Particle-spring systems for structural form finding

Article *in* Journal of the International Association for Shell and Spatial Structures · August 2005

2 authors:

Axel Kilian
Massachusetts Institute of Technology

John Ochsendorf
Massachusetts Institute of Technology

PARTICLE-SPRING SYSTEMS FOR STRUCTURAL FORM FINDING

Axel Kilian[1] and John Ochsendorf[2]

[1] PhD Candidate in Computation, Department of Architecture, MIT, Cambridge MA USA 02139
[2] Assistant Professor, Building Technology Program, MIT, Cambridge MA USA 02139

SUMMARY

Particle-spring systems are well-known in computer science for creating physical simulations. In this paper, we propose the use of particle-spring systems for finding structural forms composing only axial forces. The equilibrium position of each particle is found using a Runge-Kutta solver, which allows the user to interact with the simulation while it is running. Several examples illustrate the technique, beginning with two-dimensional funicular forms and extending to three-dimensional networks. The paper proposes a novel three-dimensional design and analysis tool, which can be used by engineers and architects to find structural forms in real time.

Key words: *Form finding, hanging model experiments, tension structures, funicular form, grid shells*

1. INTRODUCTION

Structural form finding is a classical problem for long-span roof systems and lightweight structures. The methods of dynamic relaxation [1] and force density [2] have been used for decades in form finding of fabric roof systems and grid shells. Generally, these solution procedures are useful for finding the equilibrium position of a structural network with a desired level of internal force, as in the case of a pre-stressed membrane structure. Though such methods are highly developed for pre-stressed systems, conventional form finding methods are not very suitable for finding the form of statically determinate structures which act in pure tension or compression under their self weight, as in the case of a network of hanging chains. Conventional finite element methods are also unsuitable for modeling a network of hanging chains, because the large displacements involved cause numerical instabilities and violate the necessary assumptions of small displacements. More recently, optimization techniques using finite element methods have illustrated great potential for refining a structural form to minimize bending [3]. This method allows the designer to begin with an inefficient form, such as a non-structural shape for a concrete shell, and then search for a more efficient form by minimizing local bending stresses. In short, there are many analysis tools for refining forms, but there are few design tools for exploring and creating new structural forms.

Many designers have experimented with hanging chain models and other physical methods for finding efficient structural forms acting in pure tension or pure compression. For over 40 years, Heinz Isler has promoted the use of physical models as the most appropriate method to discover three-dimensional forms [4][5][6]. Similarly, Frei Otto and his colleagues in Stuttgart have developed highly accurate physical experiments for finding structural form [7]. In the early 20th century, Antoní Gaudi employed hanging models in the form-finding processes for the chapel of the Colonia Guell [8] and the arches of the Casa Mila [9]. Such hanging forms are often called *funicular* derived from the Latin word *funiculus*, meaning thin cord or rope, because they represent the shape taken by a thin cord acting in pure tension under a given set of loads. As Robert Hooke recognized in the 17th century, such tension forms could be inverted to find the shape of structural forms acting in pure compression under the same loading. Heyman [10] has translated Hooke's theorem from Latin: "As hangs the flexible line, so but inverted will stand the rigid arch."

By using a chain of axial springs which do not allow bending, it is possible to model hanging cables under various loading conditions in real time. Consider the example of two masses, *m* and *3m*, which are hanging from three weightless springs, as in Figure 1. For particular values of spring length, spring stiffness, and mass, there will be one stable equilibrium position for this system. Reducing the length or increasing the stiffness of the springs will alter the equilibrium position so that the "sag" at the center is reduced. Similarly, increasing the length of the springs or reducing the spring stiffness will cause greater sag in the system and will produce an alternative equilibrium configuration. By varying the length or stiffness of each spring, it is possible to generate an infinite number of equilibrium positions which belong to the family of funicular forms for a given loading configuration. In the context of graphic statics, this is equivalent to moving the pole to a new location to create a different funicular polygon [11].

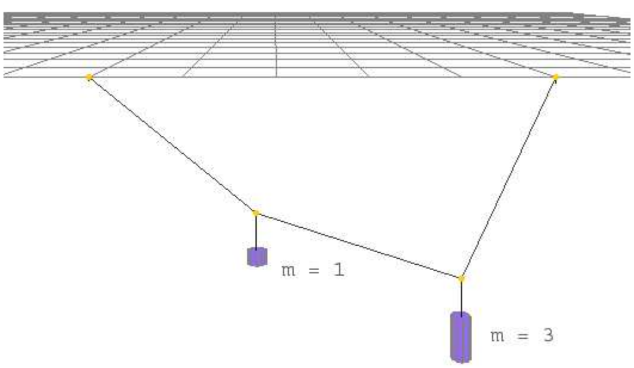

Figure 1. Equilibrium of simple particle spring system

This paper presents a novel approach for the exploration of funicular forms. Particle-spring systems serve as an excellent approximation for hanging models, by using axial springs connecting lumped masses to represent the physical behavior of weights hanging on strings. The equilibrium position of each mass is found using an iterative Runge-Kutta solver proposed by Baraff and Witkin [12]. Such methods are well-known in the computer graphics community, where researchers have developed efficient algorithms for solving large networks of particles connected by elastic springs. Particle-spring systems have been used extensively in cloth simulations and other graphics problems, particularly for making realistic simulations for the animation of clothing and other fabrics. After a brief description of the particle-spring method, two-dimensional and three-dimensional funicular forms will be derived using the method. Finally, advantages and disadvantages for the method are presented.

2. PARTICLE-SPRING SYSTEMS

Particle-spring systems are based on lumped masses, called particles, which are connected by linear elastic springs. Each spring is assigned a constant axial stiffness, an initial length, and a damping coefficient. Springs generate a force when displaced from their rest length. External forces can be applied to the particles, as in the case of gravitational acceleration. To solve for the equilibrium geometry of particle-spring systems, there are many techniques with varying degrees of efficiency and stability. The two primary classes of solution procedures are implicit and explicit solvers. For solving structural models, implicit solvers are more useful because of the high spring stiffness used in the models to minimize distortion of the initial spring lengths. With high spring constants, the small changes in the length of each element do not converge to an answer when using an explicit solver, making an implicit solver more preferable [13]. The implementation used here is based on an implicit Runge-Kutta solver, which aims to find the equilibrium position of each particle. The solver is part of a particle-spring system implemented for the Processing environment by Simon Greenwold [14]. The programming environment uses Java, which allows the method to be made freely available on the internet.

Each particle in the system has a position, a velocity, and a variable mass, as well as a summarized vector for all the forces acting on it. A force in the particle-spring system can be applied to a particle based on the force vector's direction and magnitude. Alternatively the magnitude of the force can be calculated using a function as in the case of springs. Springs are mass-less connectors between two particles that exercise a force on the particles based on the spring's offset from its rest length. Particles can be restrained in any dimension, so it is straightforward to add supports by restraining the displacement vectors of an individual particle.

The particle-spring system is usually not in equilibrium when the simulation is started and there will be movement throughout the system as the particles and springs seek their equilibrium positions. For the simulations presented here, each particle is assigned a mass and a gravitational field is applied to the entire system. Thus, the particles fall due to gravity either until the forces in the system reach equilibrium or the simulation is terminated. To prevent oscillations of the particles about their equilibrium positions, it is necessary to apply damping to the system. Damping can be applied as a coefficient to each spring. Alternatively, each particle can be subjected to viscosity in the surrounding environment as another damping method.

Eventually the system comes to an equilibrium state, if one exists. Even when the system no longer appears to move, the solver never stops as the system continues to update and refine the position of each particle. To stop the simulation, the user can define a limit, such as the velocity or the change in position of each particle as a cutoff threshold for the simulation. Papers by Baraff provide greater details on the solution procedure, including computer code for implementation [12][13].

Figure 2 illustrates the solution procedure for a series of springs supporting 40 masses at equal spacing, which approximates the form of a hanging chain. The particles are initially placed in a single line connected by springs with the outer two particles acting as fixed supports. As the simulation begins (t=0.5 seconds), the line of particles begins to fall. At t=2.0 seconds, the particles near the supports are approaching their equilibrium positions while the particles near the center continue to fall vertically. The solution at t=3.0 seconds begins to approximate a catenary, as will be discussed in more detail later. One of the primary attractions of the particle-spring approach is that it allows the user to watch the system approach equilibrium and to intervene during the solution process. The user can alter the applied loading, add or subtract structural elements, and change the support conditions in order to discover new structural forms while the simulation is running.

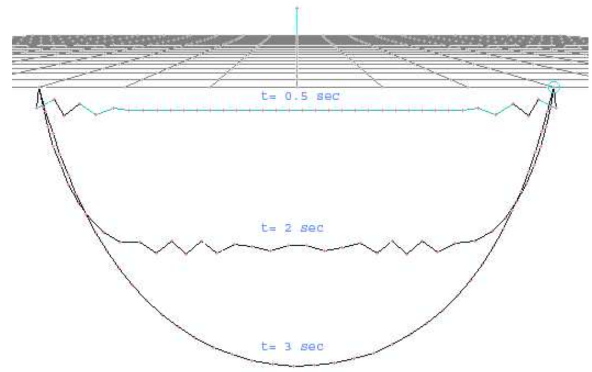

Figure 2. Solution process for a cable with forty discrete masses at equal spacing.

3. TWO-DIMENSIONAL SYSTEMS

For two-dimensional systems, unique funicular forms exist for given loadings and support conditions, as in the case of a catenary formed by a cable of given length hanging under its own weight. Such forms depend only on the length of each element and the applied loading, so that a unique solution exists. Combinations of hanging chains will create more complex funicular forms as in Figure 3, which is achieved by adding "chains" of closely spaced particles with uniform masses. Provided that each intersection has only three springs attached, the system is statically determinate and the funicular form is unique.

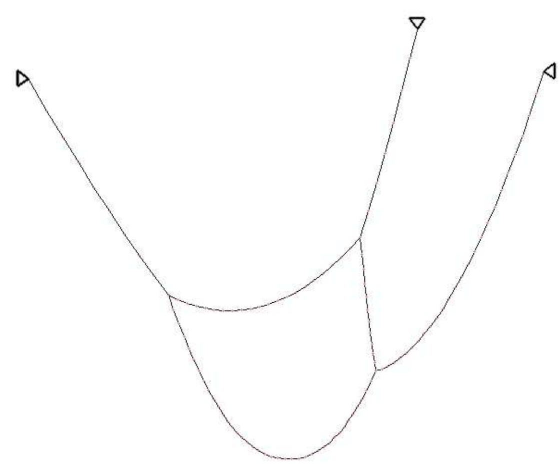

Figure 3. Statically determinate funicular form in 2D modeled with particle-spring simulations

Particle spring systems can be used to effectively model both statically determinate and indeterminate systems. For indeterminate systems, it is possible to develop compression struts which are equilibrated by tension elements and the axial spring model can account for these scenarios as well. Figure 4 illustrates the simplest example of an indeterminate two-dimensional system, a point load supported by three weightless bar elements with pinned connections. In this case, the force in each element varies with its length and axial stiffness and the exact value of force in each element can vary drastically with small changes in the geometry. The accompanying force polygon in Figure 4 uses graphic statics and Bow's notation to illustrate the internal forces in each element of the structure, such that vector **ab** on the force polygon is equivalent to the magnitude of the applied load AB [11]. The perfect elastic solution gives the greatest force in element CD and the smallest force in element AD, with tension in all three elements, as shown by the force polygon.

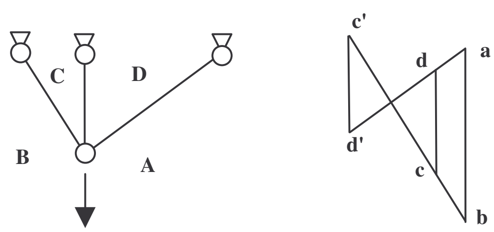

Figure 4. Statically indeterminate funicular system in 2D with the corresponding force polygon.

Many other equilibrium states are possible for this structural system, with each element acting as a tension, compression, or zero-force member. For example, if the length of member CD is longer than the prescribed length, it must be forced into position, causing a state of self-stress in the structure. In this case, a new equilibrium state will occur with element CD in compression and with increased tension in elements BC and AD. This state of self-stress is illustrated by the alternative force polygon showing the corresponding location of c'd' in Figure 4. There are an infinite number of valid equilibrium states for this structure, all of which can be illustrated with graphic statics, and can be effectively modeled with particle spring systems. However, for spring elements acting in compression, the solution procedure does not account for buckling, which must be implemented as an additional consideration.

4. THREE-DIMENSIONAL FUNICULAR SYSTEMS

If one considers the term *funicular* to mean "tension-only" or "compression-only" for a given loading, then it is also appropriate to identify three dimensional systems as funicular. Three dimensional funicular systems are considerably more complex because of the multiple load paths which are possible. Unlike the case of a hanging cable with a single funicular form, hanging membranes have multiple possible forms.

As an example, Figure 5 illustrates two continuous surfaces supported on a circular base: a cone and a shallow spherical dome. Both forms can contain compression-only solutions according to the membrane theory under an applied load of gravity [15][16][17]. Therefore, for statically indeterminate networks of intersecting elements, there is no such thing as a unique funicular solution. Each solution will depend on the exact length of each element.

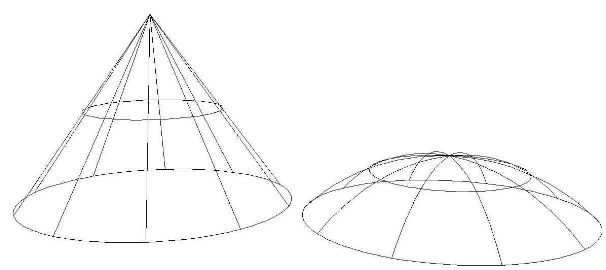

Figure 5. Examples of 3D membrane systems which can act in pure compression.

This fact is well-known to designers who seek three-dimensional funicular forms through experimentation. For example, Heinz Isler varies the amount of fabric in his hanging membrane models in order to alter the shape of the final design [5]. Simply by changing the length of each element, new equilibrium solutions are possible. For three dimensional networks, a spherical dome can become a conical shell by adjusting the length of each element. Both belong to the family of valid solutions for compression-only networks acting under their own weight. Of course some solutions perform better under varying load conditions and the double curvature of the dome is structurally superior to the single curvature of the cone in the event of asymmetrical live loading.

5. DESIGN EXAMPLES

To illustrate the potential for structural form finding with particle spring systems, four examples will be presented: a catenary, a square mesh, a 3-D cathedral structure, and a free-form grid shell.

Example #1: A catenary of length, L, is modeled by seven masses distributed initially at equal spacing along spring segments. As illustrated in Figure 6, the particle-spring solution varies only slightly from the theoretical catenary solution because of linear approximations of the particle spring solution. Though the solution from the particle spring model is not an exact catenary, it is a valid equilibrium solution for a hanging chain with equal weights distributed at slightly *uneven* spacing along the chain. It is not an exact catenary because the length of each spring segment is slightly different due to the varying internal forces. The difference between the nodes of the particle spring system is given as a percentage of the maximum sag of the catenary. In all cases, the particle spring solution varies by less than 0.01% of the sag of the cable.

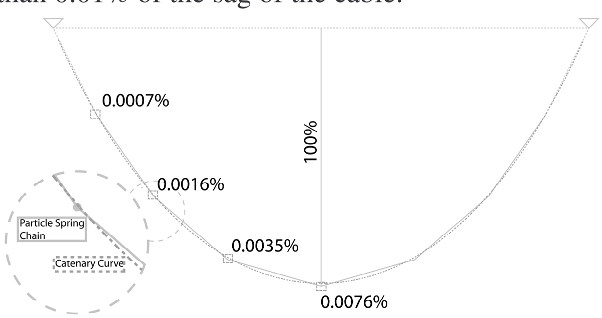

Figure 6. Comparison between theoretical and computational solutions for the catenary

Example #2: A rectangular grid mesh with 25 squares in each direction is restrained at four nodes near each corner. Each intersection of the mesh is assigned an equal mass and the particle-spring network is subjected to gravitational forces, causing it to form a dome-like structure. Figure 7 illustrates the solution procedure and the final equilibrium state of the shell. The hanging structure has been inverted to demonstrate the form of a compression shell. Each support is located four squares from the nearest edge, so that the resulting structure has upturned edges to stiffen the shell against buckling and asymmetrical loading. Because the interior particles in this mesh are connected to four springs, the system is statically indeterminate and the solution illustrated in Figure 7 contains both tension and compression elements. In general, the particle spring system gives solutions with upturned edges when the area of the surface lies outside the support points, as in the case of Isler's shells with upturned edges.

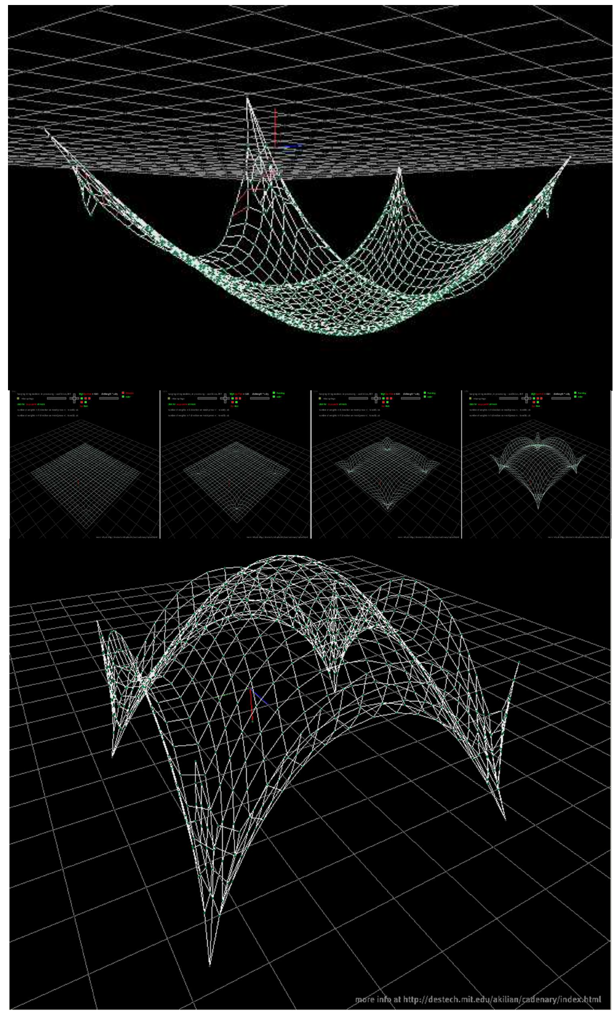

Figure 7. Simple square mesh supported near the corners

Example #3: A skeletal cathedral structure, similar to forms designed by Catalan architect Antoní Gaudi, is created from intersecting elements. The mass is lumped at a series of nodes, which are spaced evenly along each line element. Figure 8 provides a view through the central nave of such a structure. The building geometry is only represented by lines of force and not as meshes or surfaces. For simplicity, four bays of two-dimensional transverse arches are represented, though it is straightforward to add intersecting arches between each bay to create a fully three-dimensional structure. It is also straightforward to add additional arches and towers to this virtual model in the same manner that Gaudi experimented with physical hanging models. Using the particle spring approach, a three-dimensional structure such as this cathedral can be created in only a few minutes, whereas Gaudi developed his physical models over many years. In addition to its merits as a design tool, this method can also be used as an interactive analysis tool for historic masonry structures such as Gothic cathedrals, in which particle-spring systems can be used to represent thrust lines or thrust surfaces acting within the mass of the masonry.

Example #4: A grid shell is created with irregular support locations, as illustrated in Figure 9. A regular grid mesh is supported multiple times in addition to its corner attachment points. As before, each intersection of the mesh is assigned a mass and the structure deforms according to the applied gravity. This is an example of a more free form design exploration in which the user experiments with various support conditions. As with the square grid of example #2, this shell is statically indeterminate and the solution illustrated contains elements acting in both tension and compression.

6. DISCUSSION

Finding structural form using particle-spring systems has a number of advantages. Most importantly, the user can change form and forces in real time while the solution is still emerging. The environment educates the user as to the effects of forces on the form of structures and provides an interactive form-finding environment that was previously limited to physical models. By nature of the solution procedure, the particle-spring system always finds a possible load path for which the forces are in equilibrium. If no equilibrium solution exists, as in the case of an unsupported structure, then the masses will continue to translate in space until the user intervenes. The primary improvement over existing finite element methods is that this environment is fully dynamic, allowing the accurate computation of large displacements, velocities, and accelerations for the design of structural systems. The improvement over existing relaxation methods is that the design environment is fully interactive and dynamic, allowing the user to invent forms rather than analyzing existing forms.

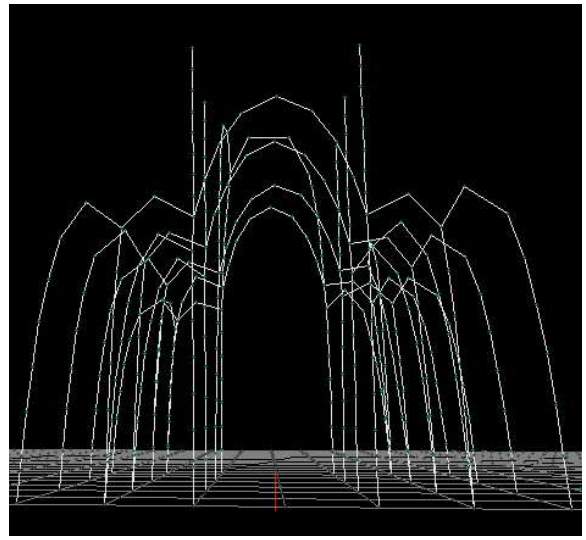

Figure 8. Cathedral structure

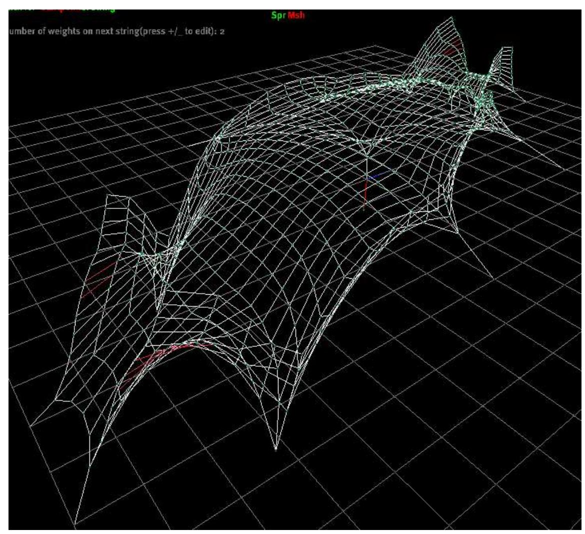

Figure 9. Free-form grid shell

Particle-spring systems can help to introduce structural evaluation environments into the architectural design process as early as possible, allowing the designer to interact with a form and to experiment with alternative solutions. The goal is to increase intuitive understanding for structural behavior of complex forms at the early design stage. The addition of some file export capabilities allows for the exchange of data with other applications so that models created in the particle-spring environment can be exported to more conventional design environments, such as CAD and standard finite element programs. This methodology can allow architects and engineers to explore forms with a structural basis in the same manner that some designers have used physical models in the past.

A number of disadvantages exist at present. Due to the properties of axial springs they not only contract upon being stretched but also expand upon being compressed. This creates the possibility of tension and compression members being present in a hanging structure at the same time. There are several remedies for this problem, such as removing the compression members in order to guarantee tension-only solutions. Another solution is to dynamically subdivide the compression member into two springs, which renders the springs incapable of taking compression forces and provides a tension-only solution. Additionally, the choice of an appropriate mesh pattern can allow for statically indeterminate systems, such that each particle is attached to three springs or fewer for three dimensional structures.

The solution procedures are somewhat expensive computationally, which limits the size of real-time simulation at this point. Current solutions are practical for solving up to 1,000 particles in real time on a desktop computer. Also, as exemplified in the case of the catenary, there is a slight loss of precision due to the approximation of the simulation. Using the system to model actual project geometry has its own challenges. The process is not static but the addition of string elements is dynamic (depending on the solver used) so that the shape and form is constantly in flux. The final equilibrium shape emerges from the modeled topology. As in the case of a physical hanging model, any addition disturbs the balance of the initial shape. Finally, another disadvantage is that for very complex problems in combination some solutions can become unstable. In general, the solution procedure is quite robust.

7. CONCLUSIONS

Particle spring systems provide a powerful new method for structural form-finding in a design context. The solutions allow large displacements and are valid beyond the realm of conventional finite element formulations, which assume small displacements. This method allows for real-time discovery of structural form rather than analysis or optimization of an existing form. Form finding processes at the design stage of the process make the designer aware of structural responses. This application represents an extension of particle spring systems beyond their initial role in character animation and cloth simulation within computer graphics.

There is significant future work to be done for this method to become a standard approach in structural form finding. Solutions without distortions of the original geometry can be accomplished by updating the length of each element to maintain its initial length. Additionally, various structural elements can be modeled with the particle-spring system, including trusses, slabs, beams, and rigid elements, which would allow users to explore a much wider range of structural forms. Future work will also address the question of different degrees of optimization. In the current model, the equilibrium form is not always optimal within the chosen topology and designers may wish to experiment with alternative solutions to achieve greater or lower efficiency. Designers may choose a less optimal structural form in order to accommodate other design constraints. Because design always mediates between many goals and hardly ever fits one of them perfectly, users should be able to diverge from the optimal structural solution to achieve greater design flexibility.

ACKNOWLEDGEMENTS

This paper presents the preliminary results of an ongoing research project at the Massachusetts Institute of Technology between the Department of Architecture and the Department of Computer Science. The initial particle-spring system used was implemented by Simon Greenwold in Java as a library for “Processing” (www.processing.org), an environment developed by Ben Fry and Casey Reas. The first author also developed an early prototype in collaboration with Megan Galbraith and Dan Chak. In addition, the authors co-taught two research workshops on this topic in collaboration with Dr. Barbara Cutler, Simon Greenwold, Dr. Erik Demaine, and Martin Demaine, who each contributed substantially through discussions on the topic and in making the workshops a success. In the course of the workshops a more robust C++ application was developed in a group project headed by Ryo Shimizu. Finally, students in each workshop helped to extend the range of structural applications for particle spring systems, and the authors gratefully acknowledge their contributions.

REFERENCES

[1] ***Day, A.S.*** "An Introduction to Dynamic Relaxation," The Engineer, Vol. 219, 1965.

[2] ***Schek, H.J.*** "Force Density Methods for Form Finding and Computation of General Networks," *Computer Methods in Applied Mechanics and Engineering*, 1974.

[3] ***Ramm, E.***, "Shape Finding of Concrete Shell Roofs," *Journal of the IASS*, 45(1), 2004.

[4] ***Isler, H.*** "New Shapes for Shells," *Bulletin of the IASS*, No. 8, 1960.

[5] ***Chilton, J.*** *The Engineer's Contribution to Contemporary Architecture: Heinz Isler*, Thomas Telford Press, London, 2000.

[6] ***Ramm, E., and Schunck, E.,*** *Heinz Isler Schalen*, Exhibition Catalogue, ETH Zurich, 2002.

[7] ***Rasch, B. and Otto, F***. *Finding Form*, Axel Menges, 1996.

[8] ***Tomlow, J.,*** *The Model,* Information of the Institute for Lightweight Structures, No. 34, University of Stuttgart, 1989.

[9] ***Huerta, S.,*** "El cálculo de estructuras en la obra de Gaudí," Ingeniería Civil, No. 130, CEDEX, Madrid, 2003, pp. 121-133.

[10] ***Heyman, J***. *Structural Analysis: A Historical Approach*, Cambridge University Press, 1998.

[11] ***Zalewski, W. and Allen, E.***, *Shaping Structures*, John Wiley, 1998.

[12] ***Baraff, D. and Witkin, A.*** "Large steps in cloth simulation," *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques* (SIGGRAPH), July, 1998.

[13] ***Baraff, D.*** "Implicit methods for differential equations," *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques* (SIGGRAPH), 2001.

[14] ***Fry, B. and Reas, C.,*** Processing web site, <http://www.processing.org>, accessed 12 January 2005.

[15] ***Billington, D.P.*** *Thin Shell Concrete Structures*, McGraw Hill, 1965.

[16] ***Heyman, J.*** *Equilibrium of Shell Structures*, Oxford University Press, 1977.

[17] ***Calladine, C.R.*** *Theory of Shell Structures,* Cambridge University Press, 1983.