Geometric Visualization of LP Algorithms

Robert J. Vanderbei

2017 October 22

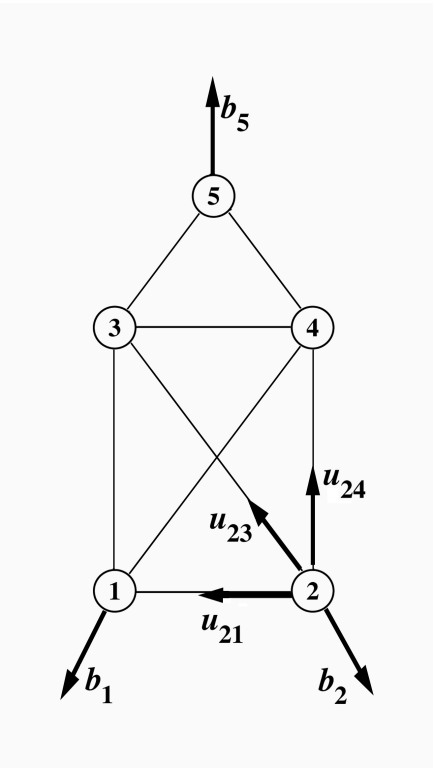
INFORMS
Houston TX

http://www.princeton.edu/∼rvdb

Structural Optimization

Forces: x_{ij} = tension in *beam* (aka *member*) $\{i, j\}$.

- $x_{ij} = x_{ji}$.
- Compression = -Tension.



Force Balance:

Look at joint 2:

$$x_{12}\begin{bmatrix} -1 \\ 0 \end{bmatrix} + x_{23}\begin{bmatrix} -0.6 \\ 0.8 \end{bmatrix} + x_{24}\begin{bmatrix} 0 \\ 1 \end{bmatrix} = -\begin{bmatrix} b_2^1 \\ b_2^2 \end{bmatrix}$$

Notations:

$\vec{p}_i$ = position vector for joint i

$\vec{u}_{ij} = \dfrac{\vec{p}_j - \vec{p}_i}{\|\vec{p}_j - \vec{p}_i\|}$ (Note $\vec{u}_{ji} = -\vec{u}_{ij}$)

Constraints:

$$\sum_{\substack{j: \\ \{i,j\}\in\mathcal{A}}} \vec{u}_{ij} x_{ij} = -\vec{b}_i \qquad i = 1, \ldots, m.$$

Matrix Form

$$Ax = -b$$

$$x^T = \begin{bmatrix} x_{12} & x_{13} & x_{14} & x_{23} & x_{24} & x_{34} & x_{35} & x_{45} \end{bmatrix}$$

$$A = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 1 \end{bmatrix} & \begin{bmatrix} .6 \\ .8 \end{bmatrix} & & & & & \\ \begin{bmatrix} -1 \\ 0 \end{bmatrix} & & & \begin{bmatrix} -.6 \\ .8 \end{bmatrix} & \begin{bmatrix} 0 \\ 1 \end{bmatrix} & & & \\ & \begin{bmatrix} 0 \\ -1 \end{bmatrix} & & \begin{bmatrix} .6 \\ -.8 \end{bmatrix} & & \begin{bmatrix} 1 \\ 0 \end{bmatrix} & \begin{bmatrix} .6 \\ .8 \end{bmatrix} & \\ & & \begin{bmatrix} -.6 \\ -.8 \end{bmatrix} & & \begin{bmatrix} 0 \\ -1 \end{bmatrix} & \begin{bmatrix} -1 \\ 0 \end{bmatrix} & & \begin{bmatrix} -.6 \\ .8 \end{bmatrix} \\ & & & & & & \begin{bmatrix} -.6 \\ -.8 \end{bmatrix} & \begin{bmatrix} .6 \\ -.8 \end{bmatrix} \end{bmatrix}, \quad b = \begin{bmatrix} b_1^1 \\ b_1^2 \\ b_2^1 \\ b_2^2 \\ b_3^1 \\ b_3^2 \\ b_4^1 \\ b_4^2 \\ b_5^1 \\ b_5^2 \end{bmatrix}.$$

Notes:

- $\|\vec{u}_{ij}\| = \|\vec{u}_{ji}\| = 1$.
- $\vec{u}_{ij} = -\vec{u}_{ji}$.
- Each column contains a $\vec{u}_{ij}$, a $\vec{u}_{ji}$, and rest are zero.
- In one dimension, exactly a node-arc incidence matrix.

Minimum Weight Structural Design

$$\begin{aligned} \text{minimize} \quad & \sum_{\{i,j\}\in\mathcal{A}} l_{ij}|x_{ij}| \\ \text{subject to} \quad & \sum_{\substack{j:\\ \{i,j\}\in\mathcal{A}}} \vec{u}_{ij}x_{ij} = -\vec{b}_i \qquad i = 1, 2, \ldots, m. \end{aligned}$$

Not quite an LP.
Use a common trick:

$$\begin{aligned} x_{ij} &= x_{ij}^+ - x_{ij}^-, \qquad x_{ij}^+, x_{ij}^- \geq 0 \\ |x_{ij}| &= x_{ij}^+ + x_{ij}^- \end{aligned}$$

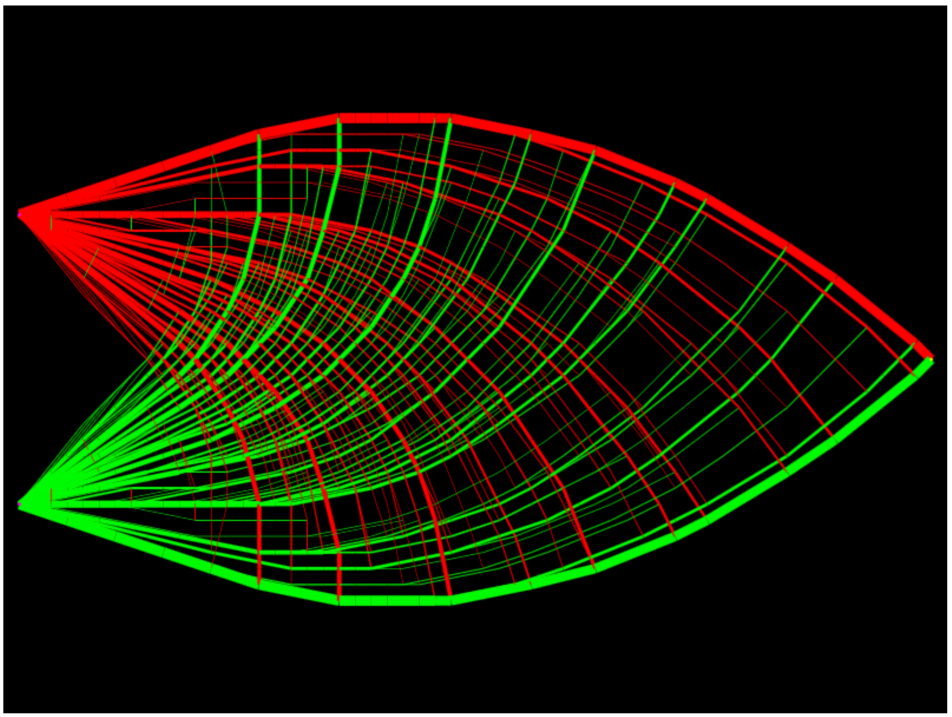
Reformulated as an LP:

$$\begin{aligned} \text{minimize} \quad & \sum_{\{i,j\}\in\mathcal{A}} (l_{ij}x_{ij}^+ + l_{ij}x_{ij}^-) \\ \text{subject to} \quad & \sum_{\substack{j:\\ \{i,j\}\in\mathcal{A}}} (\vec{u}_{ij}x_{ij}^+ - \vec{u}_{ij}x_{ij}^-) = -\vec{b}_i \qquad i = 1, 2, \ldots, m \\ & x_{ij}^+,\ x_{ij}^- \geq 0 \qquad \{i,j\} \in \mathcal{A}. \end{aligned}$$

The Michell Bracket (1904)

Height: 45 Width: 60 Step Length: 0.9 Beam Strength: 3.0 x-load: 0.0 y-load: -1.0

Setup Solve Stop



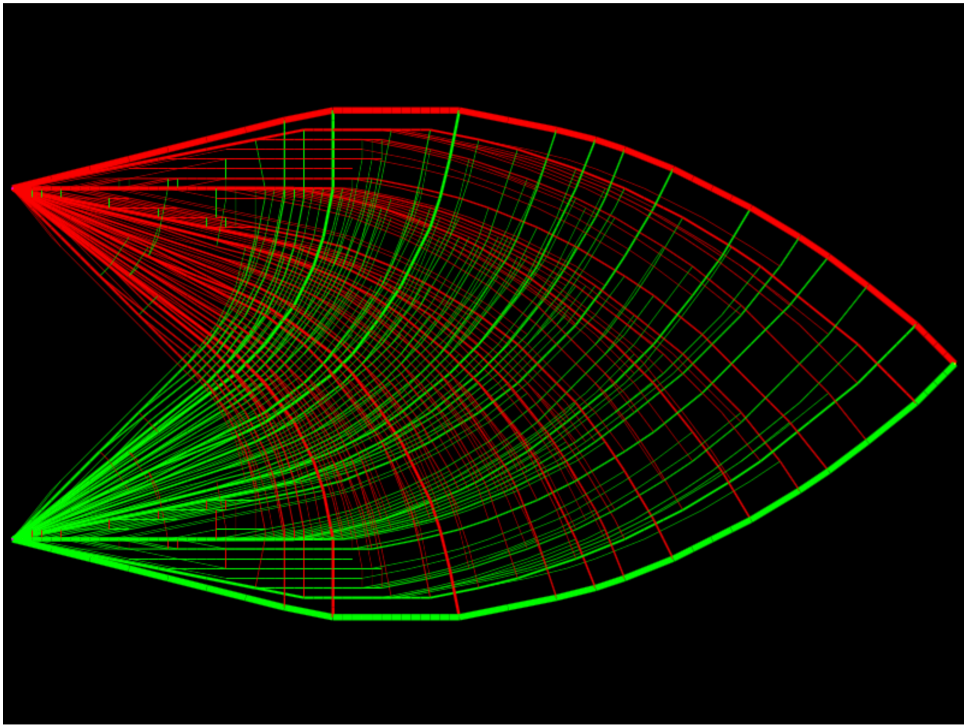
Constraints: 5,396
Variables: 193,310
Time: 270 seconds

Click here for parametric self-dual simplex method animation tool.
Click here for affine-scaling method animation tool.

The Michell Bracket (1904)

Height: 75 Width: 100 Step Length: 0.9 Beam Strength: 3.0 x-load: 0.0 y-load: -1.0

Setup Solve Stop



Constraints: 14,996
Variables: 536,972
Time: 1994 seconds

Click here for parametric self-dual simplex method animation tool.
Click here for affine-scaling method animation tool.

Algorithm Animation: Two-Phase Simplex Method

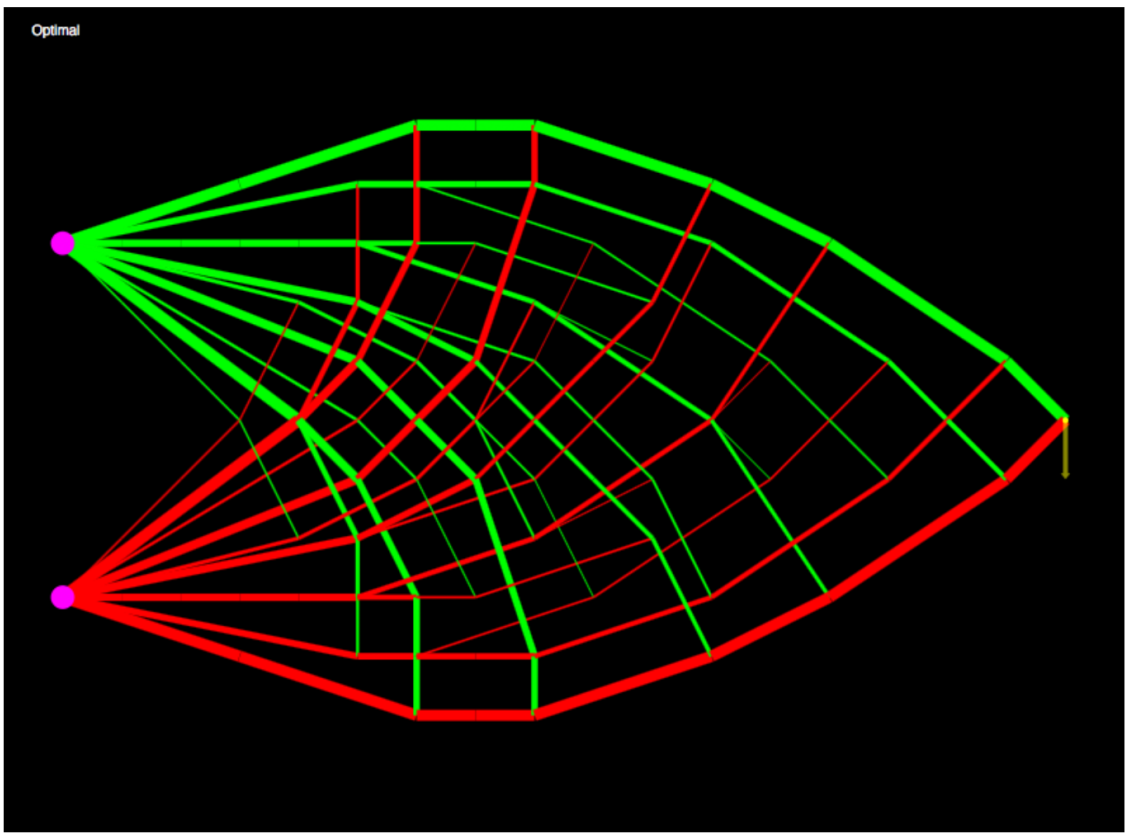
via Structural Optimization

(Click here for older Java version.)

Height: 15 Width: 20 Setup

Use the mouse with x-load: 0.0 y-load: -1.0 ... or ... Select from this list: Wall Bracket

Beam Strength: 10.0 Show: Every 200 Iterations Solve



Algorithm Animation: Primal-Dual Simplex Method

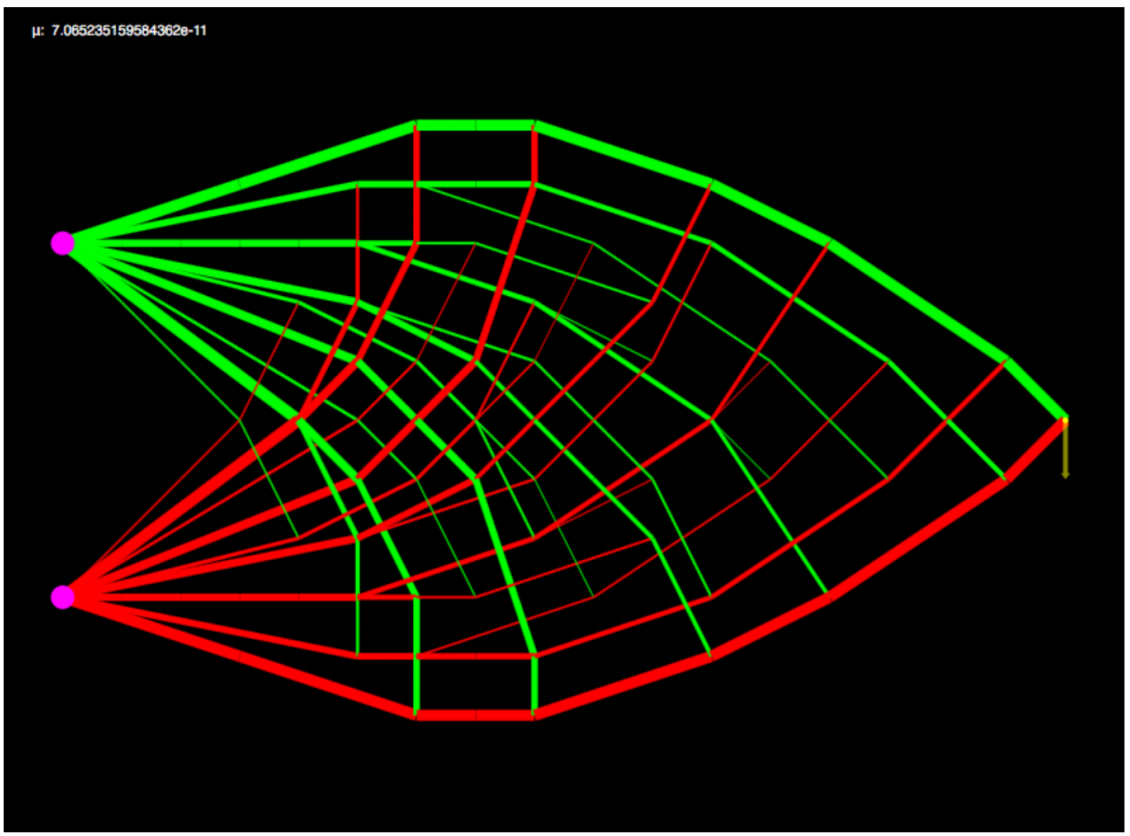
via Structural Optimization

(Click here for older Java version.)

Height: 15 Width: 20 Setup

Use the mouse with x-load: 0.0 y-load: -1.0 ... or ... Select from this list: Wall Bracket

Beam Strength: 10.0 Show: Every 200 Iterations Solve



Algorithm Animation: Affine Scaling

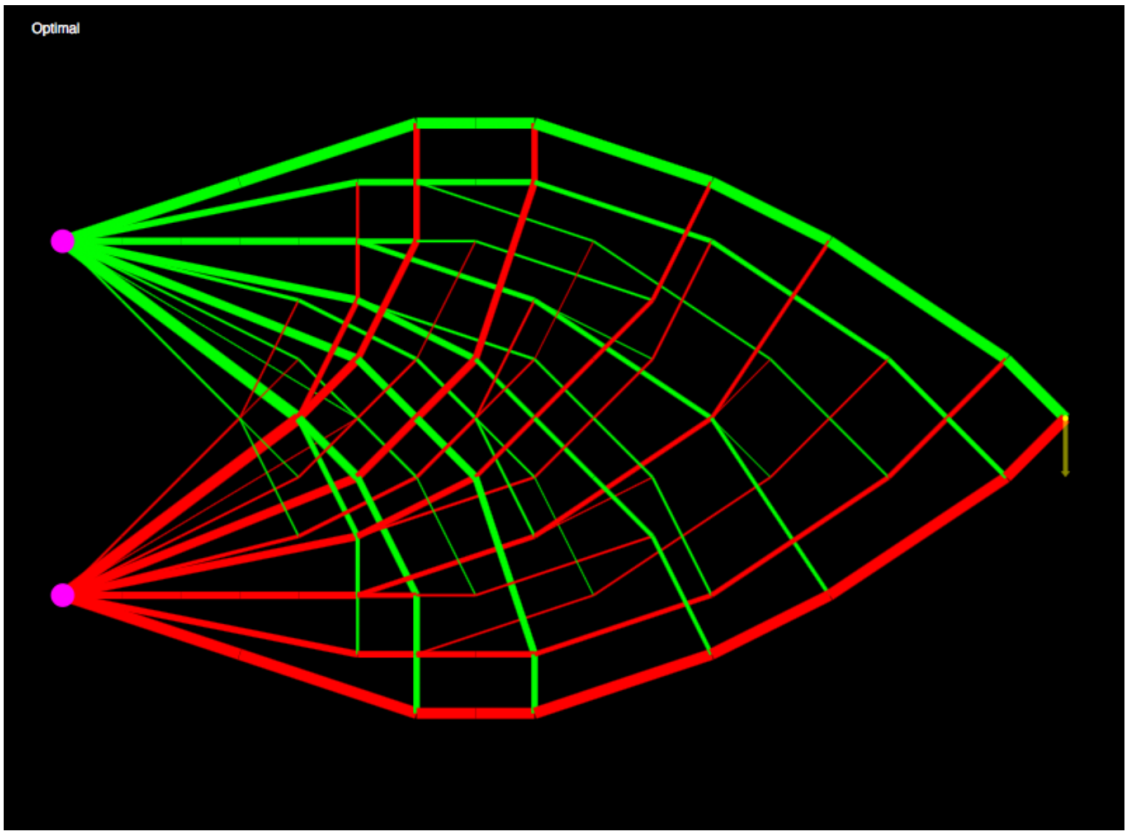
via Structural Optimization

(Click here for older Java version.)

Height: 15 Width: 20 Neighbors: 5 Setup

Use the mouse with x-load: 0.0 y-load: -1.0 ... or ... Select from list Wall Bracket

Step Length: 0.9 Beam Strength: 10.0 Solve



Algorithm Animation: Two-Phase Simplex Method

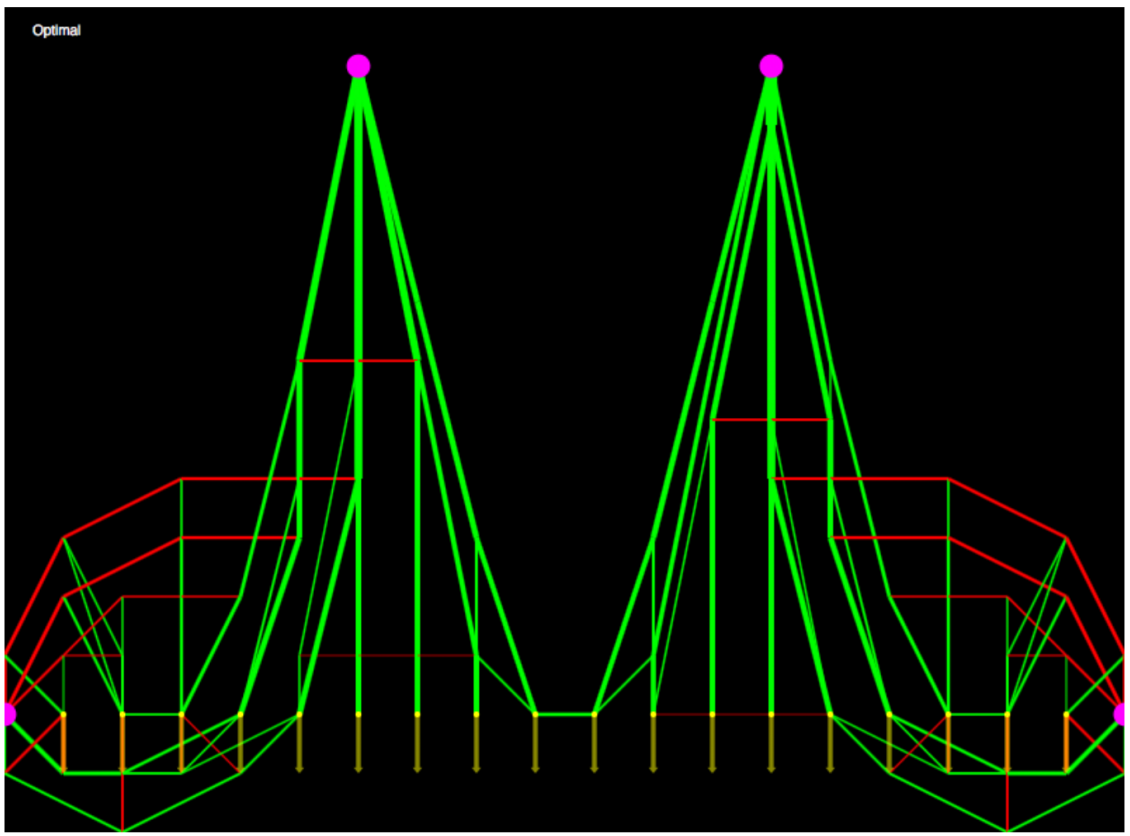
via Structural Optimization

(Click here for older Java version.)

Height: 15 Width: 20 Setup

Use the mouse with x-load: 0.0 y-load: -1.0 ... or ... Select from this list: Suspension Bridge

Beam Strength: 10.0 Show: Every Iteration Solve



Algorithm Animation: Primal-Dual Simplex Method

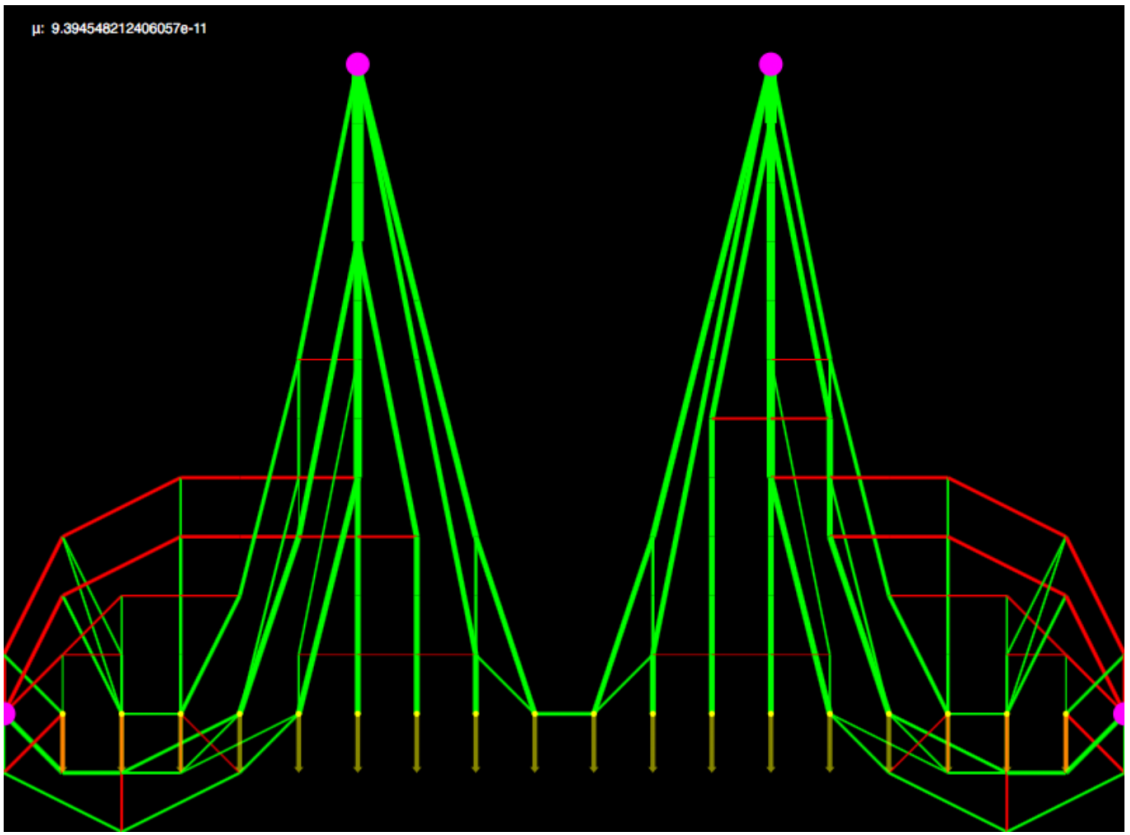
via Structural Optimization

(Click here for older Java version.)

Height: 15 Width: 20 Setup

Use the mouse with x-load: 0.0 y-load: -1.0 ... or ... Select from this list: Suspension Bridge

Beam Strength: 10.0 Show: Every Iteration Solve



Algorithm Animation: Affine Scaling

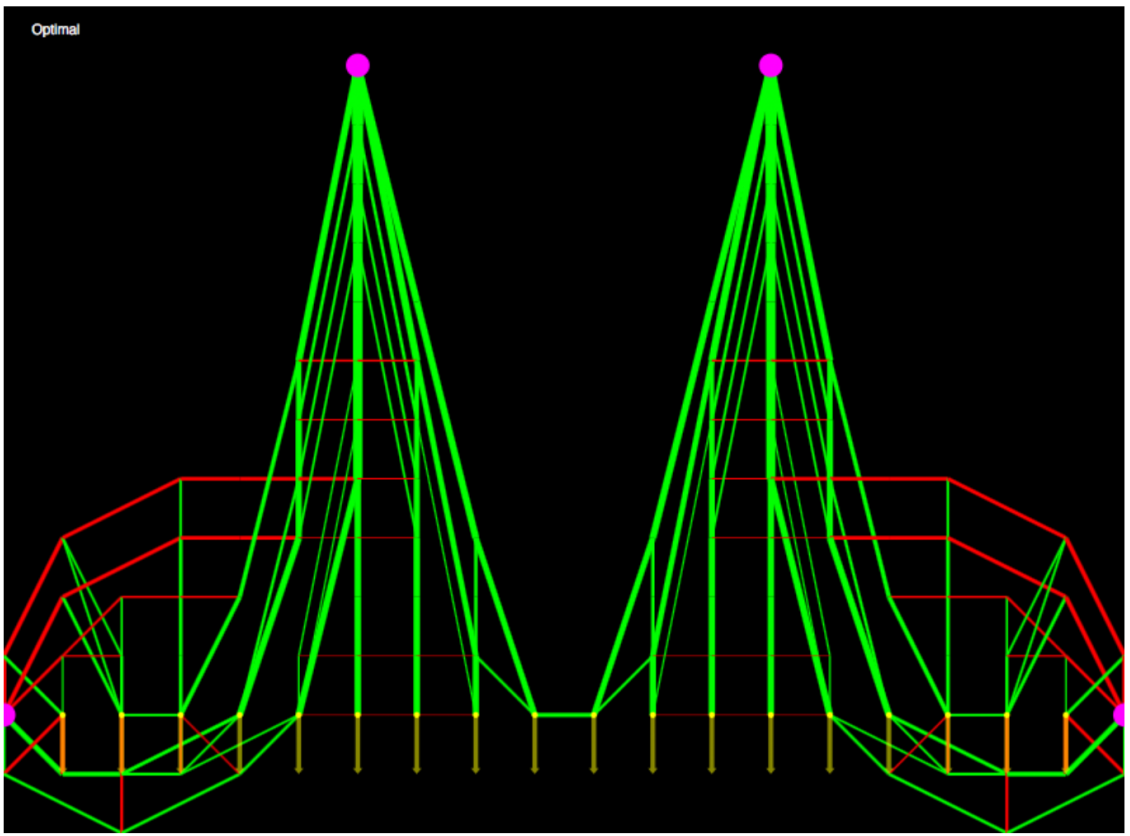
via Structural Optimization

(Click here for older Java version.)

Height: 15 Width: 20 Neighbors: 5 Setup

Use the mouse with x-load: 0.0 y-load: -1.0 ... or ... Select from list Suspension Bridge

Step Length: 0.9 Beam Strength: 10.0 Solve



Observations

- Basic optimal solutions lack geometric symmetry.
- The Two-Phase Simplex Method finds a feasible solution in about half of solution time.
- The feasible-but-not-optimal solutions look terrible (and bizarre).
- The Affine Scaling algorithm looks good all along but is not technically feasible until the very end.