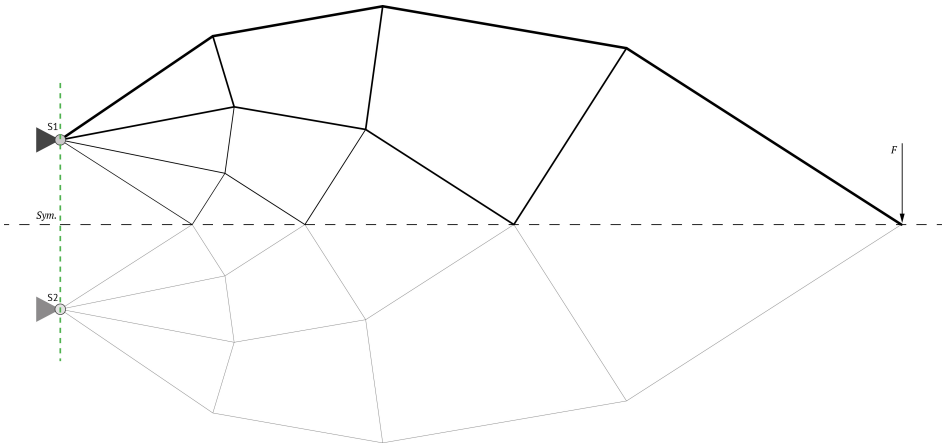
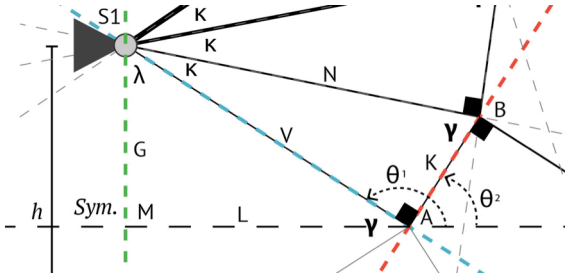
Exploring Michell Structures

Alexandros Charidis
MIT Spring 2015



Parametric Framework

In its standard symmetric form, the discrete optimum truss (see above image) can be constructed geometrically given a single angle γ and combinations of the following variables: H, the vertical distance between the two supports, L the horizontal distance between the supports (S1 and S2), a point load F applied at the tip of the structure, and n the total number of bars in the truss. There exist certain geometric constraints that aid in the construction of the optimum truss, but also provide a convenient parametric framework for automating the generation of alternative optimum trusses by varying the values of h and n. In the context of this application, L is kept constant at 40 ft.



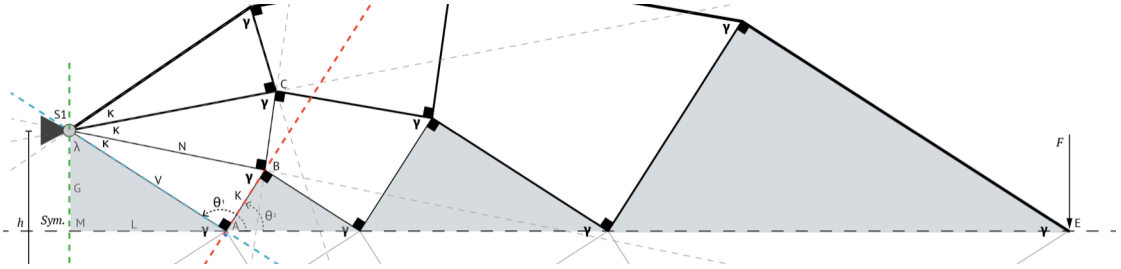
Start from the triangle defined by the first support point S1, the point M, which lies on the symmetry axis – the midpoint of the line defined by S1 and S2 – and A, the point of intersection of the blue line with the symmetry axis (green line in the Figure on the left). Point A is calculated using trigonometric rules. In particular, angle θ_1 is $90^o + \lambda$ where λ is $90^o - \gamma/2$ as observed from Mazurek et al. (2011) and it represents the directional factor λ_1 of the blue line. Thus A can be derived by the formula that computes the directional factor of any line given two points that lie on that line – S1 and A are those two points.

$$\lambda_1 = \frac{y_2 - y_1}{x_2 - x_1} \text{ where } S1(x_1, y_1) \text{ and } A(x_2, y_2)$$

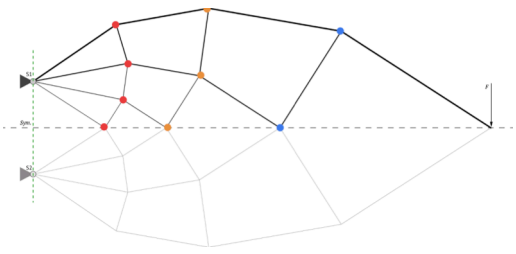
We know that y_1 is h/2 and $x_1 = y_2 = 0$. Therefore, we derive that $x_2 = x_1 - y_1/\lambda_1$. To find point B(x, y) I use the fact that the red line where A and B lie is perpendicular to the blue line. We know that $\lambda_1 \cdot \lambda_2 = -1$ where λ_2 is the directional factor of the red line, and thus $\lambda_2 = -1/\lambda_1$. We also know that the directional factor of a line is equal to the tangent of the angle θ_2 between the line itself and the *x'x* axis. From that I deduce that angle $\theta_2 = \arctan(\lambda_2)$. In addition, it is easy to find the length of K by using the Pythagorean Theorem in the triangle *<S1–A–B>*. To find the coordinates of point B, I use polar coordinate representation using the length of K, angle θ_2 and L.

$$x = L + K \cdot cos(\theta_2), \qquad y = L + K \cdot sin(\theta_2)$$

Both from a computational point of view and an implementation point of view, my geometric solution for producing parametric Michell structures is based on two steps. The first step is computing the points that are located on the symmetry line – this is done in the same way I computed point A previously. By simple observation, we can see that the number of points located on the symmetry line is equal to the number of elements connected to a single support point. This number has the following form: $n = 2 \cdot n \cdot a^2 \Longrightarrow n \cdot a = \sqrt{n/2}$. Each of those points creates a right triangle irrespective of the number of elements of the structure. Those triangles are depicted in Figure below in grey color.



The rest of the points are computed based on the condition of verticality between two lines – the red one and the blue one. This system of two vertical lines works as a driver and designates a series of strands. A strand is a set of nodes located on a principle stress line – see the points in the following figure.



Furthermore, it is clear that all the equations that I used for computing the Michell geometries depend on a single angle γ that is between 0^0 and 90^o. This value was left without explicit definition. To find the value of γ for each geometry, I linearly search for a γ untill the overall length L is approximately equal to the given restriction of 40ft – within an epsilon threshold. There are other search mechanisms that are more efficient, but were not explored here. According to my runs, the values for angle γ for each different case are shown in Table 1.

Table 1

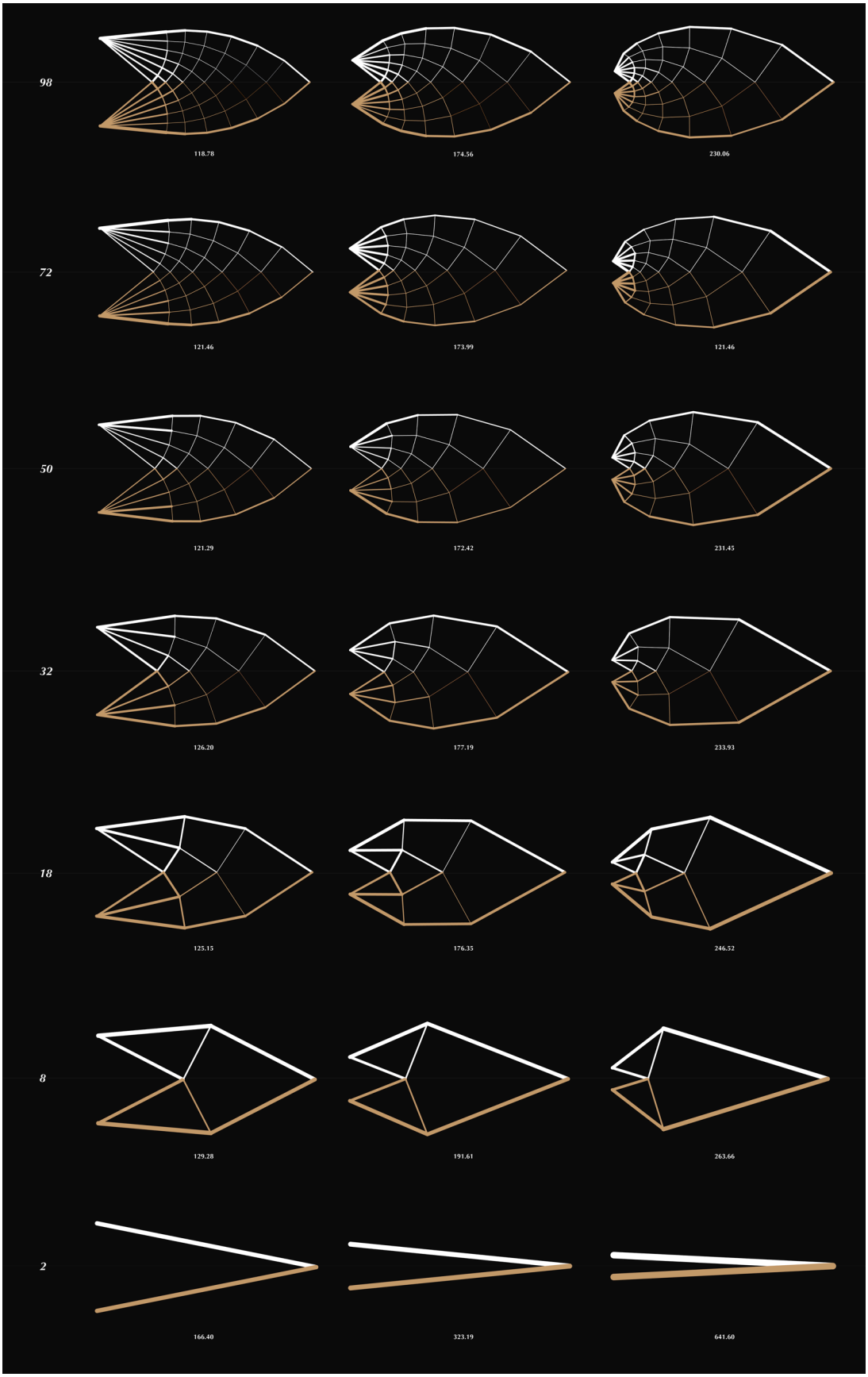
ne	*H (ft)*	*γ(degrees)*	*Performance*
2	16	22.6	166.40
8	16	54.6	129.28
18	16	66.5	125.15
32	16	72.4	126.20
50	16	76.2	121.29
72	16	78.6	121.46
98	16	80.4	118.78
2	8	11.4	323.19
8	8	43.1	191.61
18	8	57.6	176.35
32	8	65.3	177.19
50	8	70.3	172.42
72	8	73.6	173.99
98	8	75.9	174.56
2	4	5.7	641.60
8	4	34.1	263.66
18	4	49.5	246.52
32	4	58.7	233.93
50	4	64.7	231.45
72	4	68.8	228.98
98	4	71.8	230.06

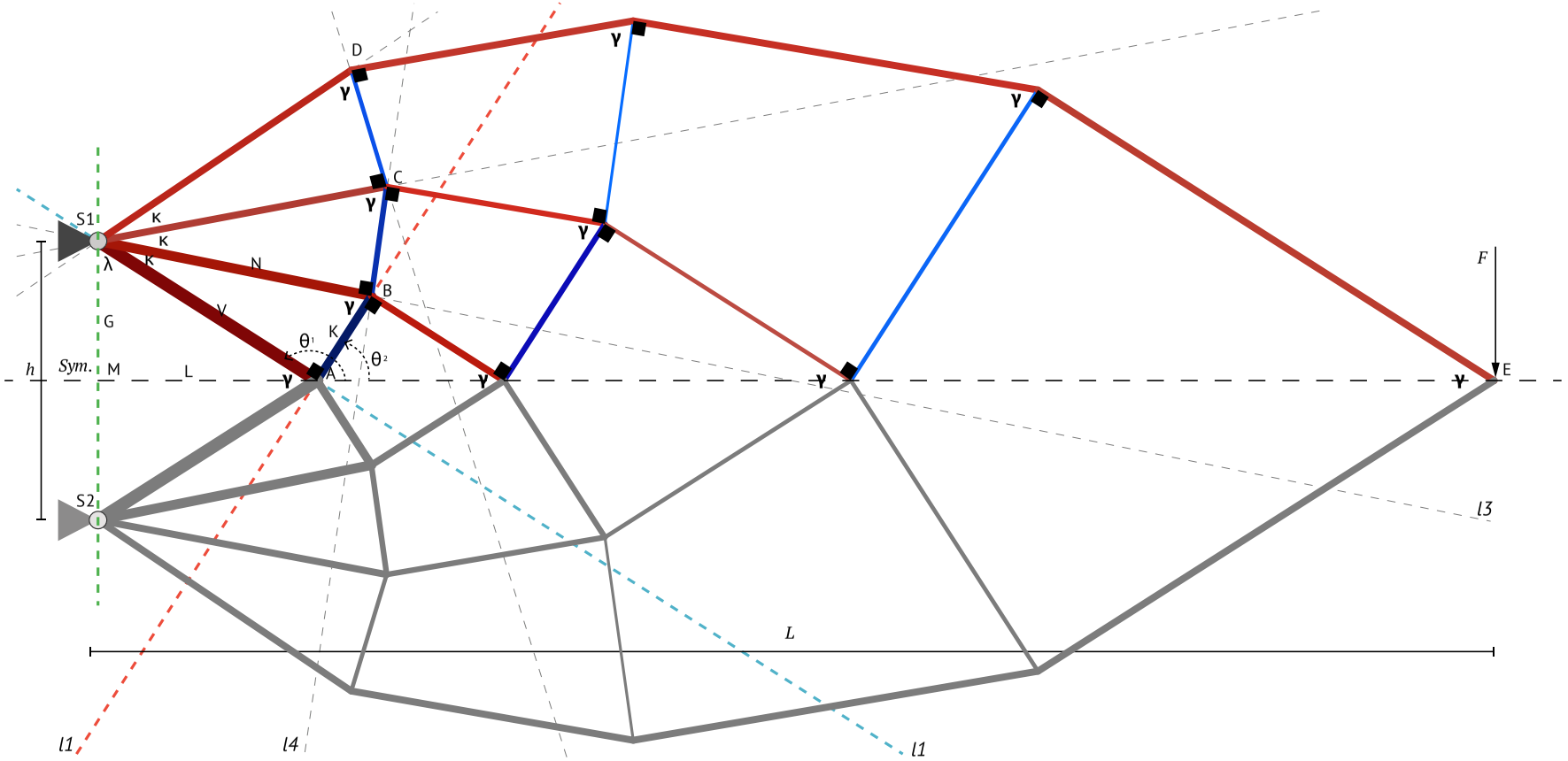
Structural Analysis

For each each Mitchell geometry produced from the above parametric framework, I measure its performance according to the the formula $\delta = \sum \frac{f_i \cdot l_i}{L}$ where f_i the force on each element, l_i the length of each element, and L the characteristic length of each element equal to unity (I assume all elements have the same characteristic length). I use the method of joints, implemented as a custom routine (`StructuralAnalysis.java`) which computes the forces on each member in each geometry. The performance for each geometry is also shown in Table 1. All the elements have their symmetrics, which are flipped along the horizontal symmetry axis. Two symmetric elements have forces of equal absolute value but opposite sign. The colors in the figure given at the end of this writup show the material distribution (optimal) for each different geometry. The optimum cross section areas of the elements are proportional to the absolute values of the forces that act on these elements (all elements of each optimized structure are subject to the same absolute stress value).

Acknowledgements

This application was developed as part of an assignment in the course *4.s48 Computational Structural Design and Optimization* (now 4.450J/1.575J) taught by Prof. Caitlin Mueller at MIT in the Spring semester of 2015. The routine for the method of joints was originally written in Matlab by Stavros Tseranidis (MIT 2015).


98
118.78
174.56
230.06
72
121.46
173.99
121.46
50
121.29
172.42
231.45
32
126.20
177.19
233.93
18
125.15
176.35
246.32
8
129.28
191.61
263.66
2
166.40
323.19
641.60


D
C
B
A
S1
S2
K
K
K
λ
N
V
G
h
Sym.
M
L
θ¹
θ²
γ
F
E
l3
L
l1
l4
l1