DynaShape

Designing Complex Forms in Dynamo

Long Nguyen

Researcher & Lecturer, ICD University of Stuttgart

Las Vegas 2018

AUTODESK. UNIVERSITY

About the speaker

Long Nguyen

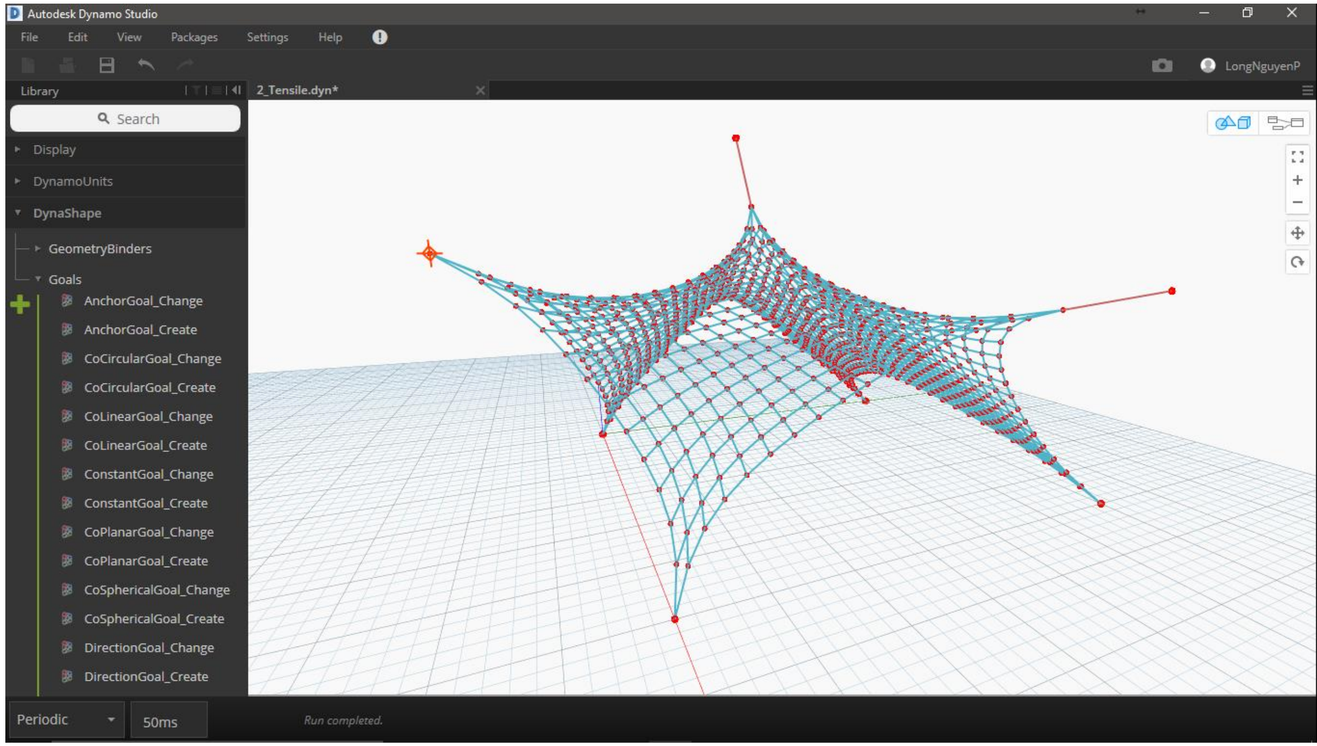
- Computer Scientist
- Special interests in
  - Computer Graphics
  - Architecture
  - BIM
  - Computational Design
  - Generative Design (whatever it currently means)
  - Digital Photography
  - Lightsabers
- Main languages: Vietnamese, English, C#, C++, Python, Java & JavaScript
- Researcher & lecturer in computational @ ICD University of Stuttgart
- Love teaching programming and algorithms for computational design

 *@LongNguyenP*

 *www.linkedin.com/in/LongNguyenP*

DynaShape

- Form-finding by computing the equilibrium of the physical forces driving the design geometries
- Form-finding based on solving multiple, often conflicting, geometric constraints
- Optimizing existing designs



Curvature Before Dynashape

Display.BySurfaceColors

surface

colors

Display

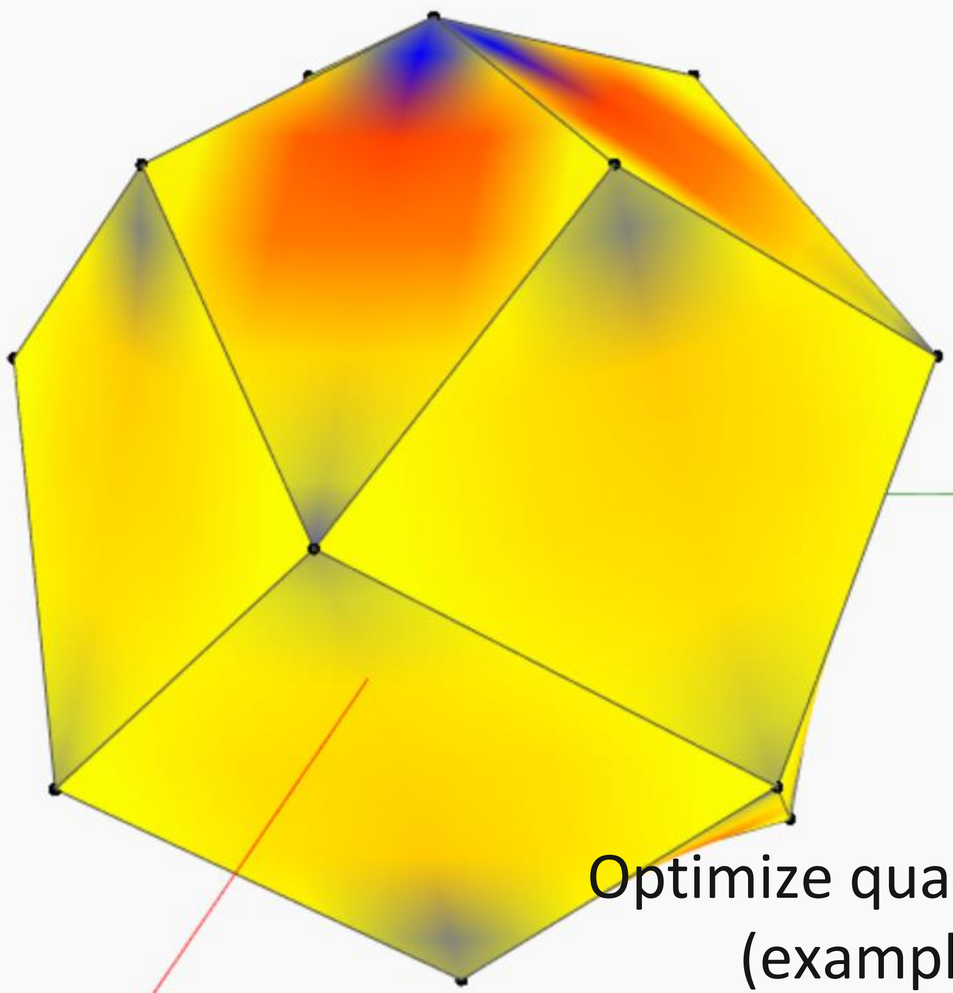
Curvature After Dynashape

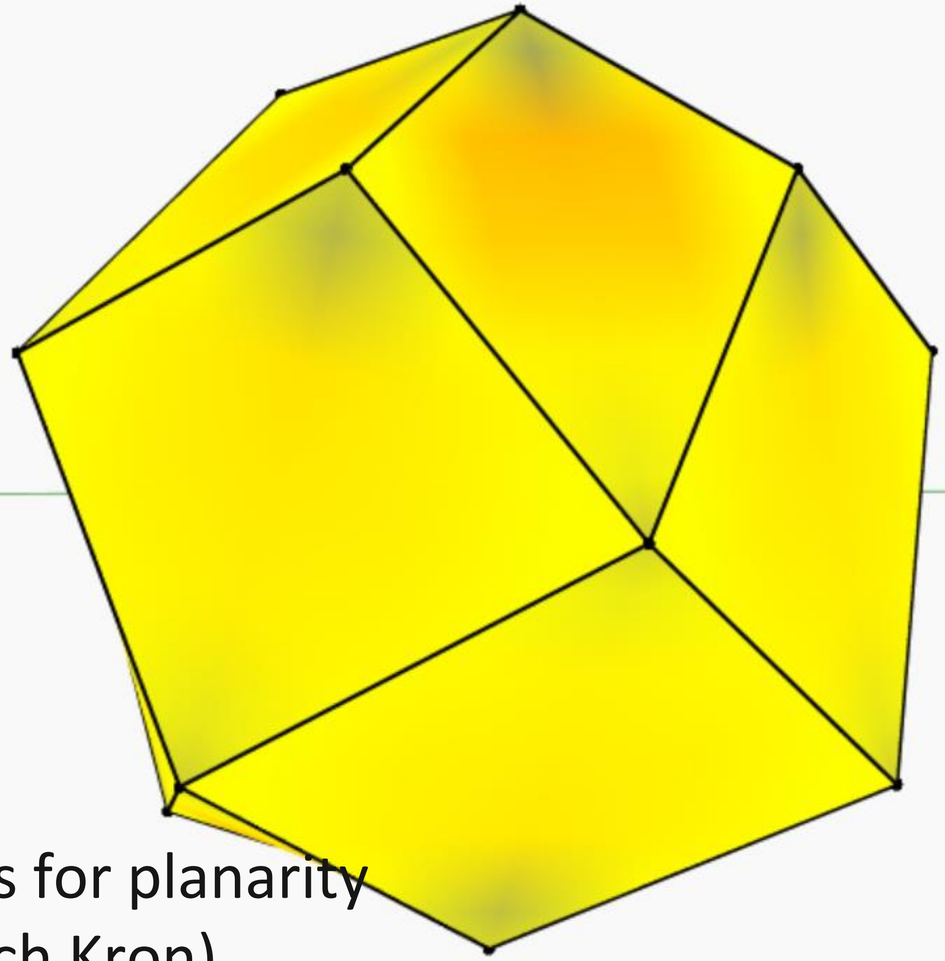
Display.BySurfaceColors

surface

colors

Display

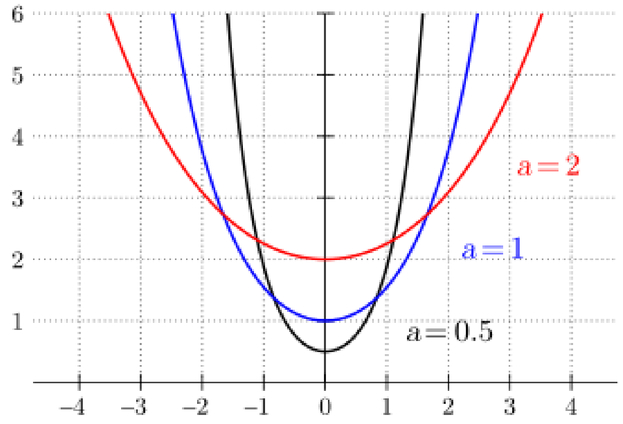




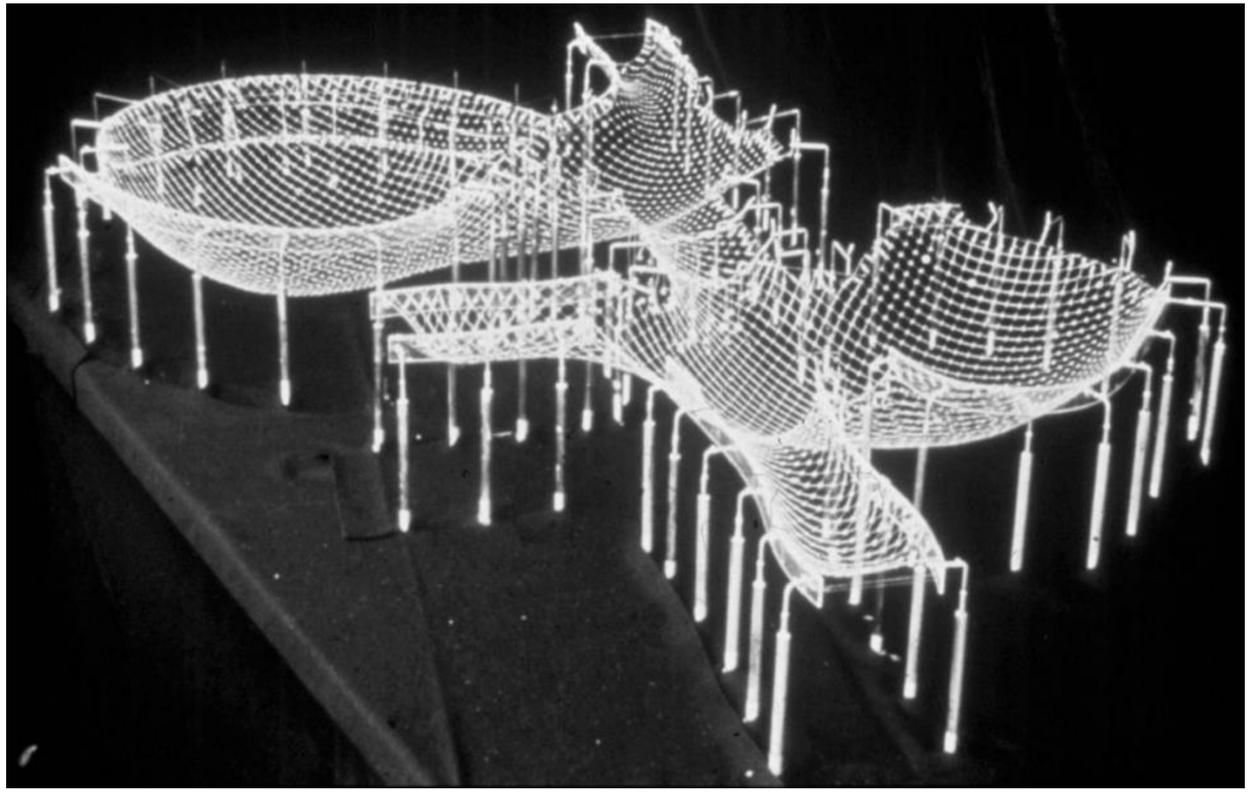
Optimize quad panels for planarity
(example by Zach Kron)

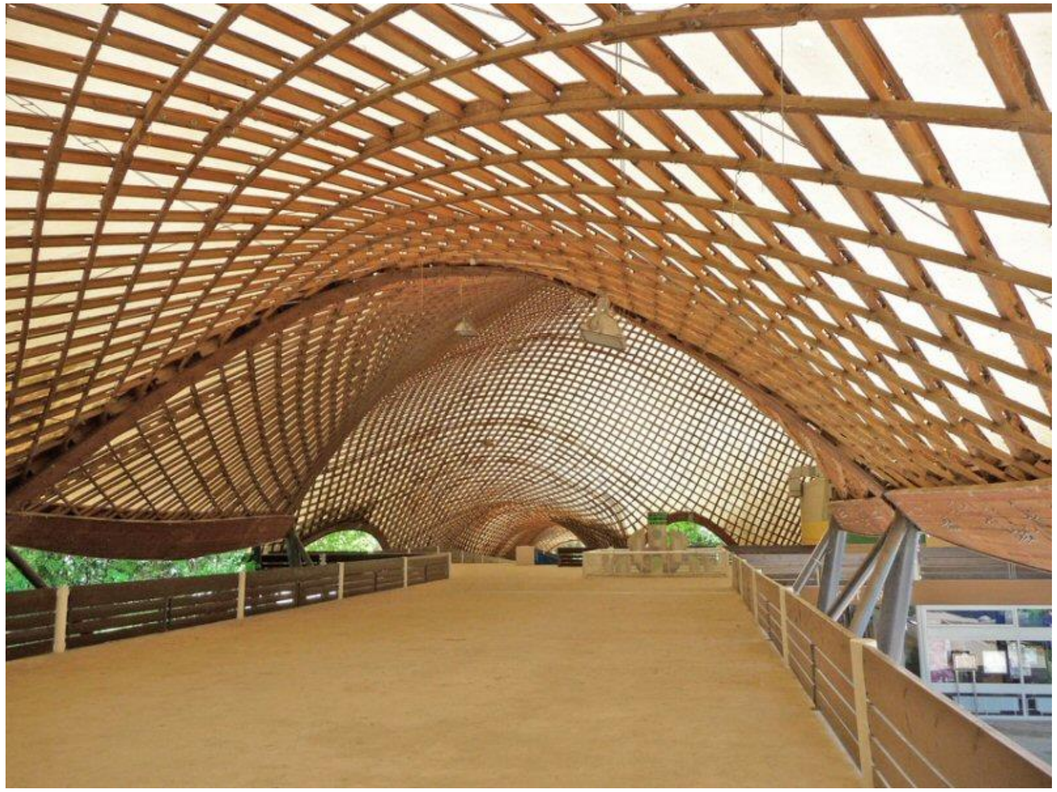
Hanging Chains

An elegant way to generate compression-only form (e.g. arches)

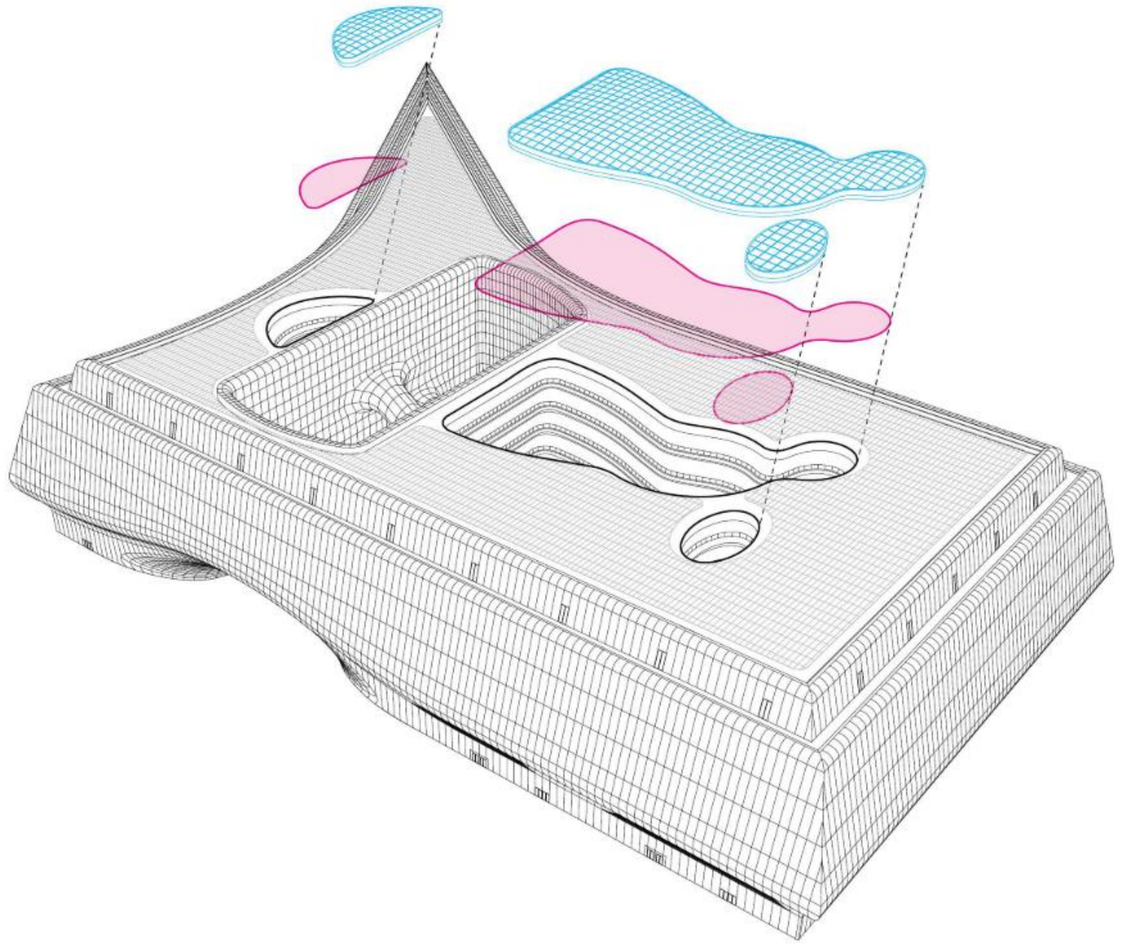


Hanging Chains





Mannheim Multihalle, Frei Otto



Skylight Roof of Sberbank (Russia), Zaha Hadid Architects

For more information, check out Eckart Schwerdtfeger’s AU2018 class “Custom Computational Workflows for BIM Design Implementation”. Recording will be available.

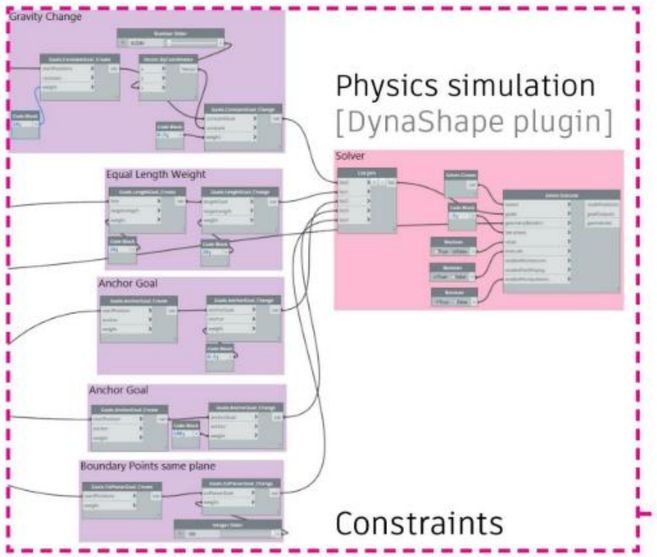
Dynamo . Skylight

Base geometry from Rhino

Geometry transfer via cloud service

Visual scripting
Physics simulation

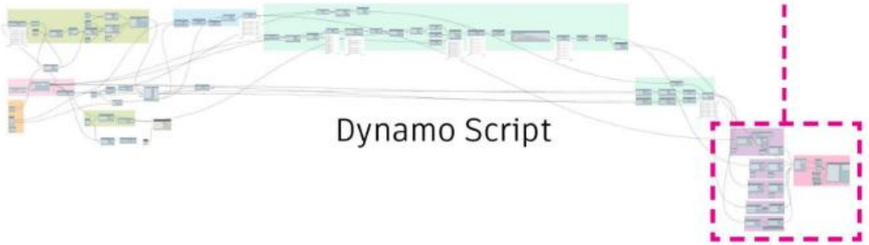
Simulated form
[DynaShape]



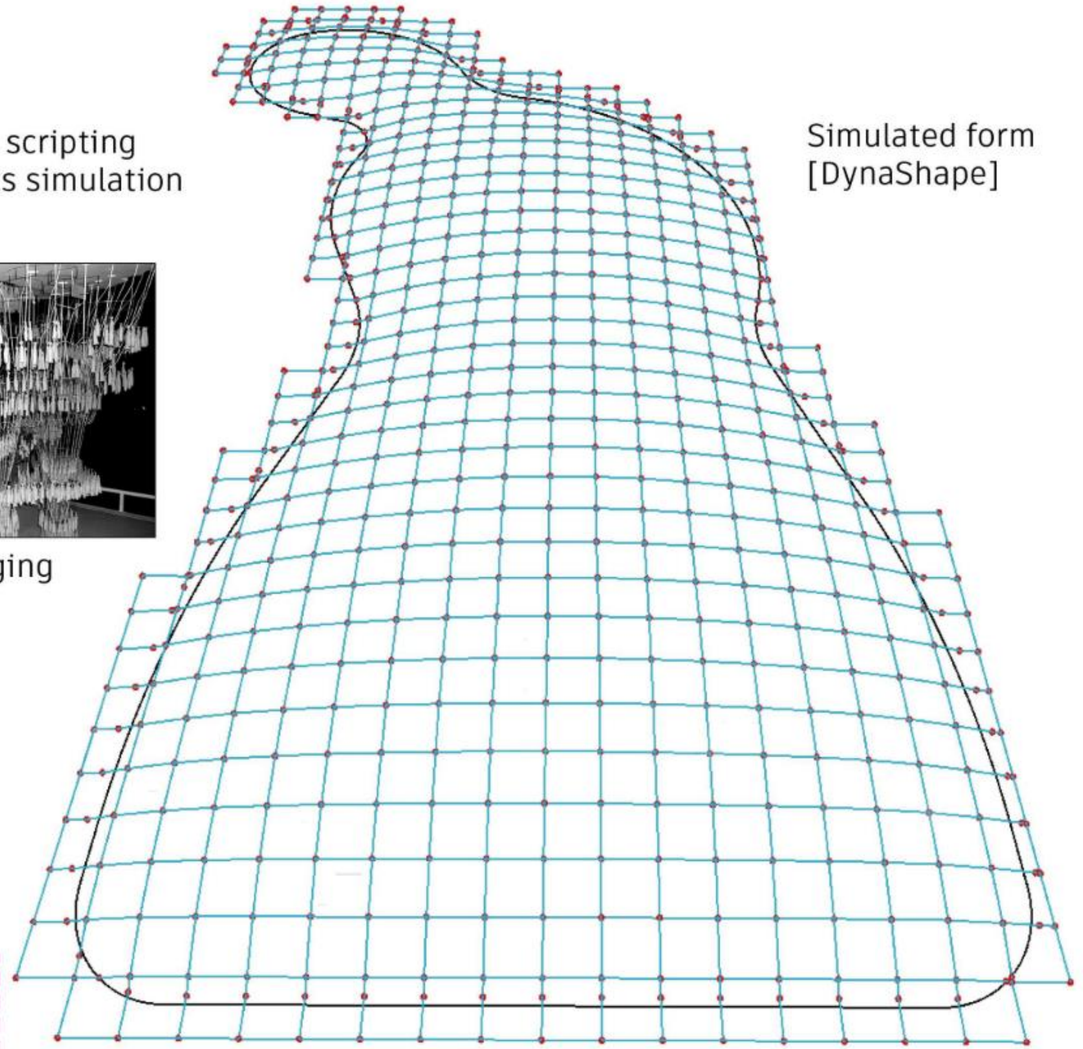
Physics simulation
[DynaShape plugin]

Constraints

Gaudi . Hanging
chain model



Dynamo Script



Dynamo script by Mauro Sabiu
DynaShape plugin by Long Nguyen

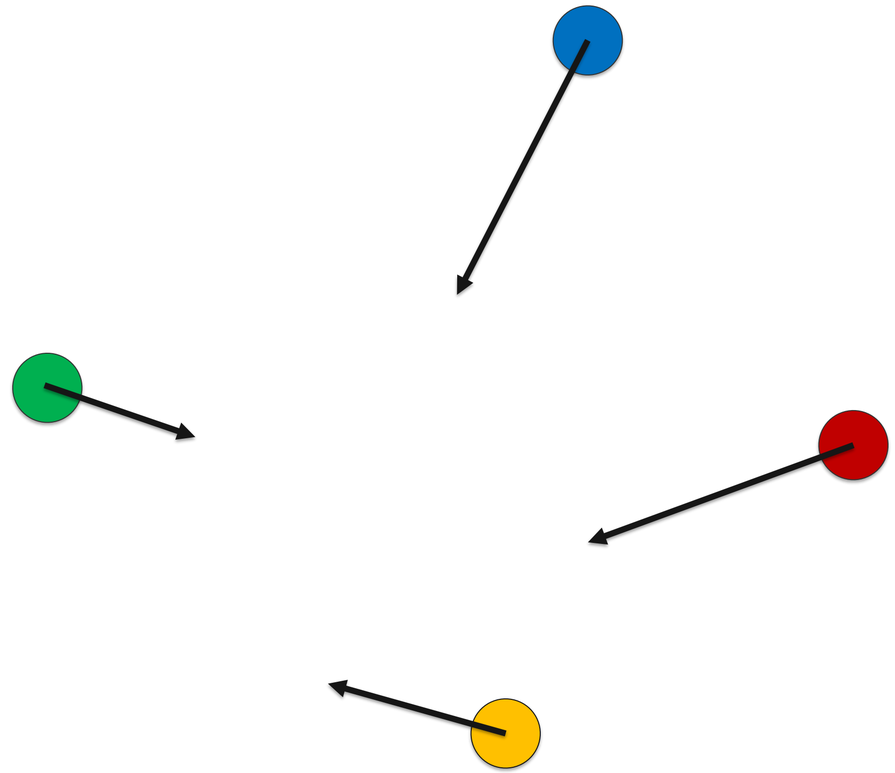
Eckart Schwerdtfeger
AU Las Vegas 2018

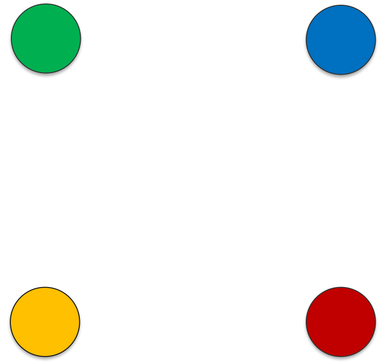
Zaha Hadid Architects

Skylight Roof of Sberbank (Russia), Zaha Hadid Architects

For more information, check out Eckart Schwerdtfeger’s AU2018 class “Custom Computational Workflows for BIM Design Implementation”. Recording will be available.

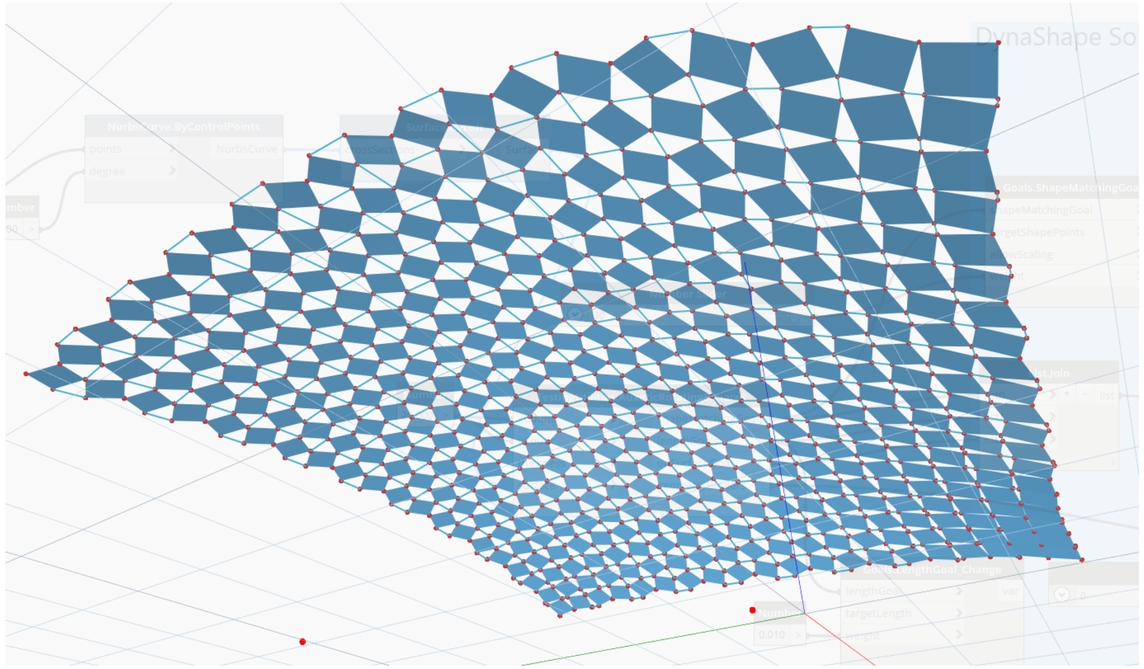
Shape Matching constraints

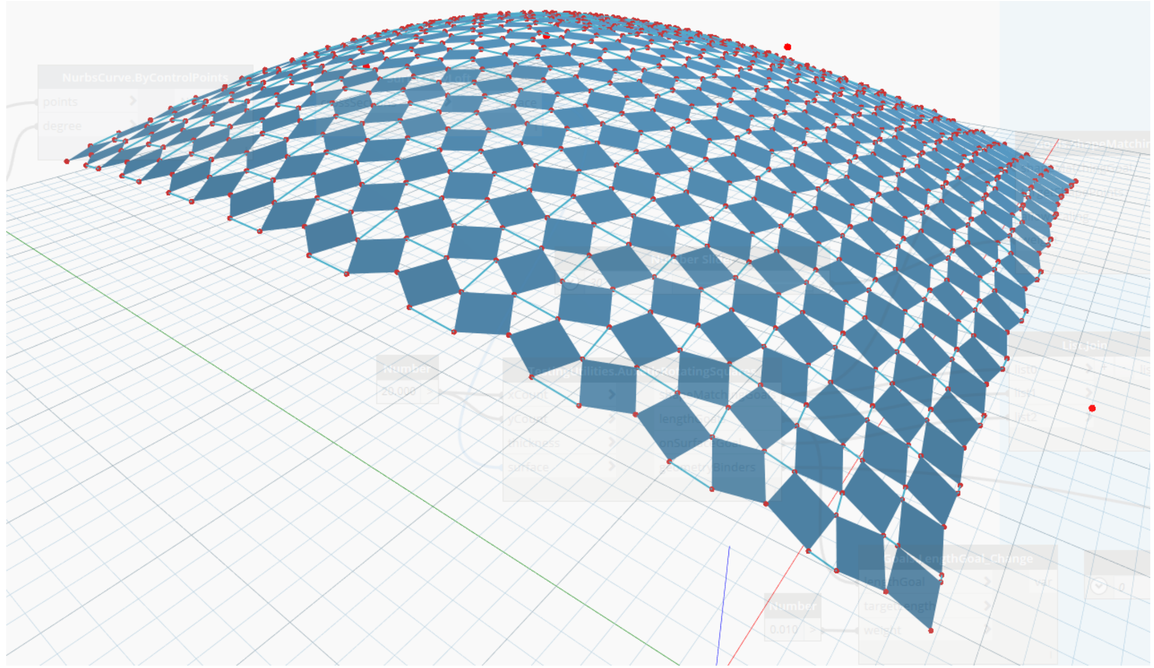




Target Shape
(e.g. a square)

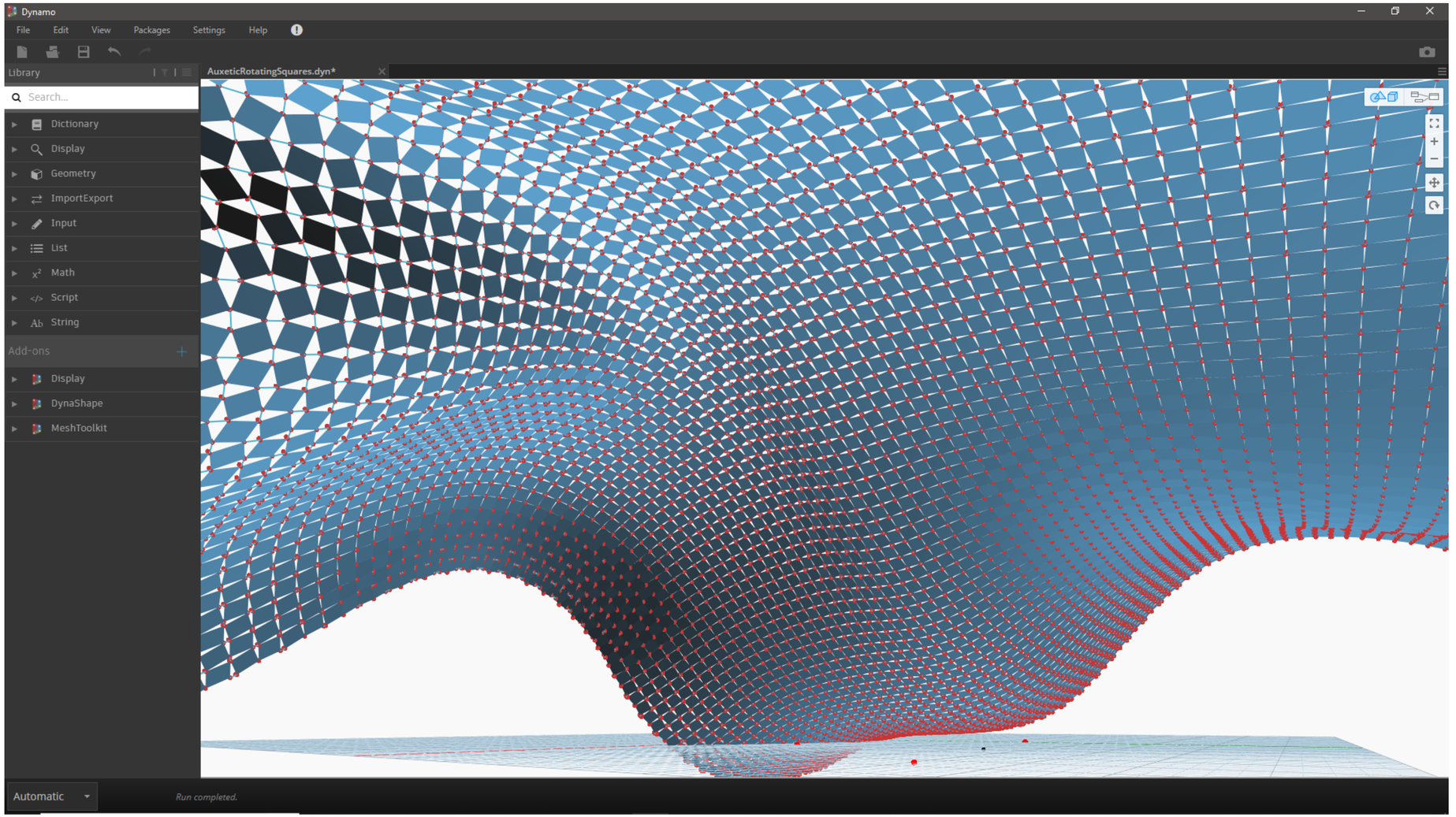
Shaping matching constraints





(Partially) covering a doubly-curved surface with identically-shaped elements

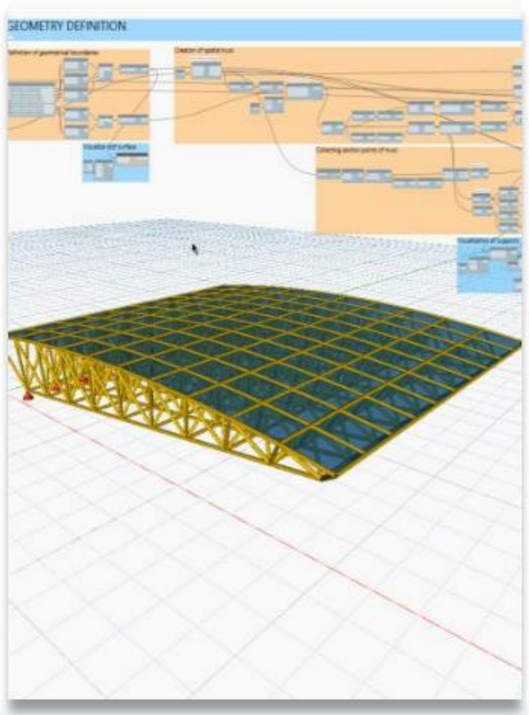
Shaping matching constraints



(Partially) covering a doubly-curved surface with identically-shaped elements

DynaShape and Generative Design!!!

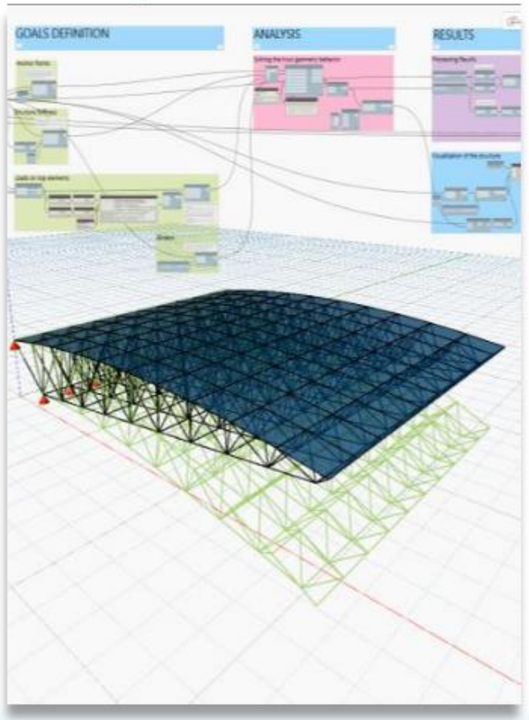
DYNAMO



Geometry Definition

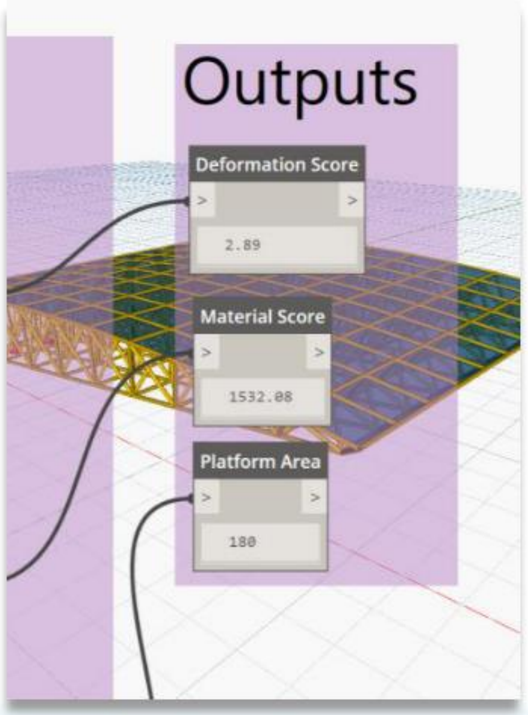
DYNAMO

DynaShape



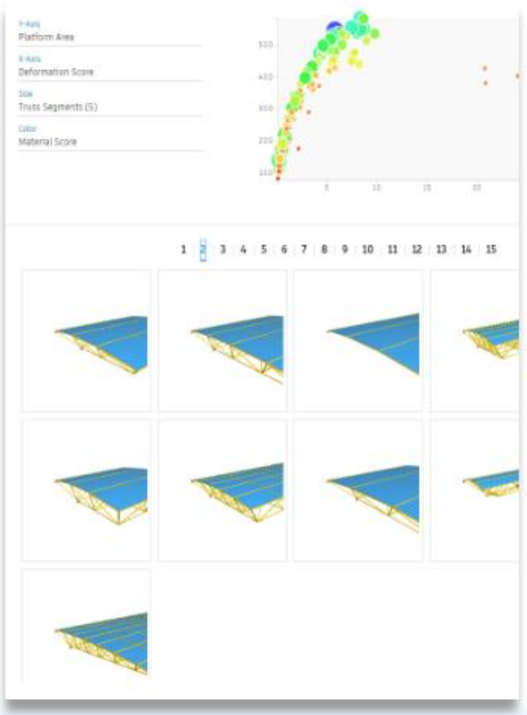
Deformation Analysis

DYNAMO



Results Evaluation

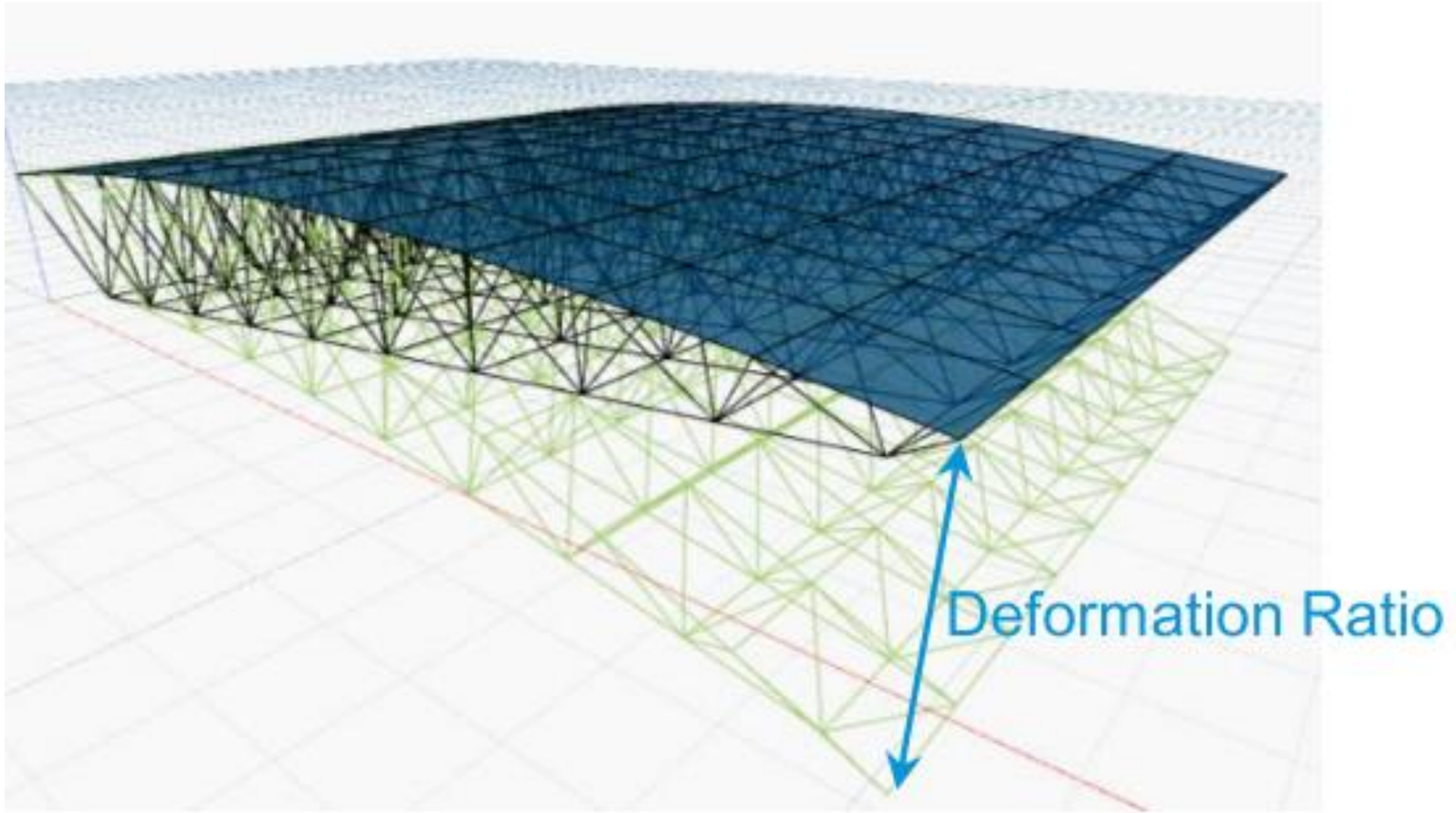
DYNAMO



Optimization in Refinery

For more information, Check out Dieter Vermeulen's AU2018 class "Structural Dynam(o)ite: Optimized Design & Fabrication Workflows with Dynamo". Recording will be available.

DynaShape and Generative Design!!!



For more information, Check out Dieter Vermeulen's AU2018 class "Structural Dynam(o)ite: Optimized Design & Fabrication Workflows with Dynamo". Recording will be available.

“Behind the scene”

Package installer and example sets and Q&A on DynamoBIM forum

Completely free and open source

Extensible: by defining custom goals and constraints (using C#)

The core theoretical framework is based on projection-based constraint solver and projective dynamic

- Projective Dynamics: Fusing Constraint Projections for Fast Simulation
- ShapeOp: a robust and extensible geometric modelling paradigm

Implementation based on weighted vector averaging (similar to KangarooPhyiscs). Easier for YOU to define custom goals & constraints

(Many thanks to Daniel Piker for very insightful discussions on usage concepts and many important implementation details)

AUTODESK®

Make anything™