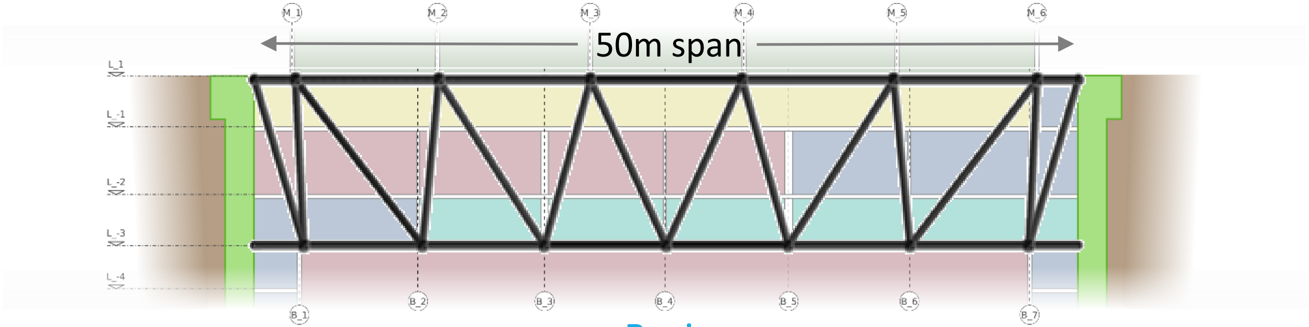
Layout optimization of simplified trusses

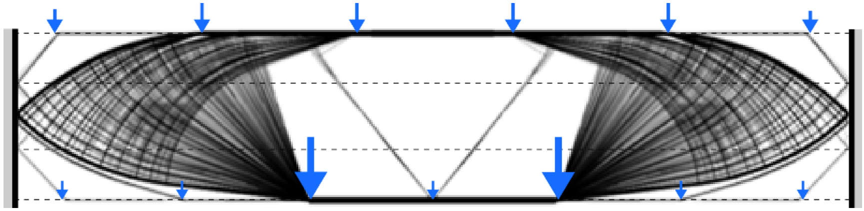
HELEN FAIRCLOUGH, MATTHEW GILBERT

UNIVERSITY OF SHEFFIELD / LIMITSTATE LTD.

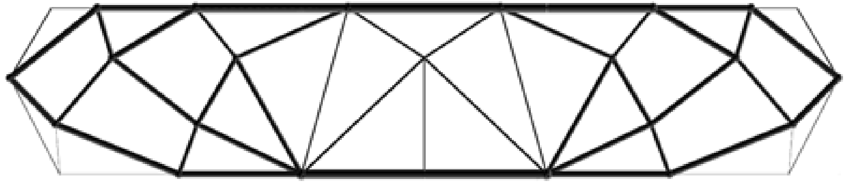
Why do we need simplified trusses?



Basic



Optimized (-56%)

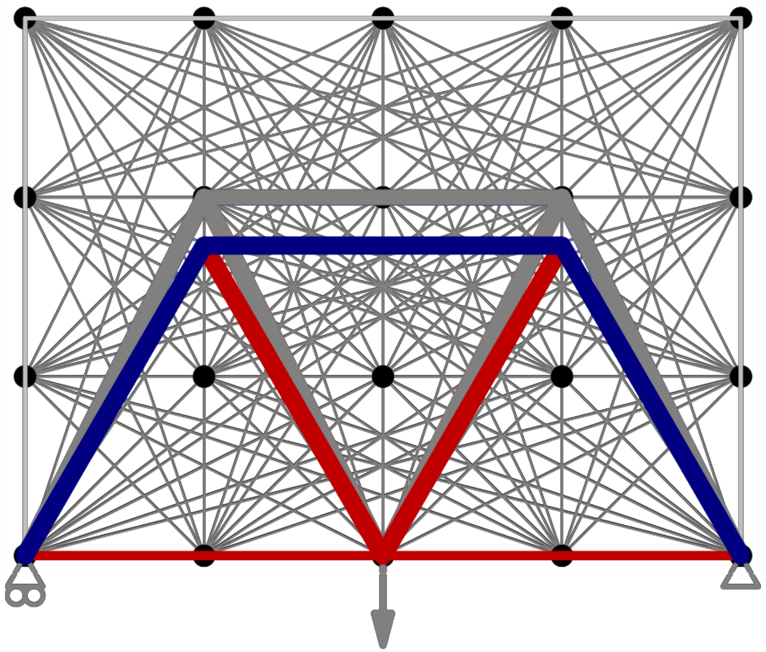


Optimized & simplified (-51%)

Fairclough et al, The Structural Engineer, 2019

Layout Optimization & Geometry Optimization

≈ Ground Structure Method (GSM)
≈ Truss topology optimization



Educational Python script available:
He et al, SMO, 2019

min	$V = \boldsymbol{l}^{\mathrm{T}}\boldsymbol{a}$	minimising volume
subject to	$\boldsymbol{B}\boldsymbol{q} = \boldsymbol{f}$	equilibrium
	$\lvert\boldsymbol{q}\rvert - \sigma\boldsymbol{a} < \boldsymbol{0}$	limiting stress

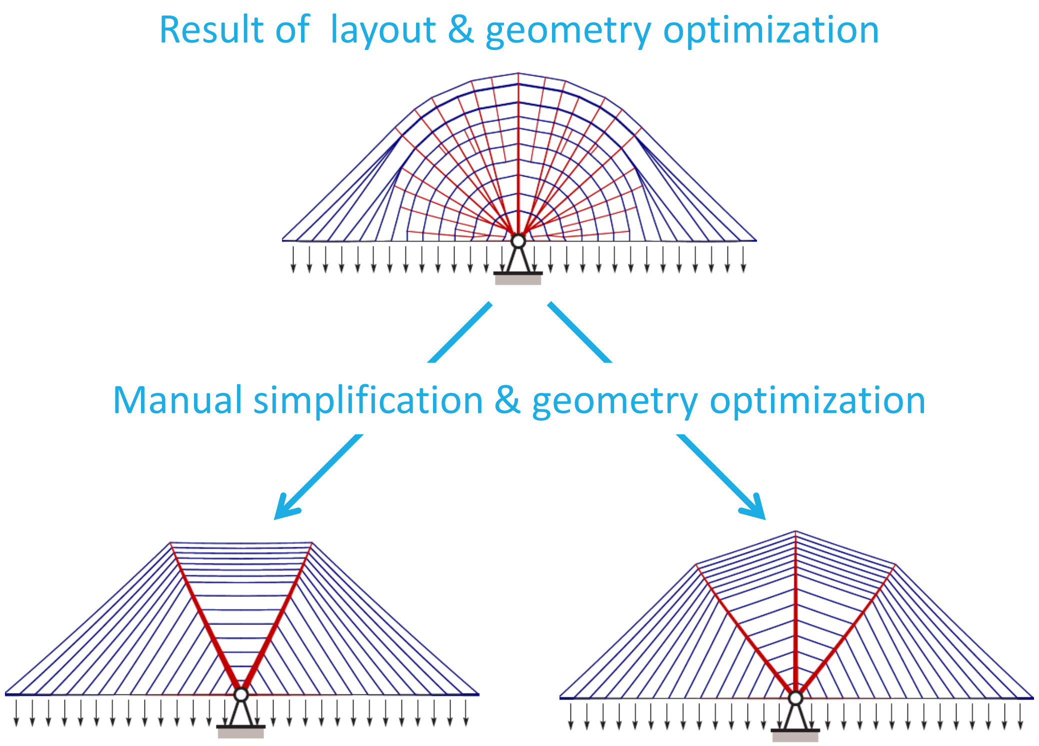
Force variable   Area variable

- Linear programming – very fast and globally optimal

- Geometry optimization - adds node positions as variables. Non-convex, but uses layout optimization as starting point

Manual simplification

- Can identify structures that are reasonably simple **and** have low volume
- Geometry optimization can be used to improve a manually interpreted solution
- But, time consuming, and not always easy



Fairclough et al, Proc. R. Soc. A, 2018

Automatic simplification approaches

Approach 1

Adding complexity constraints from the outset

Rigorous, but often very slow

Approach 2

Automatically post-processing an existing solution

Fast, but results not rigorous

New complexity constraints

$$\text{min} \qquad V = \boldsymbol{l}^{\mathrm{T}}\boldsymbol{a}$$ minimising volume

$$\text{subject to} \qquad \boldsymbol{Bq} = \boldsymbol{f}$$ equilibrium

$$|\boldsymbol{q}| - \sigma\boldsymbol{a} < \boldsymbol{0}$$ limiting stress

$$\widehat{M}v_j - \sum_{i \in J_j} a_i \geq 0$$

$$v_j \in \{0,1\}$$

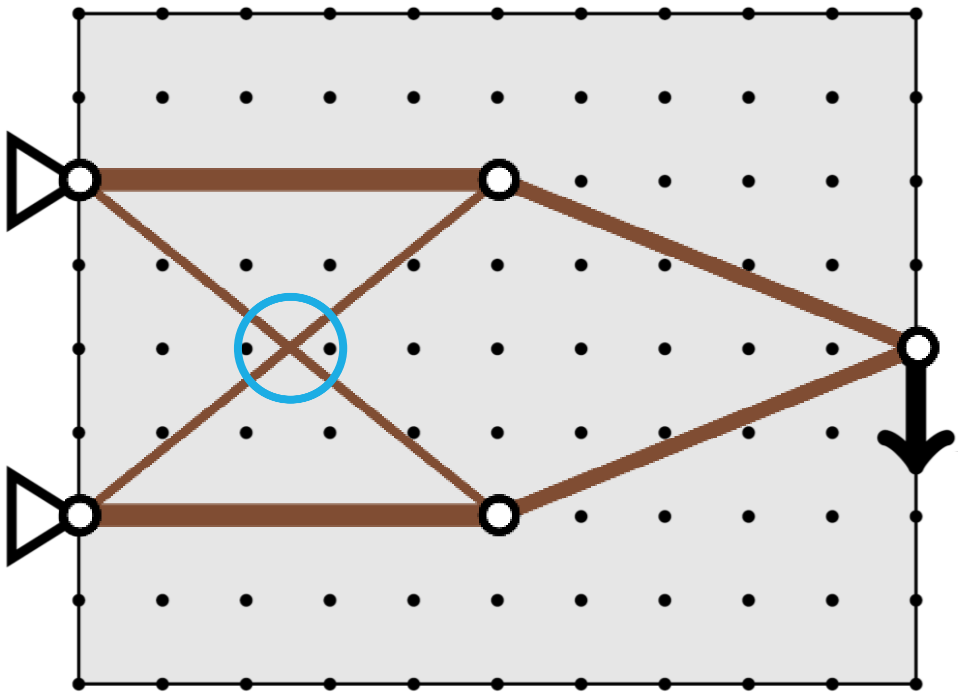
$v_i = 1$ if node i has any connected members with non-zero area (i.e. if it exists)

$$\Sigma v \leq \eta$$ allow up to η joints

Now a Mixed Integer Linear Programming (MILP) problem

Simple cantilever example

- Maximum of 5 joints permitted – using MILP approach
- Fully connected ground structure used:
  - 99 nodes
  - 4851 potential members = new integer variables
  - 11.8 million pairs of potential members (to check)
  - 2.8 million of which intersect = additional constraints



Max. 5 joints

Preventing crossovers

min	$V = \boldsymbol{l}^{\mathrm{T}}\boldsymbol{a}$	minimising volume
subject to	$\boldsymbol{Bq} = \boldsymbol{f}$	equilibrium
	$\lvert\boldsymbol{q}\rvert - \sigma\boldsymbol{a} < \boldsymbol{0}$	limiting stress
	$v_j - \sum_{i\in J_j} a_i \geq 0,\ v_j \in \{0,1\}$	v_j = 1 if joint j exists
	$\Sigma v \leq \eta$	allow up to η joints

4851 new integer variables	$Mw_i - a_i \geq 0$ $w_i \in \{0,1\}$	$w_i = 1$ if member i has non-zero area (i.e. exists)
~~2.8 million new constraints~~ implemented as 'lazy' constraints	$\cancel{w_h + w_i \leq 1}$	intersecting members h and i can't both exist

Michell cantilever example – with ‘lazy’ crossover constraints

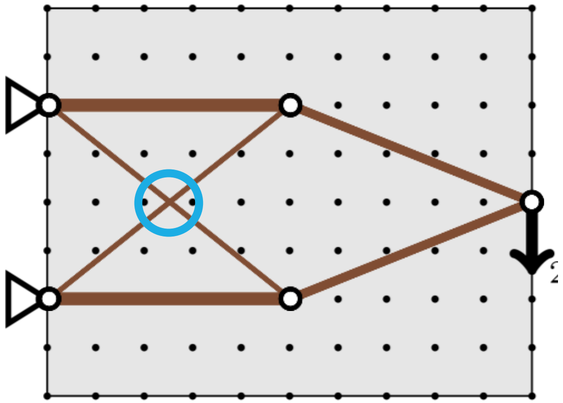
- Add new constraint:
  $$w_h + w_i \leq 1$$
  where h and i are the intersecting bars

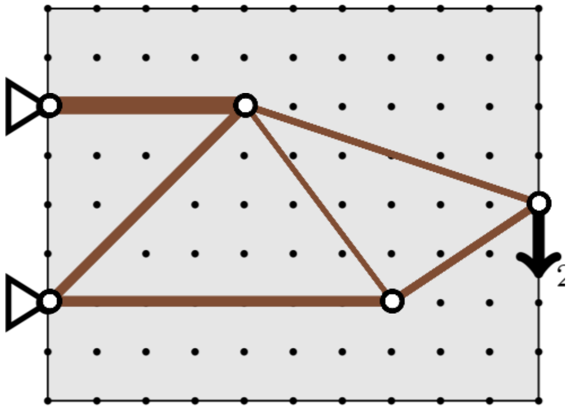
- Potential bound rejected, search continues…

- 3 lazy constraints added before optimal solution obtained (cf 2.8 million)

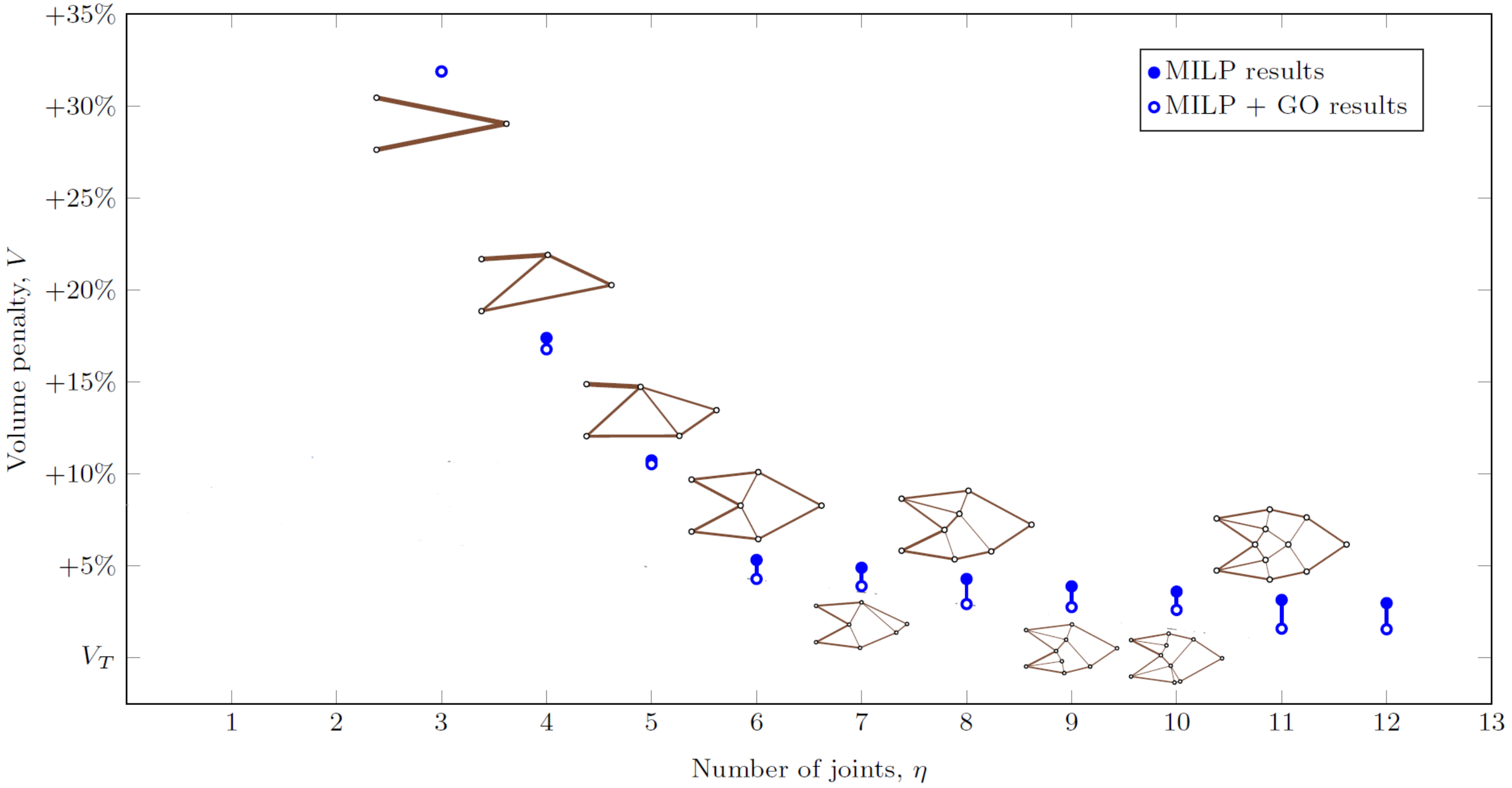
- Total solution time massively reduced to tractable levels

Max. 5 joints





Further developments:



- CPU times for shown results in range 10 – 290 seconds

- Multiple load cases
- Allow (and count) crossovers
- Enforce symmetry
- Minimum angle between members
- Speed up of 20 times (for smaller problems)

For details see:

Fairclough H, Gilbert M, ‘Layout optimization of simplified trusses using mixed integer linear programming with runtime generation of constraints’. SMO, Online First, 2020.

Automatic simplification approaches

Approach 1

Adding complexity constraints from the outset

Rigorous, but often very slow

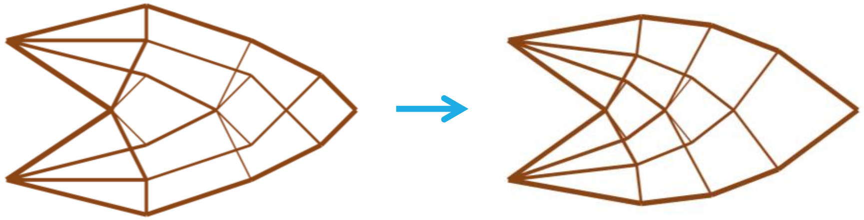
Approach 2

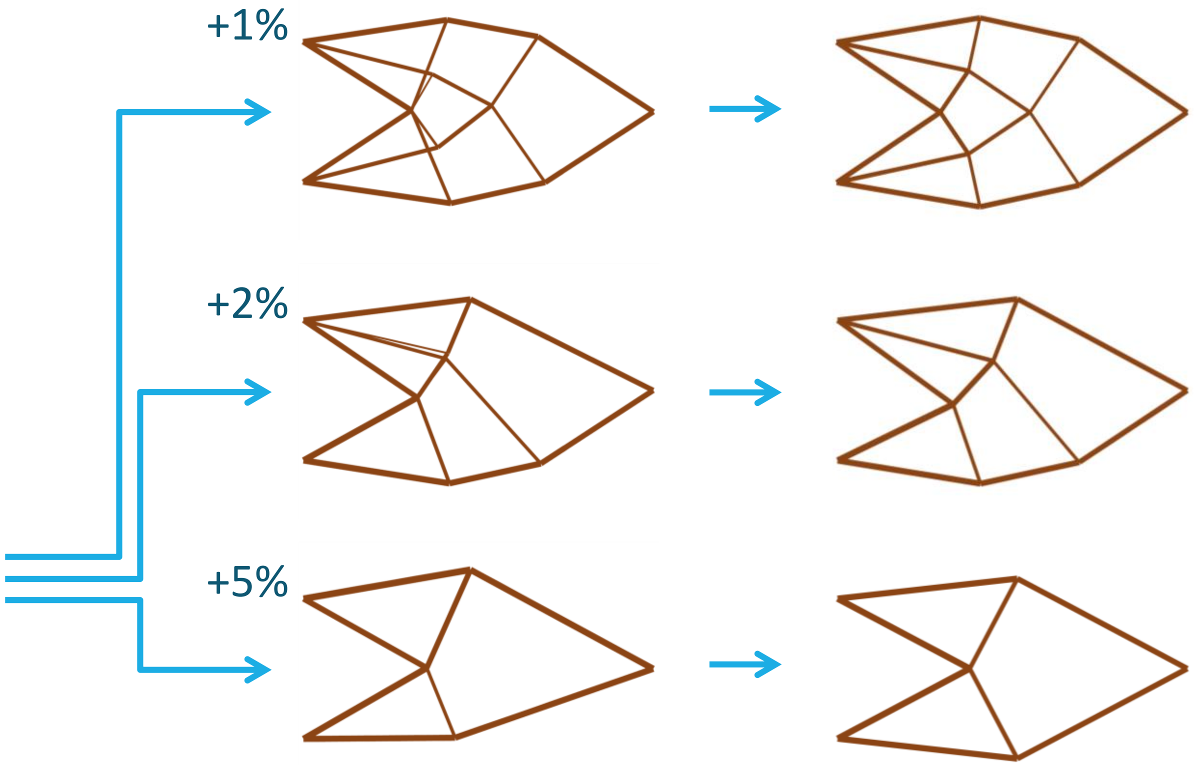
Automatically post-processing an existing solution

Fast, but results not rigorous

Fast post-processing approach

- Modified geometry optimization formulation:
  - Minimize number of members
  - Subject to limit on increase in volume
- 0-1 member on-off variables approximated via smooth Heaviside function





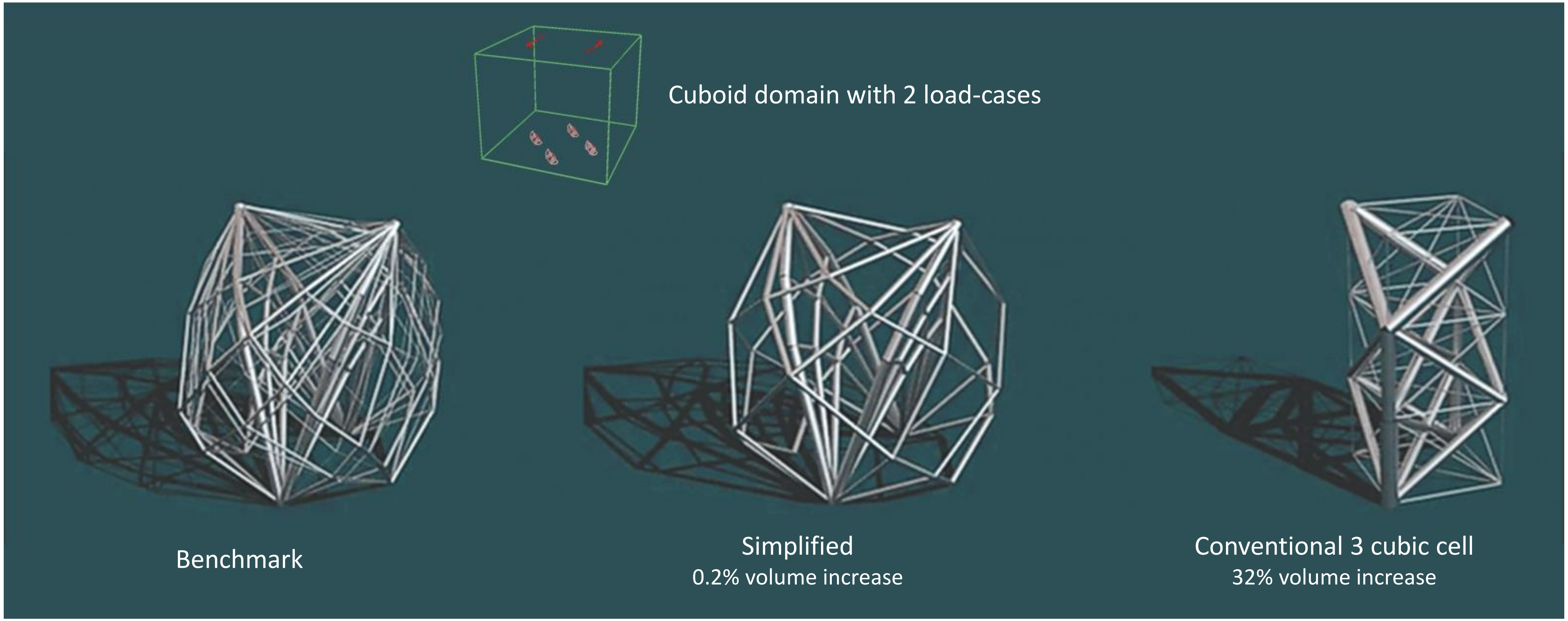
Layout optimization result (0.1s)

Geometry optimization result (0.3s)

Simplification result (~0.5s)

Geometry optimization result (~0.2s)

More complex example

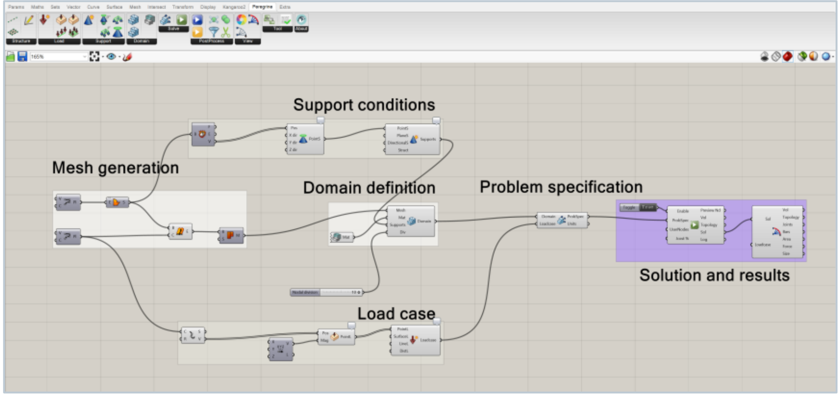


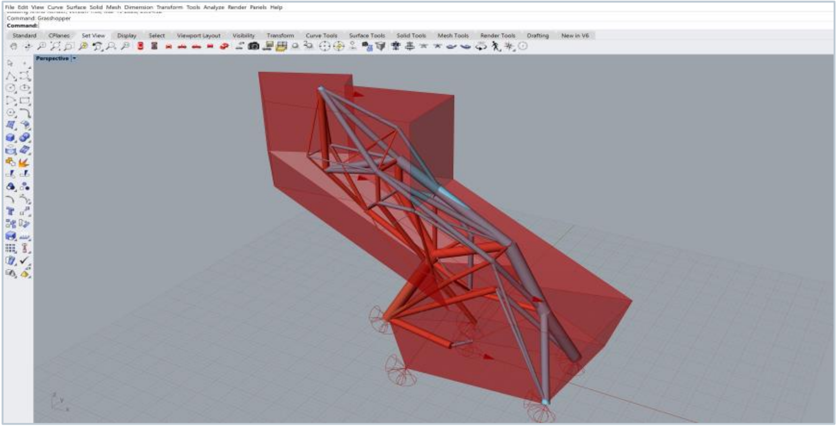
He et al, Proc. 2018 IASS Symposium, MIT, 2018.

Peregrine

- Plugin for Rhino/Grasshopper parametric CAD software
- Specify complex 3D geometries and interact in real time
- Link to download and to register for upcoming webinars (first on 9th July 2020):

www.buildopt.org





LayOpt

- Launches **today**!
- No download - simply open and experiment
- Usable on desktop, mobile or tablet

www.layopt.com