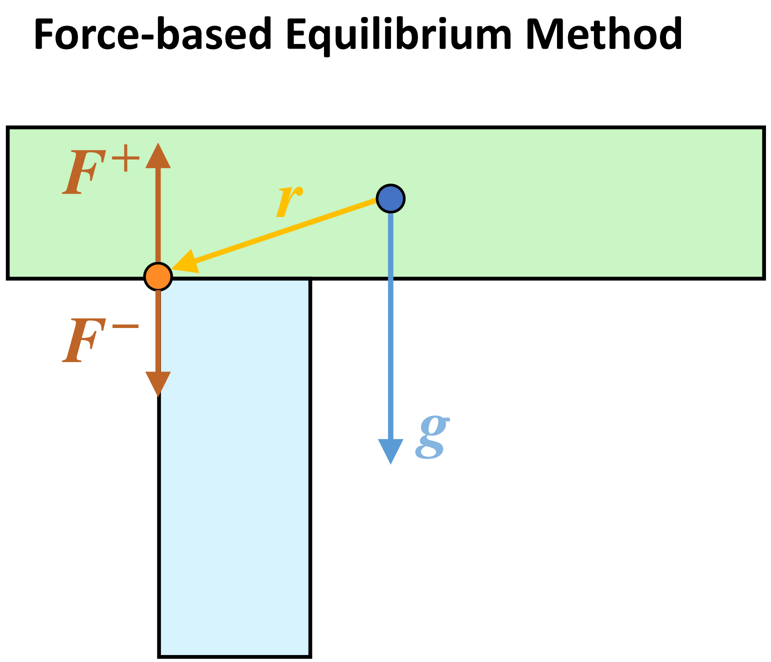
Statics

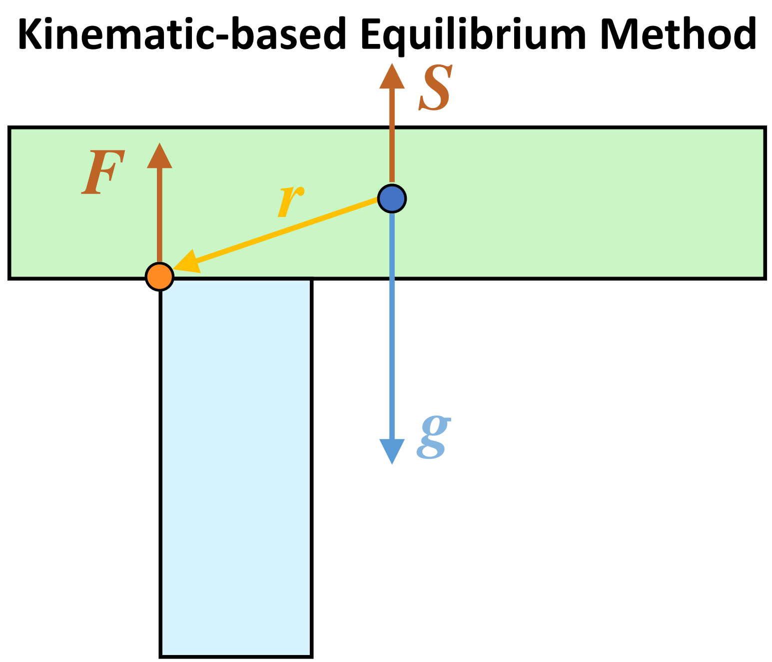


Kinematics

Static-Kinematic Duality

- The kinematic-based method can be reformulated using forces.

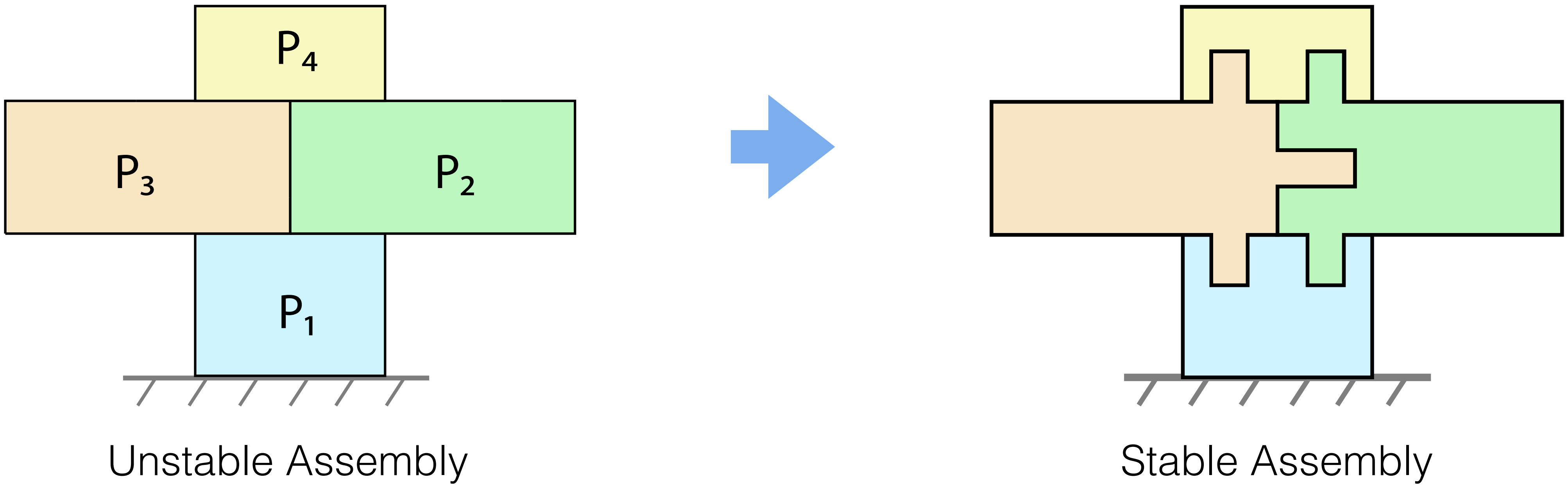




Reformulate: Non-negative Condition Force/Torque Balance Condition

Structural Stability Optimization

- Modifying the assembly for improved structural stability.

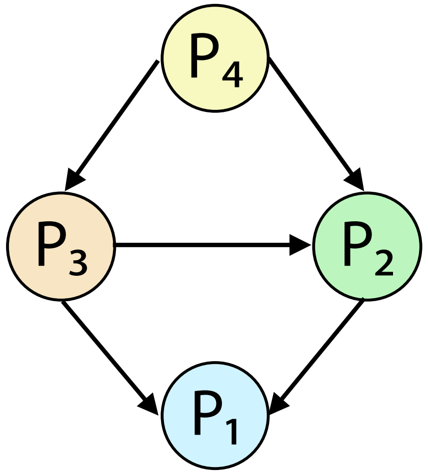


What is a good representation for structural stability optimization?

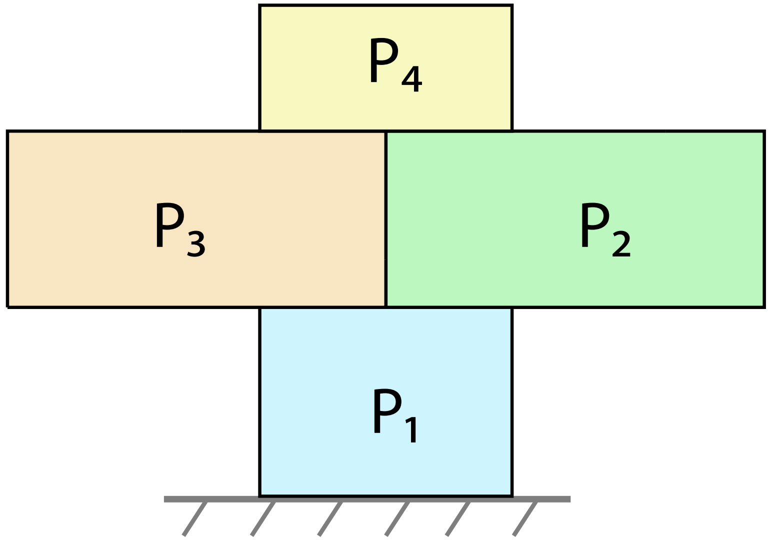
Peng Song Ziqi Wang Marco Livesu

Representation for Stability Analysis

- Both representations have their own drawbacks.



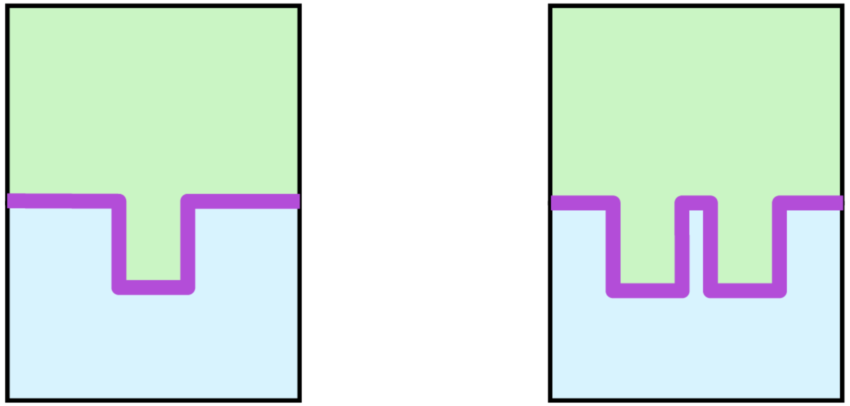
Part Graph



Part Geometry

Geometric-based Representation

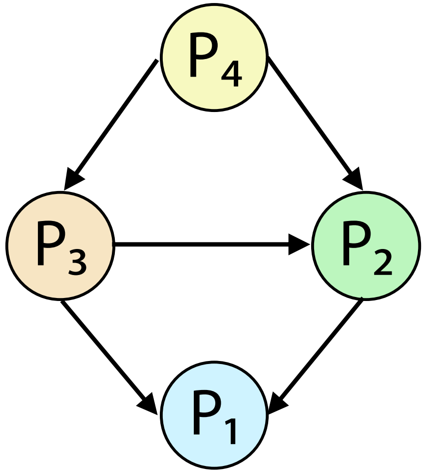
- Geometric-based Representation may have redundancy.



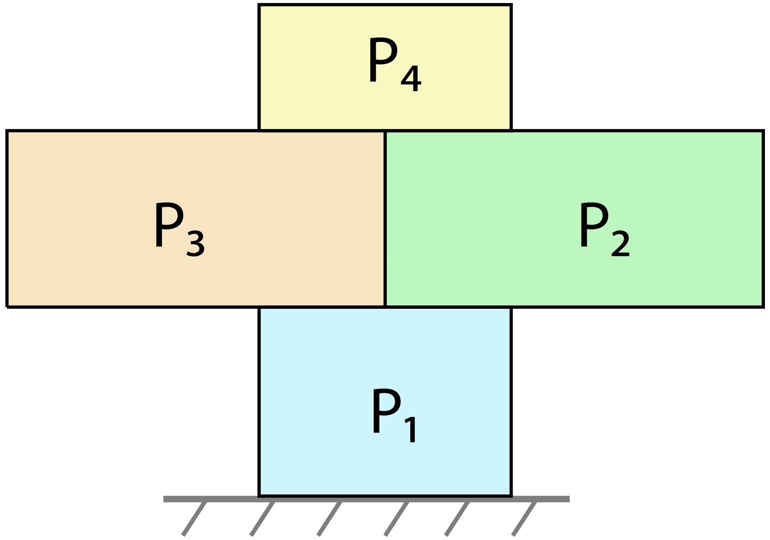
The two assemblies have the same structural stability.

Graph-based Representation

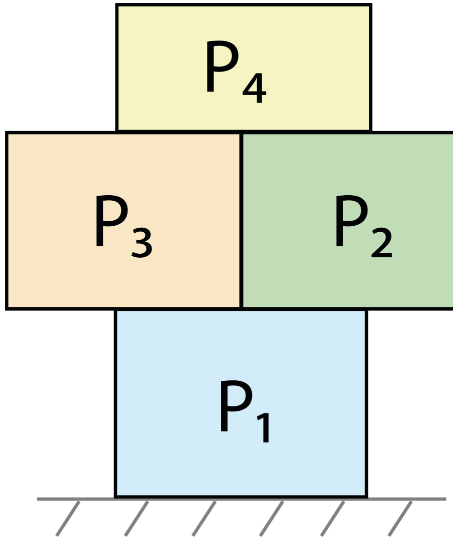
- Graph-based representation is not adequate for structural stability analysis.



Part Graph



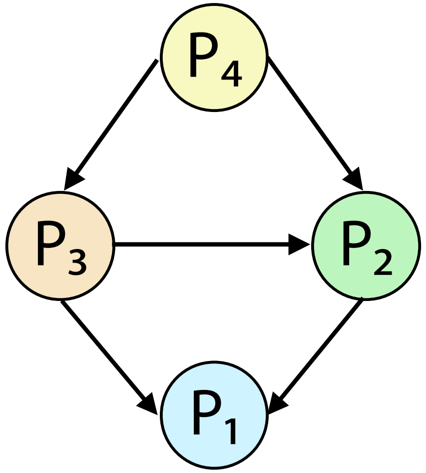
Unstable Assembly



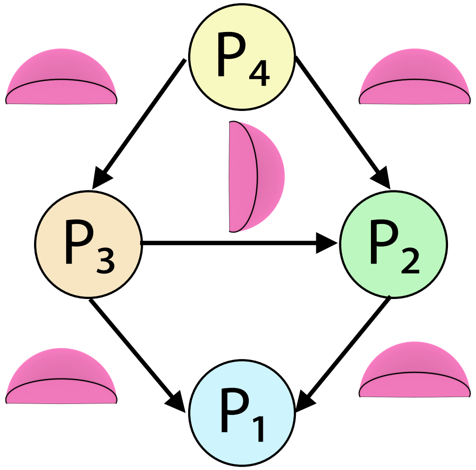
Stable Assembly

Motion-based Representation

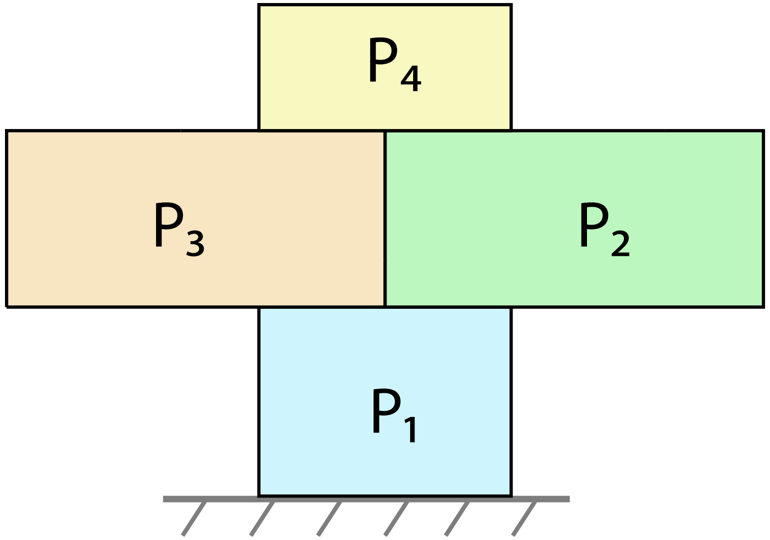
- We propose a motion-based representation which is a condensed representation for measuring structural stability of assemblies.



Part Graph



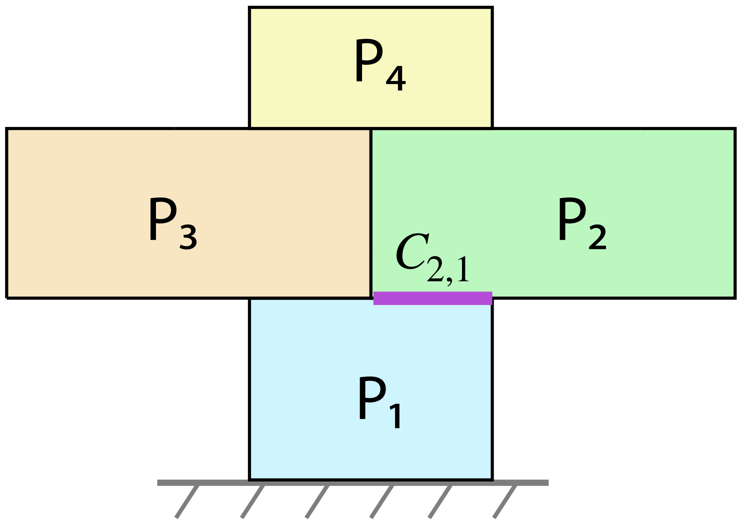
Graph-based Representation

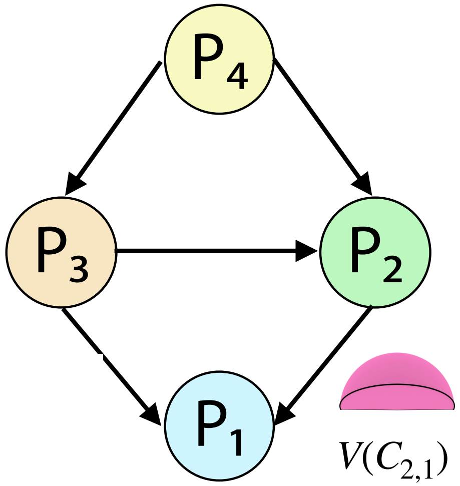


Part Geometry

Motion-based Representation

- Our motion-based representation is an augmented part graph with motion cones at its edges.

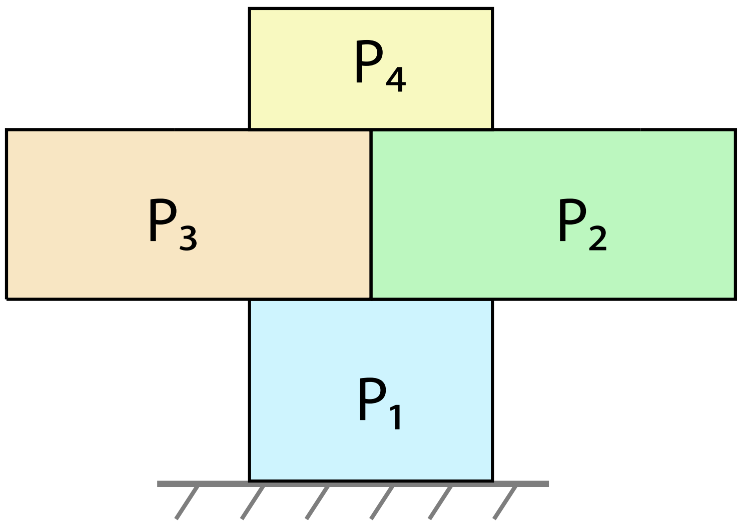


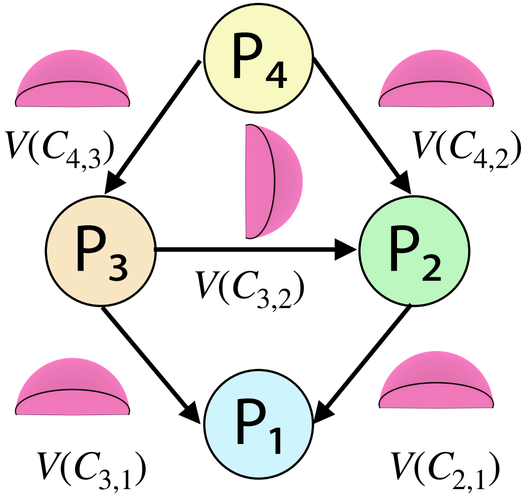


*Arrow means P_2 is installed after P_1

Motion-based Representation

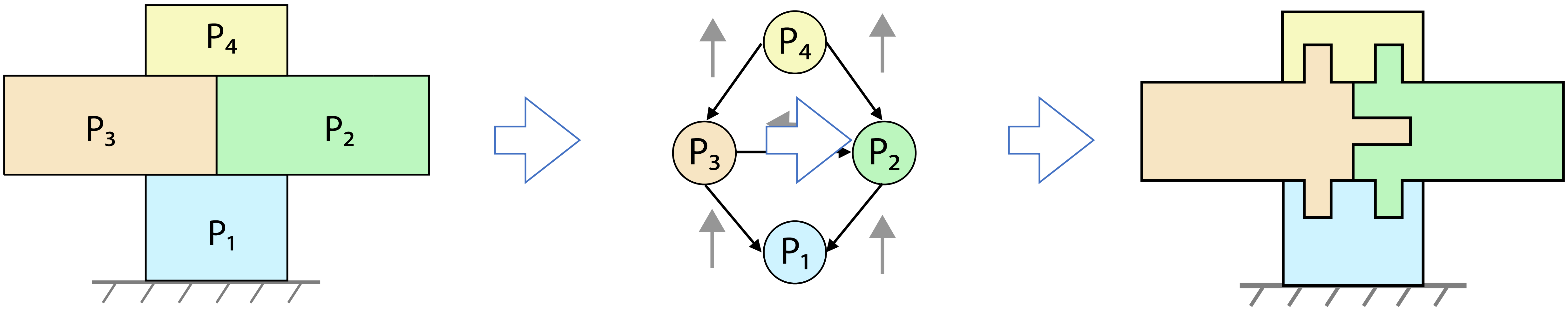
- Because of the duality between statics and kinematics, our motion-based representation can test for equilibrium.





Kinematic-Geometric Design Framework

- Decoupling motion and geometry.

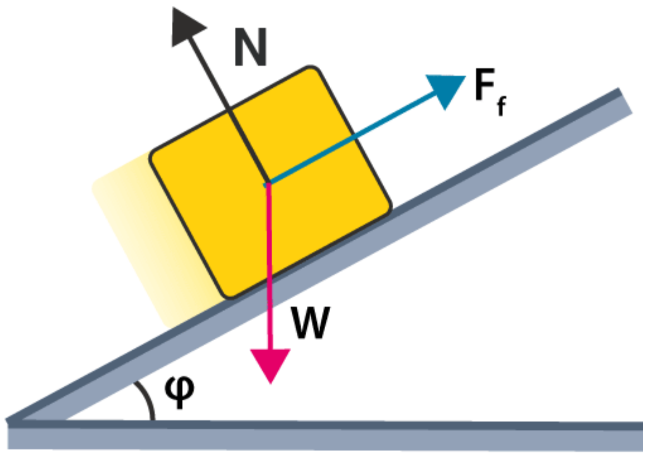


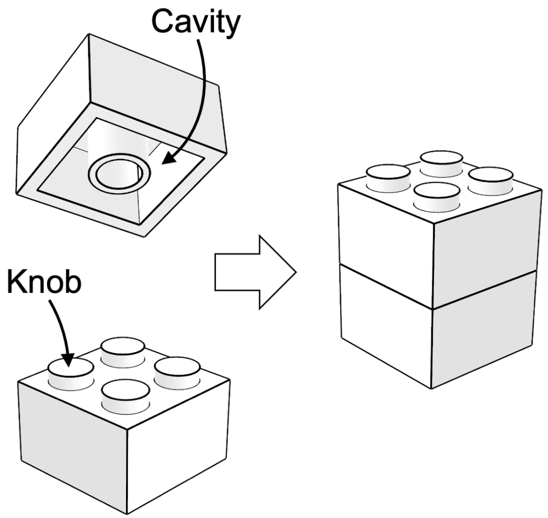
Unstable Input | Motion-based Representation | Stable Output

Friction

Friction

- Friction prevents the relative movement of adjacent parts if compression forces exist between them.
- Many assemblies that use snap joints need friction to stay stable.

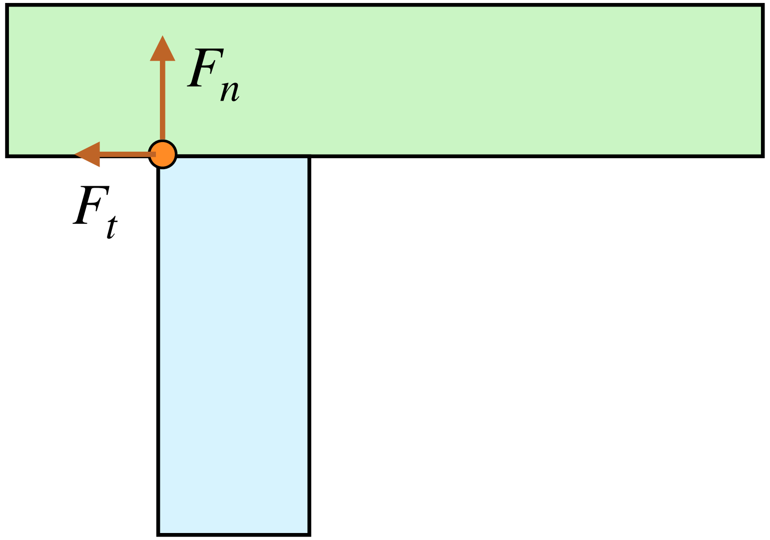




[Lego]

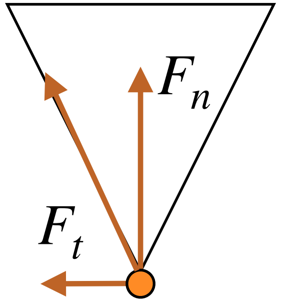
Coulomb Friction

- The widely used Coulomb friction model.
- The resultant force must be within the friction cone.



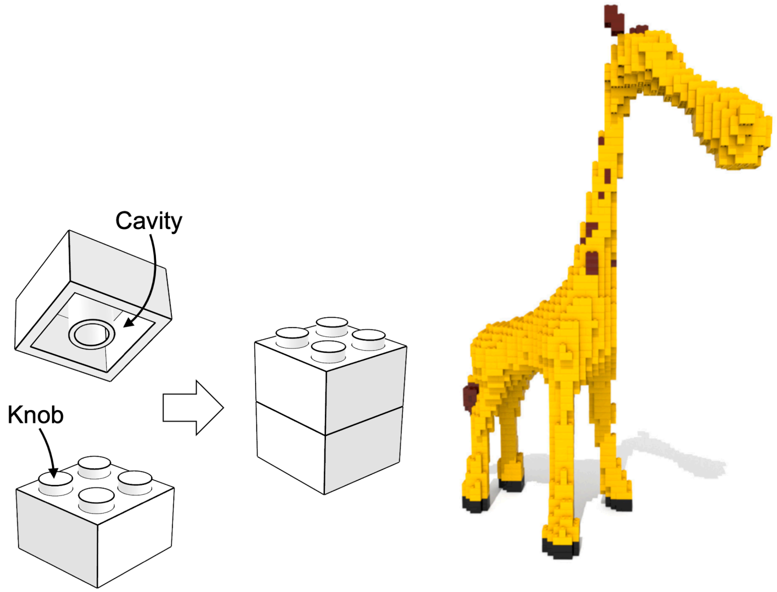
Friction Cone

$$F_t \leq \mu F_n$$

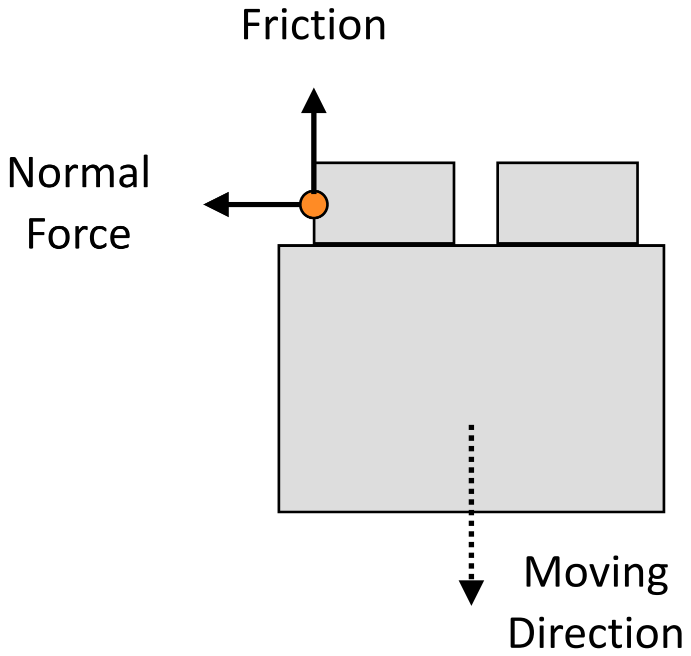


Friction for LEGOs

- For Legos, the normal forces are constant.
- The friction forces must be within a precomputed range.

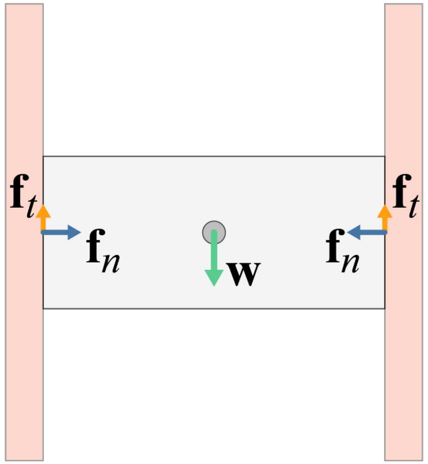


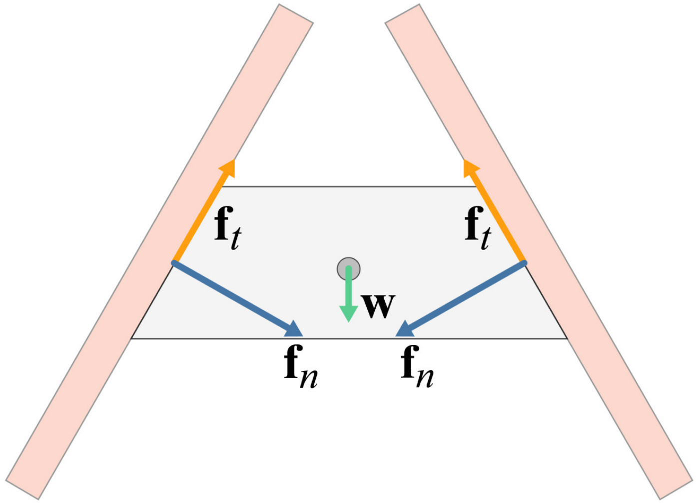
[Luo et al. 2015]



Limitations of Coulomb Friction

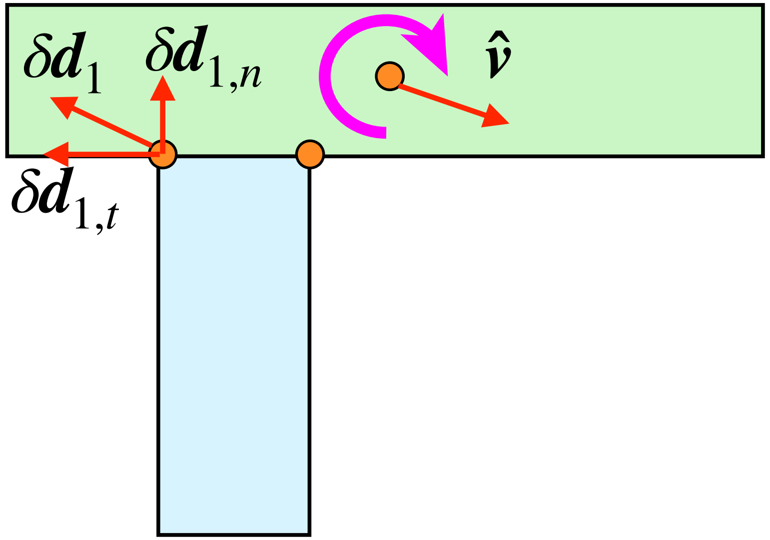
- The Coulomb friction may produce unrealistic force configurations.
- The most well-known failure case is the sliding issue.





Additional Physical Principles

- Adding more constraints to regulate the friction helps avoid unrealistic cases.



$$\delta \boldsymbol{d} = B_{\text{in}} \hat{\boldsymbol{v}}$$

Complementary Condition:

$$\delta d_{1,n} \cdot f_{1,n} = 0$$

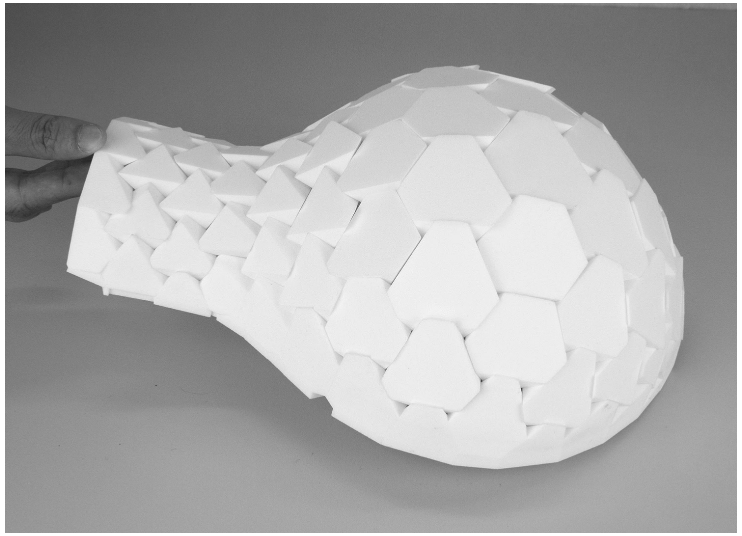
Maximum Dissipation

$$f_{1,t} = -\alpha_1 \delta d_{1,t}$$

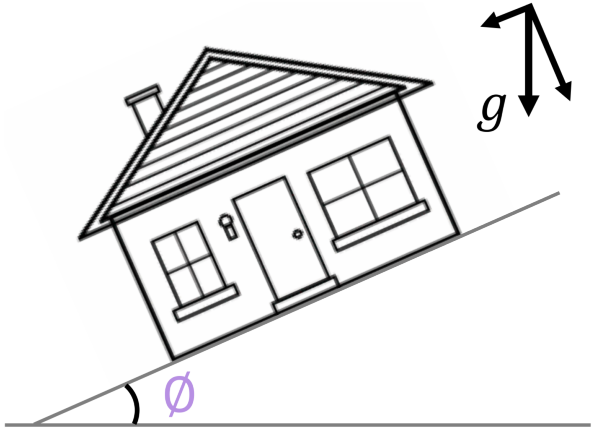
Lateral Stability

Lateral Stability

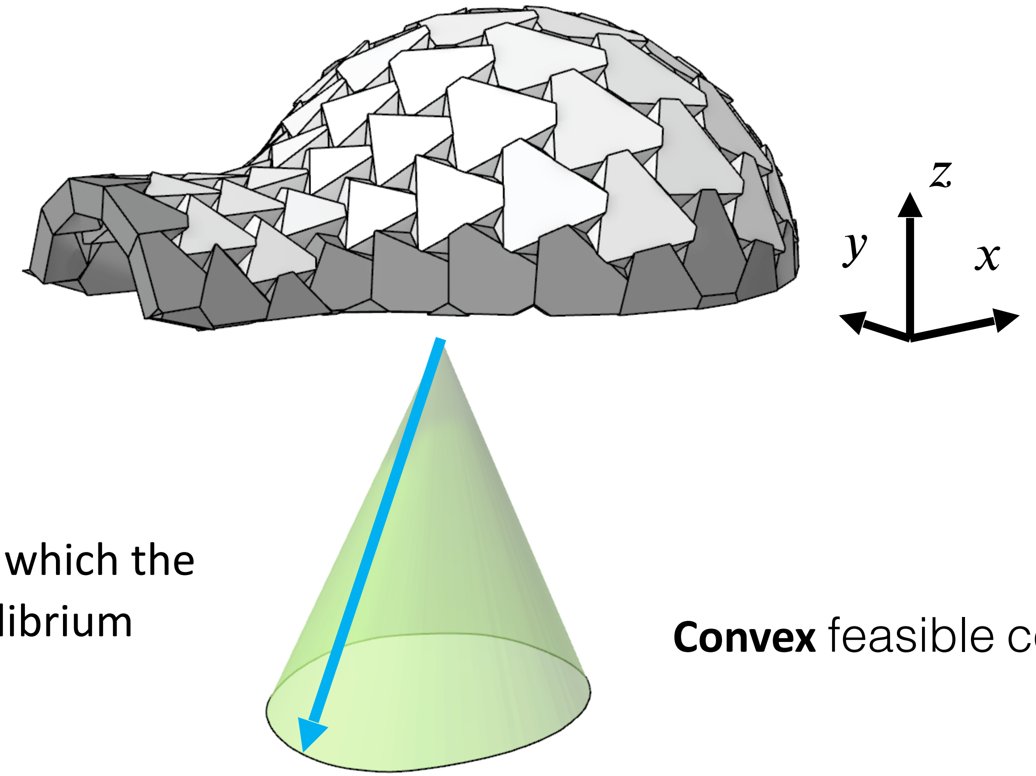
- Assemblies with later stability are in equilibrium for a cone of gravity direction.



[Wang et al. 2019]



Feasible Gravitational Cone

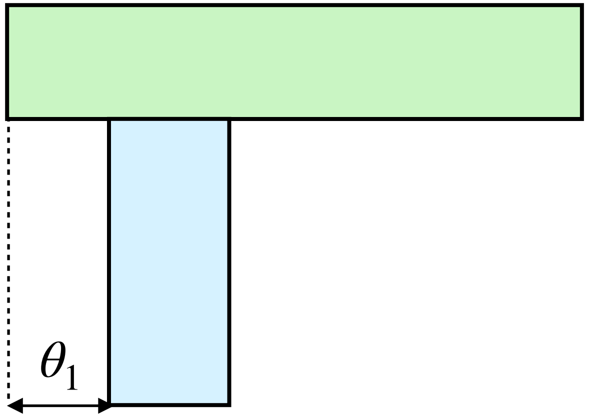


A gravity direction for which the structure is in equilibrium

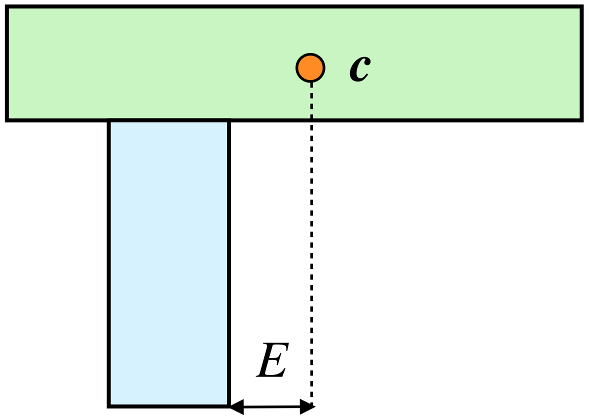
Convex feasible cone

Recap: Gradient-based Stability Optimization

- Come up with new infeasibility energy for lateral stability.



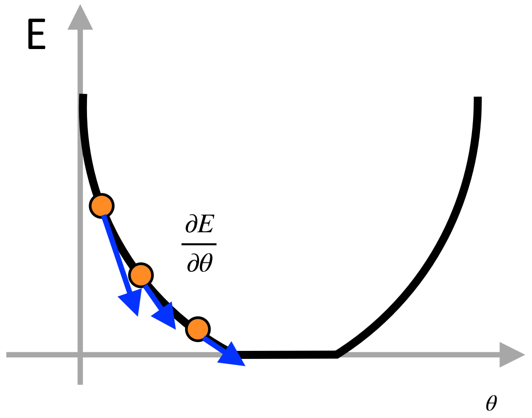
Step 1
Geometrical Property



Step 2
Infeasibility Energy

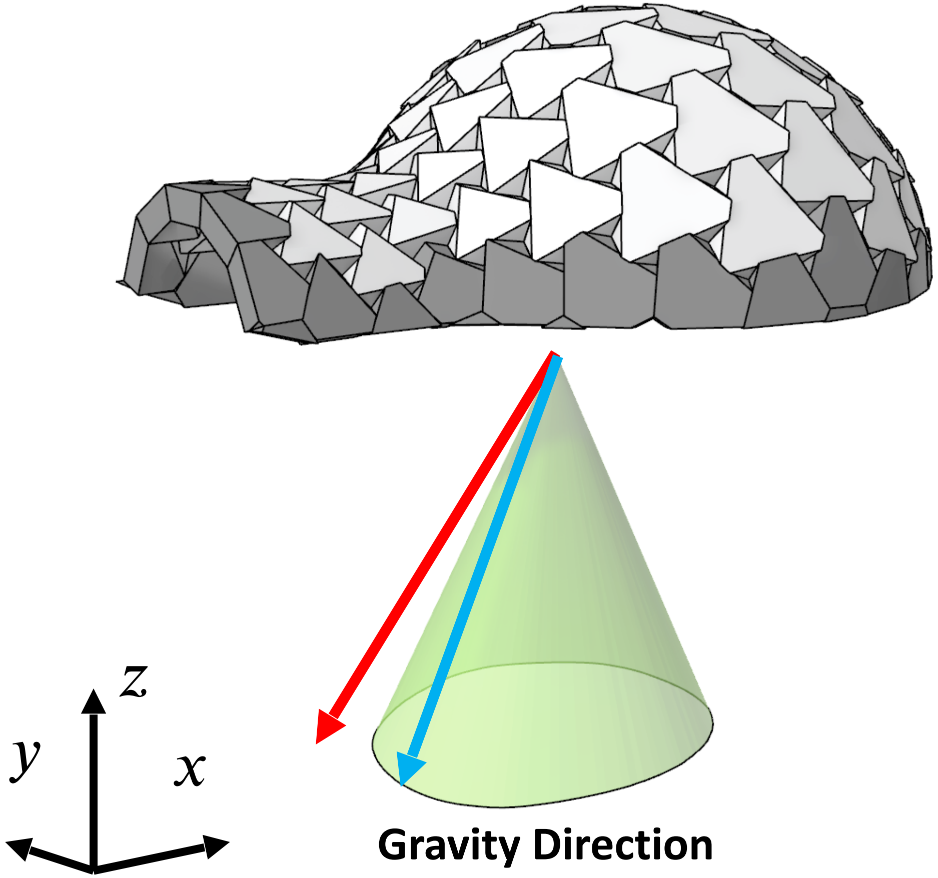
$$\boldsymbol{\theta} \xrightarrow{\partial} \boldsymbol{c} \xrightarrow{\partial} E$$

Step 3
Sensitivity Analysis



Step 4
Numerical Optimization

Lateral Infeasibility Measurement

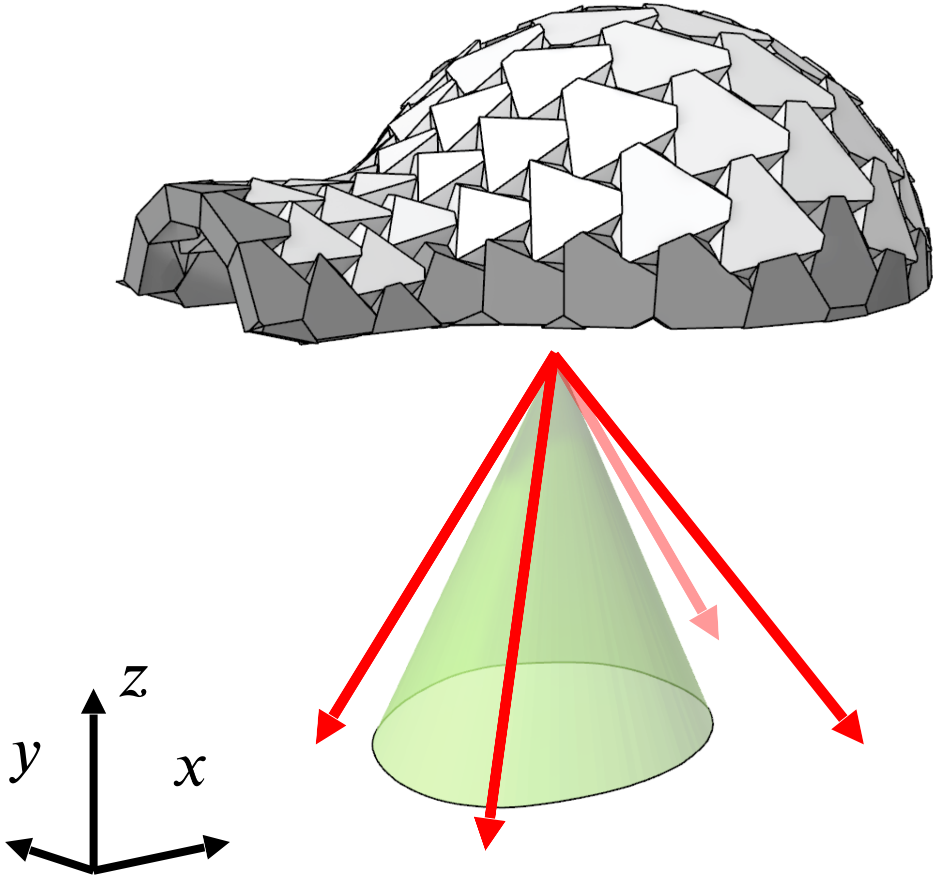


Structural Infeasibility

$$E(\swarrow) = 0$$

$$E(\downarrow) > 0$$

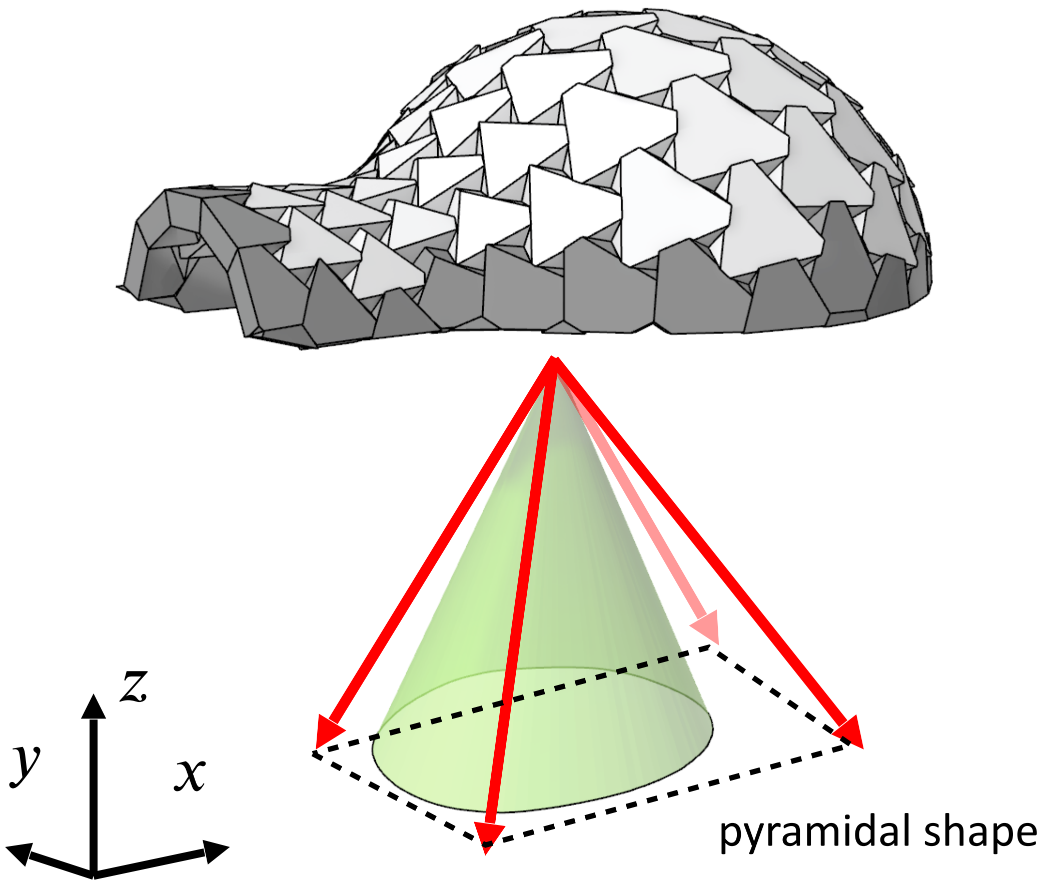
Lateral Infeasibility Measurement



min E(↙) + E(↓) + E(↓) + E(↘)

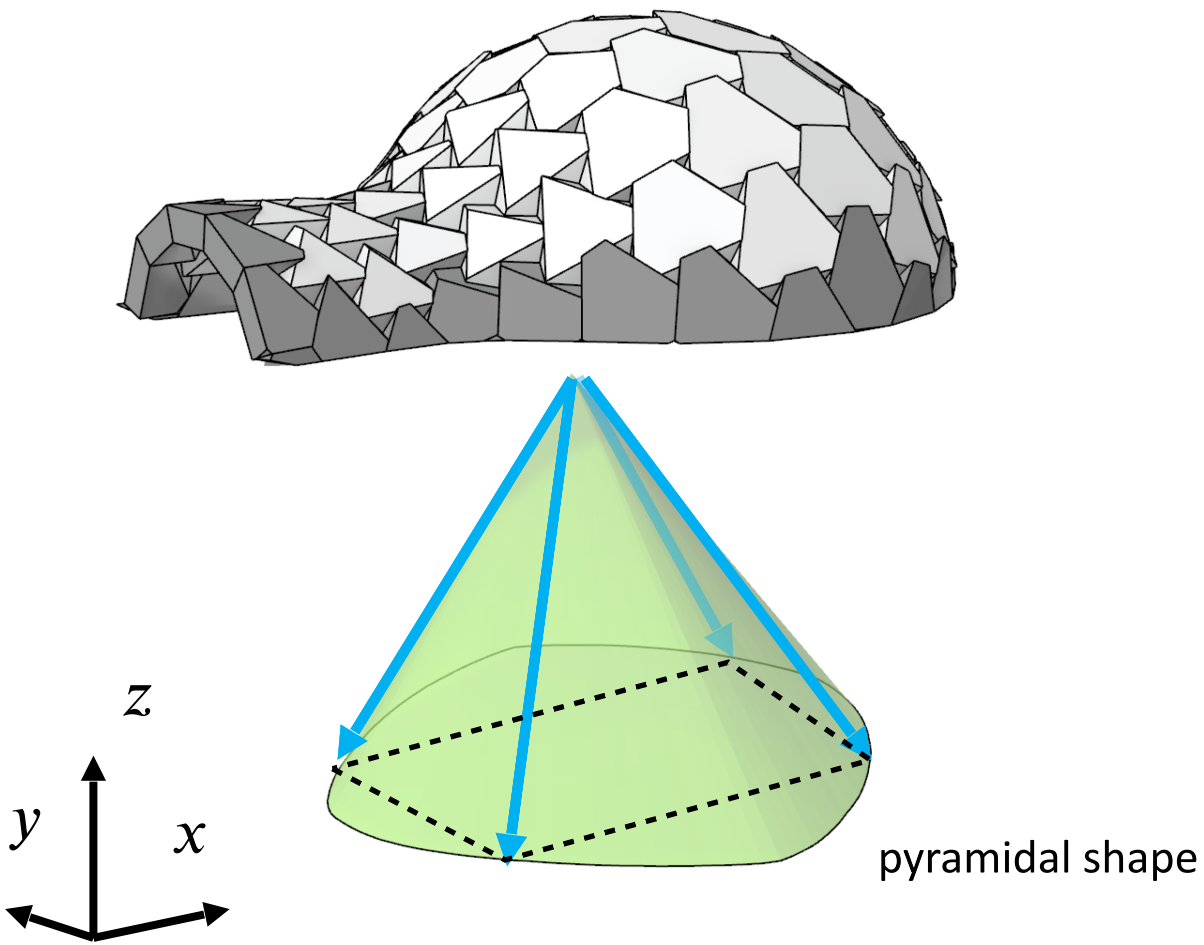
Contact Area ≥ User defined value

Lateral Infeasibility Measurement



Due to the convexity of the feasible cone

Lateral Infeasibility Measurement

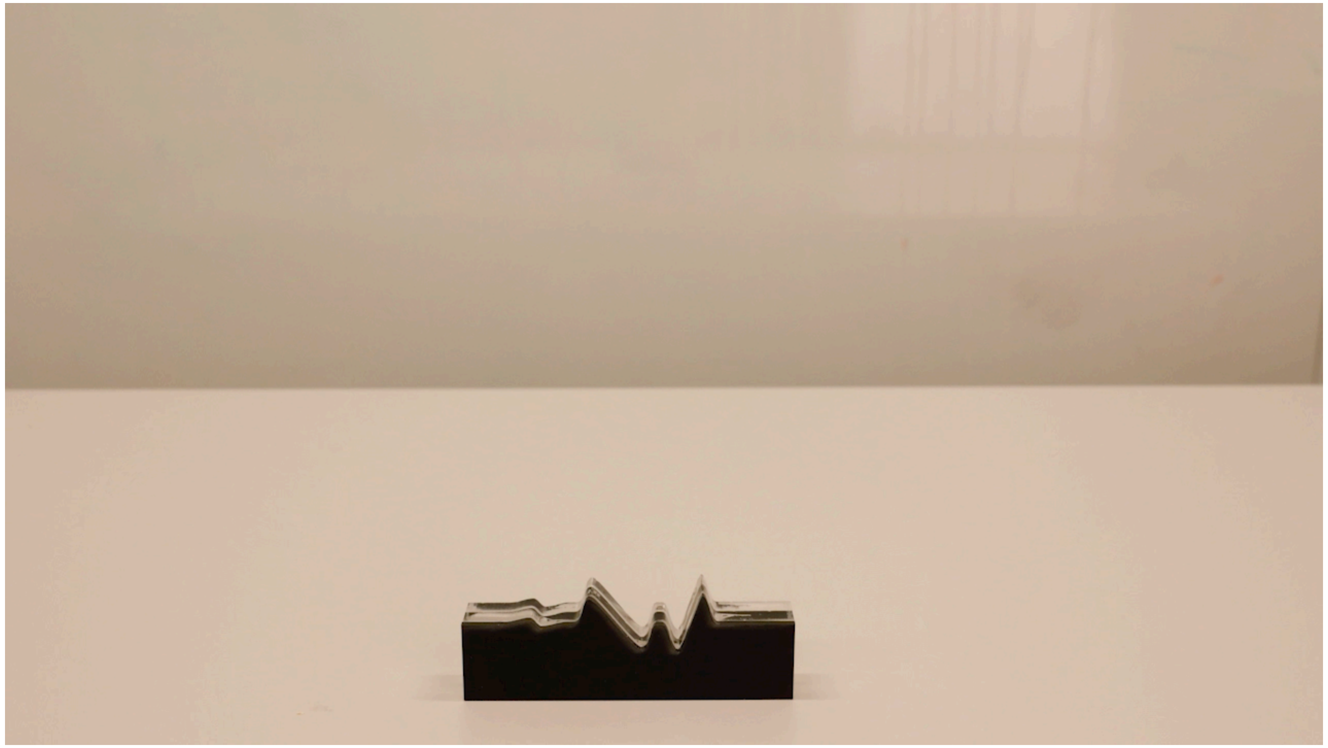


The new feasible cone
will cover the pyramidal shape

Scaffold-free Assembly

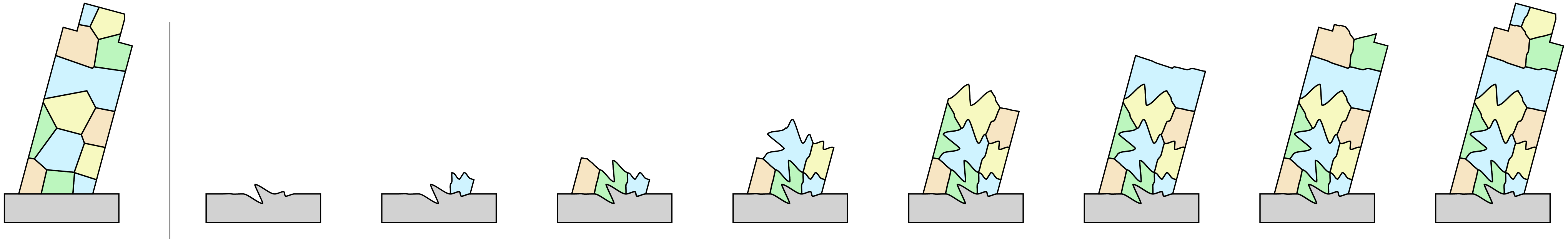
Scaffold-free Assembly

- Making the assembling process stable.



Infeasibility Energy

- The infeasibility energy is the summation of all the infeasibility energy of the structure at each assembling stage.

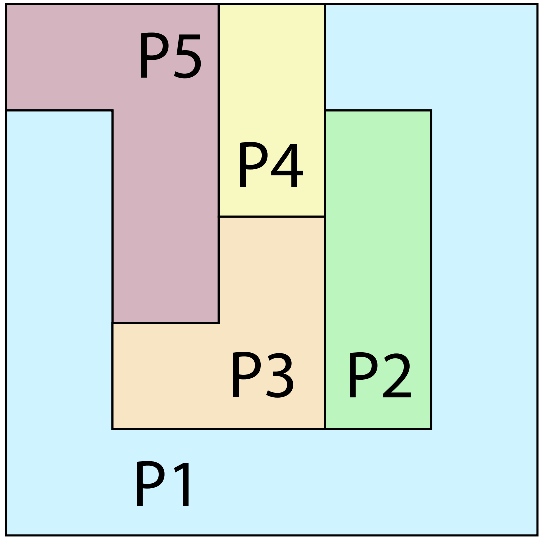


$$E = E_1 + E_2 + E_3 + E_4 + E_5 + E_6 + E_7 + E_8$$

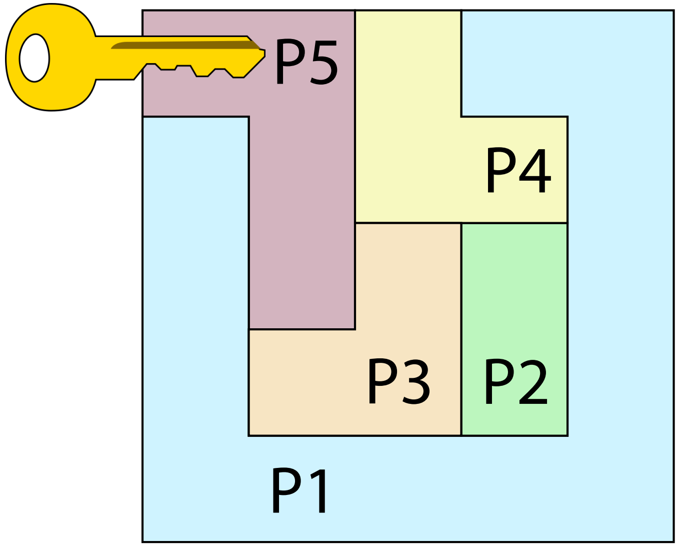
Globally Interlocking Assemblies

Recap: Globally Interlocking

Once the key and a part of the reset are fixed, no parts can be taken out from the assembly.



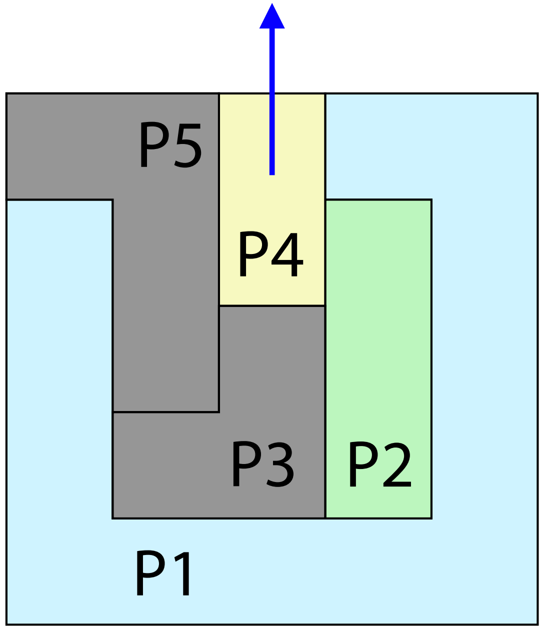
Non Interlocking



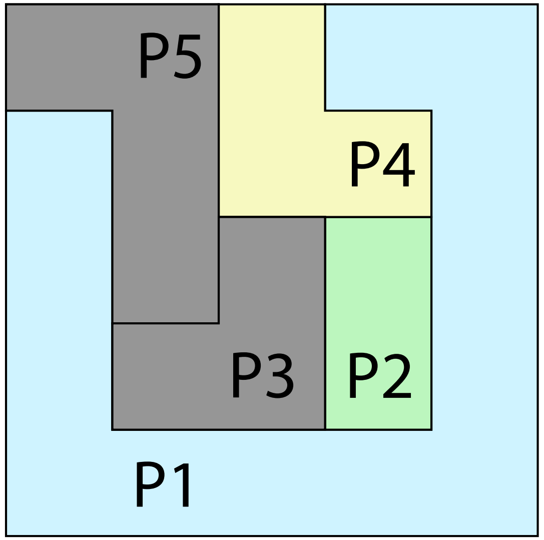
Interlocking

Recap: Globally Interlocking

Once the key and a part of the reset are fixed, no parts can be taken out from the assembly.



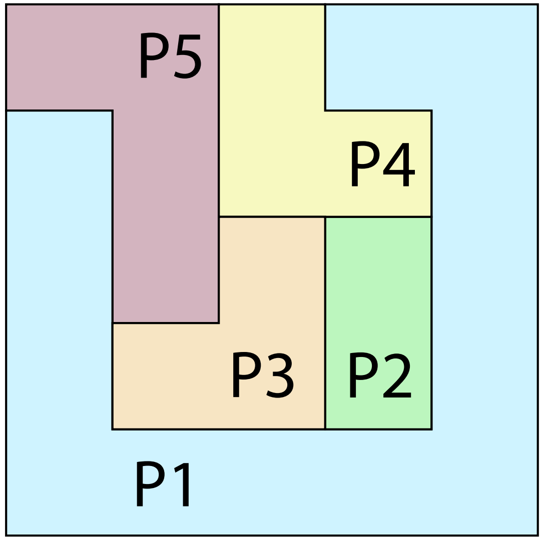
Non Interlocking

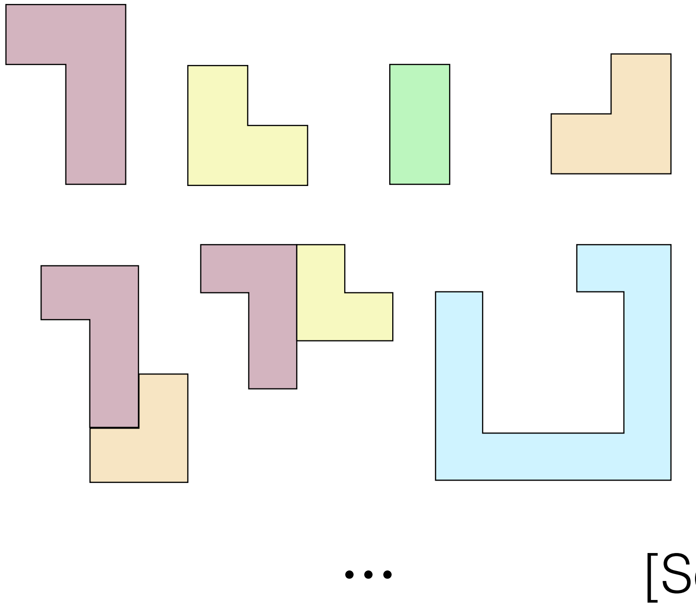


Interlocking

Recap: Classic Interlocking Test

Classic method examines every subset of parts, which has **exponential time complexity.**

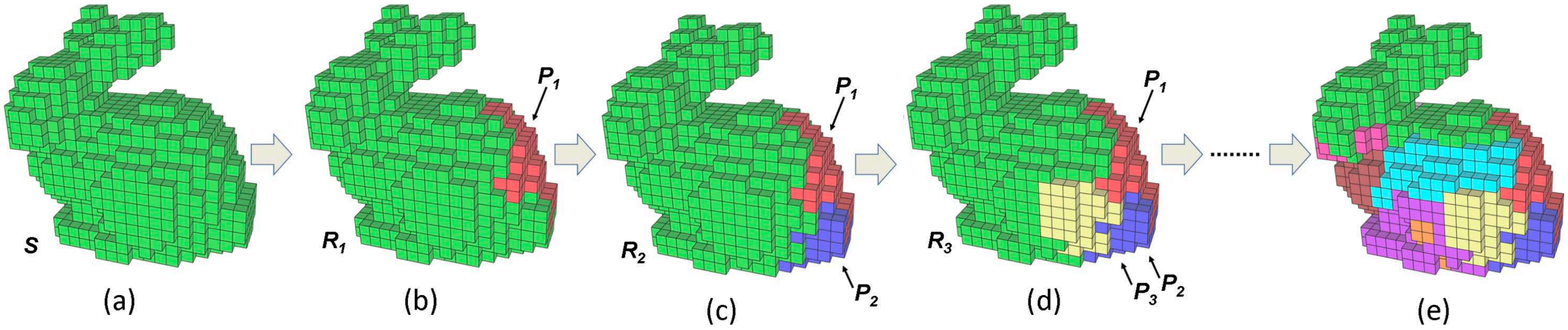




[Song et al. 2012]

Shape Decomposition

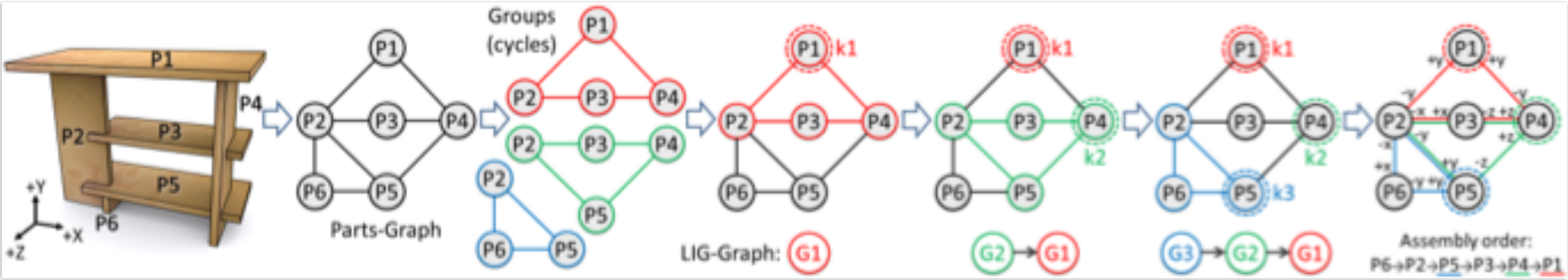
- When the input is a target shape, computational design of interlocking assemblies can be formulated as a shape decomposition problem.



[Song et al. 2012]

Joint Planning

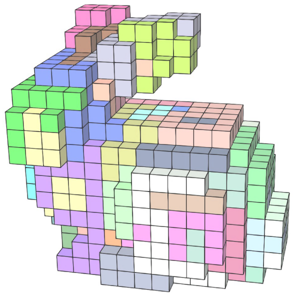
- Fu et al. computed an interlocking joint configuration following the LIG-based approach.



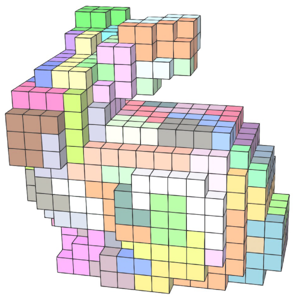
[Fu et al. 2015]

DBG-based Interlocking Design

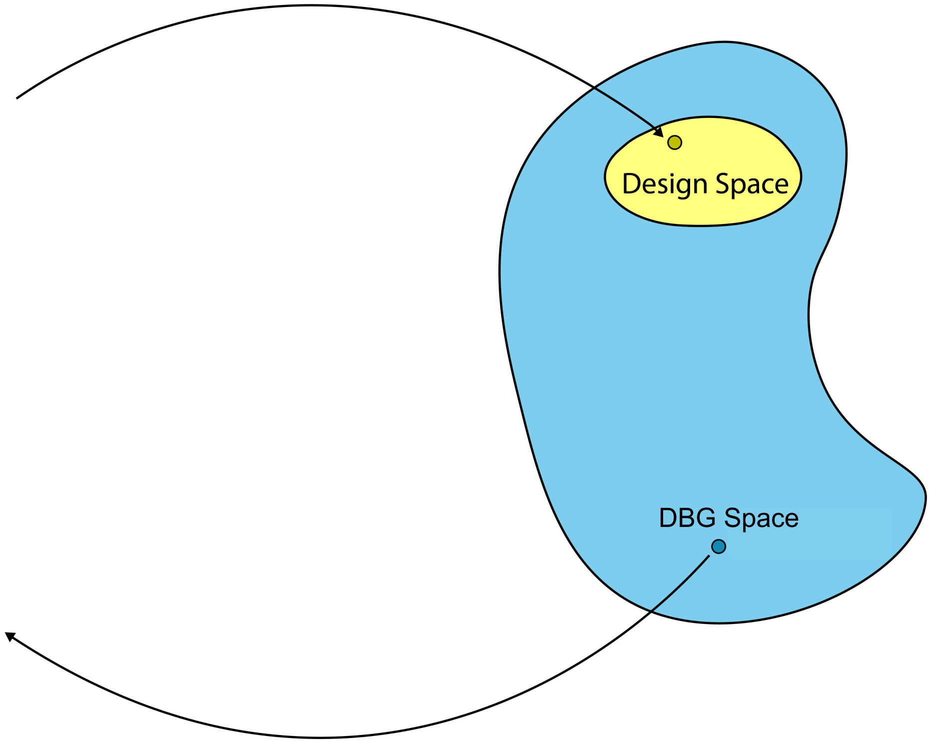
- The DBG approach allow exploring the full search space of interlocking configurations.



40-part Bunny
[Song et al. 2012]



80-part Bunny
[Wang et al. 2018]



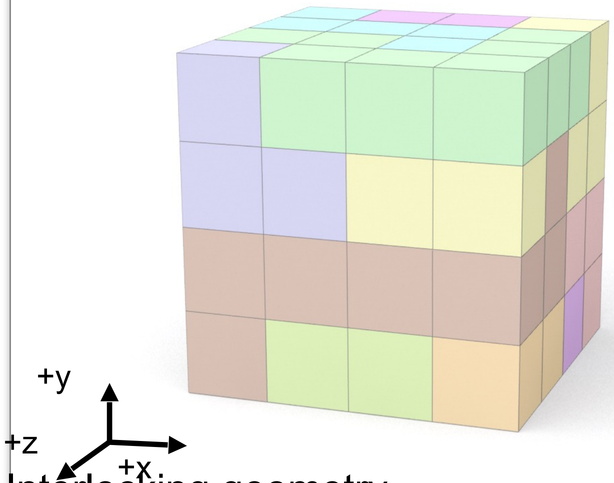
[Wang et al. 2018]

Ziqi Wang Peng Song Mark Pauly

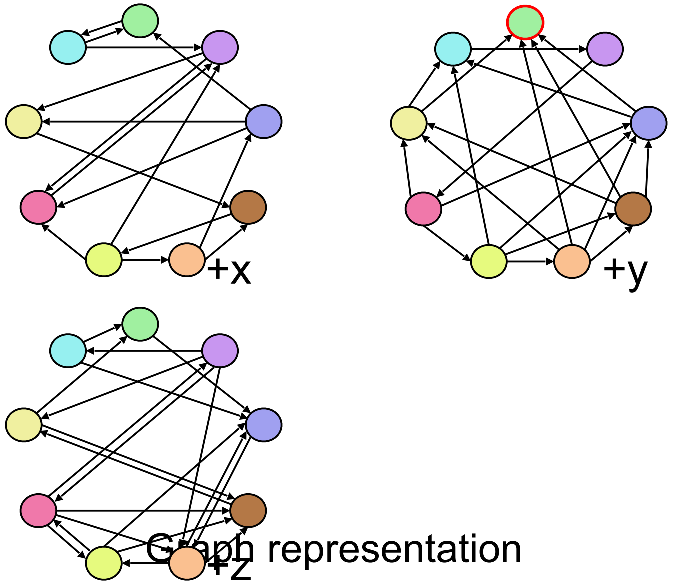
DBG-based Interlocking Design

- Wang et al. use the base DBG to test and design interlocking assemblies.

The 3D assembly is interlocking = All graphs are strongly connected
(note: ignore the key when check connectivity)



Interlocking geometry



Graph representation

et al. 2018]