Embedded Rationality: A Unified Simulation Framework for Interactive Form Finding

Ramtin Attar, Robert Aish, Jos Stam, Duncan Brinsmead, Alex Tessier, Michael Glueck and Azam Khan

Embedded Rationality: A Unified Simulation Framework for Interactive Form Finding

Ramtin Attar, Robert Aish, Jos Stam, Duncan Brinsmead, Alex Tessier,
Michael Glueck and Azam Khan

Abstract

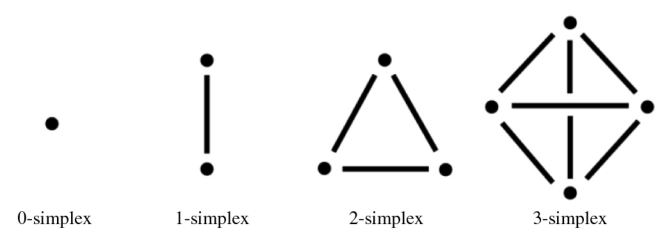
This paper describes *embedded rationality* as a method for implicitly combining fabrication constraints into an interactive framework for conceptual design. While the concept of 'embedded rationality' has been previously discussed in the context of a parametric design environment, we employ this concept to present a novel framework for dynamic simulation as a method for interactive form-finding. By identifying categories of computational characteristics, we present a unified physics-solver that generalizes existing simulations through a constraint-based approach. Through several examples we explore conceptual approaches to a fixed form where the resulting effects of interacting forces are produced in real-time. Finally, we provide an example of embedded rationality by examining a constraint-based model of fabrication rationale for a Planar Offset Quad (POQ) panelization system.

1. Introduction

As digital tools facilitate the development of increasingly complex forms, the concept of *design rationalization* has become a central research topic in finding new methods crucial to the physical realization of this *formal complexity*. The use of new software tools to rationalize formal complexity is generally discussed in reference to structural and geometric principles utilized to achieve an efficient assembly of different building components. Within this context, the use of parametric and algorithmic approaches to design rationalization has focused on *pre-rationalization* and *post-rationalization* [1, 2]. In the pre-rationalization method, the building geometry is usually predetermined by a number of geometric constraints set in the early design stage, whereas with post-rationalization, the building geometry is retroactively simplified to accommodate realistically constructible components. The Sage Performing Arts Centre in Gateshead (Foster and Partners) is an example of the pre-rationalized design method. In this case a decision was used to limit the surface to toroidal geometry, as to standardize on a limited set of roof panels. The GLA building in London (Foster and Partners) demonstrates the opposite scenario by post-rationalizing the "egg" form into PQ strips after the final form had been fixed [3]. These two methods essentially present a top-down (post-rationalization) and bottom-up (pre-rationalization) conceptualization scheme where the designer must consider changes in overall form while simultaneously exploring the consequences of different fabrication techniques. Design is a non-linear and open-ended process where a multitude of constraints dynamically converge in support of various design possibilities. For many designers, the early stages of the design process play a key role in the development of innovative design. Thus, rationalization methods are an inherent part of the exploration process of new forms and materials that directly affects the level of innovation achieved in the physical realization of the design. Our contention is that existing rationalization techniques cannot fully address the design challenges of conflicting configuration and fabrication constraints. Therefore, in contrast to pre-rationalization and post-rationalization, we describe *embedded rationality* as a method for implicitly combining fabrication constraints into an interactive framework for conceptual design. Previous research has employed the term 'embedded rationality' in the context of a parametric design environment to describe rationalization during a parameter change [4], however as Kilian describes, parametric environments require a great deal of structuring in the early stages of design [5]. In this paper we employ the term in the context of a physics-based simulation environment to convey the unique scenario where interactive form-finding tools operate directly on buildable surfaces. We present a novel unified physics-solver as a comprehensive and generalized framework where problems (even ones that do not seem to be physical systems) can be expressed as a set of constraints resulting in outcomes demonstrating emergent properties. We identify categories of the computational characteristics of key results and explore interactions of these characteristics that are only made possible by the use of a unified model. Finally, we provide an example of embedded rationality by examining a constraint-based model of the fabrication logic for a panelization system.

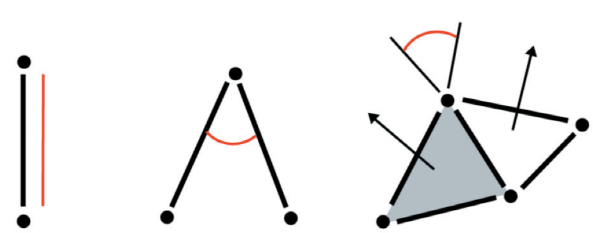
2. A Unified Constraint-Based Solver

We present a new framework for dynamics simulations [6]. The aim of this framework is to simulate the interactions between different objects and substances in a physics plausible manner. Traditionally, solvers are designed to compute the motion of a particular type of object such as rigid bodies, cloth or rope. Combining effects such as a steel post in tension using a rope can be problematic as information has to be transferred between a rope solver and a rigid body solver when contact is made between the two objects. Instead, in our system, all objects are modeled as a simplicial complex: an assemblage of points, edges, triangles and tetrahedra. These are all instances of a k-simplex, a mathematical generalization of the concept of a triangle [7]. As every shape can be approximated to any desired precision with a simplicial complex, this generalization implicitly supports control over the quality of the simulation outcome.



◀ Figure 1. k-simplex shapes used in the unified solver (left to right): point, edge, triangle, and tetrahedron

The dynamics of the framework are governed by a set of particles which correspond to the vertices of the simplicial complex under constraints [8]. By modeling three simple constraints, namely edge length (stretch), angle between two edges (shear), and angle between two faces (bend) (see Figure 2), all meaningful deformations of 1-simplex, 2-simplex, and 3-simplex objects can be represented including torsion and shear.



◀ Figure 2. Three fundamental constraints (left to right): edge length, angle between two edges, and angle between two faces

Material properties such as stretch, bend or shear are all formulated as constraints. For example, the stretch of a material is defined with respect to prescribed rest lengths. We choose this formulation as it is more stable than defining the stretch in terms of springs, for example. Springs are good at modeling bouncy objects but pose challenges when modeling stiff materials such as cloth. For very stiff materials, spring-based systems require very small time steps or fully implicit techniques which result in long simulation times. In our framework, we take the opposite approach where we start with hard links (constraints) and then allow them to be softened when more springy behavior is preferred. This results in faster and more stable simulations. Our solver falls in the category of simplectic integrators where velocities that resolve the constraints are computed implicitly while positions are updated explicitly [9].

Another key feature of our framework is its ability to resolve collisions between objects and self-collisions for deformable objects. We perform the collision detection in space-time for better accuracy. This is necessary for a fast moving object which might be in a valid state at the beginning and at the end of a simulation step but collides sometime midway. In this manner we guarantee that collisions are not missed. Our collision detection uses a fixed time step unlike solvers who treat collisions sequentially in order of their collision times. The latter approach can suffer from lockdowns and high computation times in the event of many collisions.

Collision handling can be seen as another constraint imposed on the system: no objects shall pass through each other. In general, a simplicial object in our system has to satisfy many different constraints at the same time. Sometimes these constraints can be in conflict such as a rubber band under tension between two poles. In this particular case the stretch constraint is battling the collision constraint. In most cases we want the collision to take precedence over the stretch constraint such that the rubber band is under tension. To better handle novel goals, the user can establish a preferred order of evaluation of the constraints. Rather than trying to solve each constraint one at the time, the solver interleaves them over a single time step. For each constraint, an importance weight is also assigned which determines how many times an attempt will be made to solve that constraint within each time step.

Complex emergent behavior occurs naturally. After adding air lift and drag constraints, for example, the flapping behavior of a piece of fabric emerges naturally due to these two constraints battling the stretch constraint. The air drag stretches or compresses the cloth which creates forces due to stretch. In this manner one can simulate complicated behaviors even with a very simple unidirectional wind model. Our general philosophy is to keep the basic solver steps as simple as possible and let complex behavior emerge from these simple components: complexity out of simplicity.

3. Dynamic Simulation

In recent years there have been a growing number of investigations where dynamic simulation is introduced as an integral part of the form finding process [10]. This growing interest in applying digital simulation for conceptual form-finding follows the earlier adaptation of animation tools as a dynamic framework for generative design. As Burry indicates, architects historically have been interested in using animation tools to change the static nature of design into a generative process of progressive formation and mutation [11]. However, the idea of animation as simulation has generally been limited to representational exploration of dynamic systems rather than an actual simulation of parameters responding to dynamic, material and variable contextual forces over time. Furthermore, traditional animation techniques are kinematic, that is, their composition is defined geometrically and their motion is defined through prescriptive trajectories since objects cannot interact with one another or with external forces [12].

In contrast to animation techniques, a typical simulation process involves a well-defined model for analysis, synthesis and evaluation. In the conventional design method, this process has been widely adapted to address a range of optimization problems pertaining to structural and material properties of a system. However, by keeping the design model independent from the simulation model, conventional design methods limit the designer's ability to benefit from simulation as part of the design process. A survey of existing literature indicates an ongoing attempt to integrate design and analysis as part of a performance-based generative framework. Shea et al. describes such a framework for the design of a stadium roof truss [13]. This research combines associative modeling and structural performance evaluation to address an exploration of discrete structural form in relation to various performance-related factors. Similarly, Schein and Tessmann present an integrated framework for "structural analysis as a driver in surface-based design", which involves the design of a free-form surface informed by a network of constraints [14]. These two methods enable a rapid feedback loop between the design and analysis; however, in both cases design and simulation still remain as two separate processes as there are no direct interactions between the design model and evaluation process.

The development of an integrated rationalization system for form generation could significantly improve or accelerate design outcomes. Research in human computer interaction (HCI) has shown that certain cognitive problems are more quickly perceived and solved by visual inspection of alternatives than by mentally planning and performing transformations, and choosing a candidate scenario [15]. This phenomenon is referred to as *epistemic action*: offloading mental tasks to the visual system to improve human performance. The ability of this approach, applied to the domain of interactive physics-based simulation, to parallel real world physical characteristics reduces the early need for abstract procedural and hierarchical development referred to as "designing the design" [16], or in other words, building the parametric design space in advance of evaluating specific designs. Furthermore, epistemic action is a way to "augment the cognitive process" [17] to better support the intuition and spontaneity needed in early design speculations of material and form.

Using dynamic simulation as an interactive framework for design is not without precedent. In general, physically-based modeling and related optimization techniques, as a means of geometric interaction, has been a topic of interest in computer graphics for some time [18]. Physics-based simulation dates back to the late 1980's focusing mainly on spring-based models for deformable matter [12]. Within the architectural context, the use of constrained dynamics simulations for interactive geometric modeling was described and used by Gleicher and Witkin to support 2d drawing applications [19]. More recent precedents include Kilian's experimentation with particle-spring systems inspired by the Antonio Gaudi hanging model [20]. However, each of these projects is a singleton solution where a specific simulation solver is created for a specific physical phenomenon, inherently integrating only a limited set of parameters into the form-finding process. In this section, we present a number of previously known simulations but expressed as simple sets of constraints, so that they can operate within a larger unified solver, enabling the possibility of new approaches to exploring a design space. We broadly polarize our classification of simulations as Collision-based or Equilibrium-based. Of course, when both classes are in play, we can achieve more complex emergent behaviors.

3.1. Collision

Collision refers to the calculation of forces at the point of contact among various elements in the simulation. It involves the momentum transfer at the point of contact interacting with material properties to deform and displace objects. We describe *Draping*, *wrapping*, and *bounded growth* as prime examples of collision physics-based results.

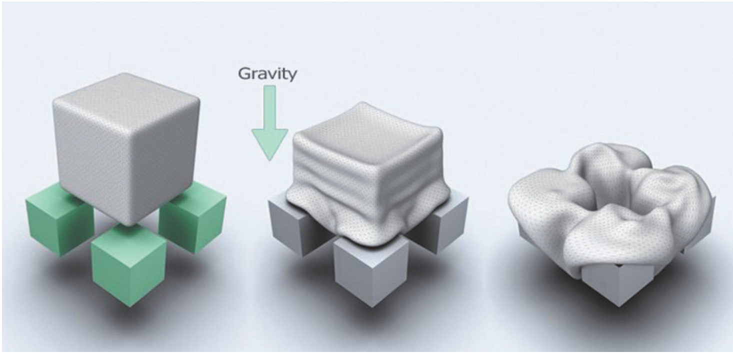
Drapery

The motif of drapery is one of the distinct characteristics of theory and practice in contemporary architecture. In the context of digital design, new advancements in digital processes have helped architects such as Frank Gehry to explore new forms of surface expression inspired by drapery [21]. Gehry's design exploration is however set as an analog between the physical and digital model where physical models of draped surfaces are required to be digitized for further investigations. Simulation could provide an alternative to alleviate the physical interim process with virtual draping which could, perhaps, result in more varied outcomes.

In the example below (see Figure 3), a rounded cloth cube, with a high level of tessellation, is dropped under gravity onto four rigid cubes. The resulting deformations of the soft cube yield an organic structure that would be difficult to prototype physically.

Wrapping

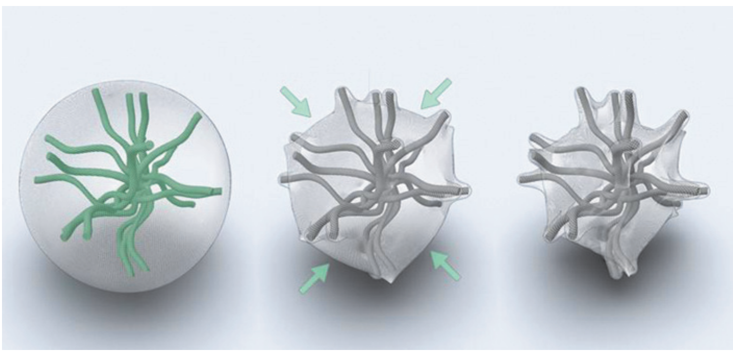
Wrapping provides a conceptual model for skinning an intended object. In a way, wrapping is analogous to a fit fabric around a body of organized data.



◀ Figure 3. A soft cube draped over rigid cubes (left to right): initial condition, collision due to gravity and resulting deformation, and final shape

For instance, an arrangement of structural framing, or a collection of particles representing a flow of architectural programs, could be set up to create an envelope that wraps around them [22].

A shrink film can be made to shrink in one direction (unidirectional or mono-directional) or in both directions (bidirectional) along an initial surface that surrounds the structural frame. To achieve this effect in our solver, the rest length between vertices is set to zero or some progressively minimal value to gradually bring an initial surface into contact with the frame over time (see Figure 4). Collision of the surface with the frame will repel the surface and in time produce a shrink wrap. An air pressure constraint can also be used to aid the surface in better conforming to deep concavities in the frame by setting pressure inside the enclosing shrink surface to zero with normal pressure on the outside. Additionally, drastically different results can be explored by varying the shape and tessellation of the initial shrink surface.

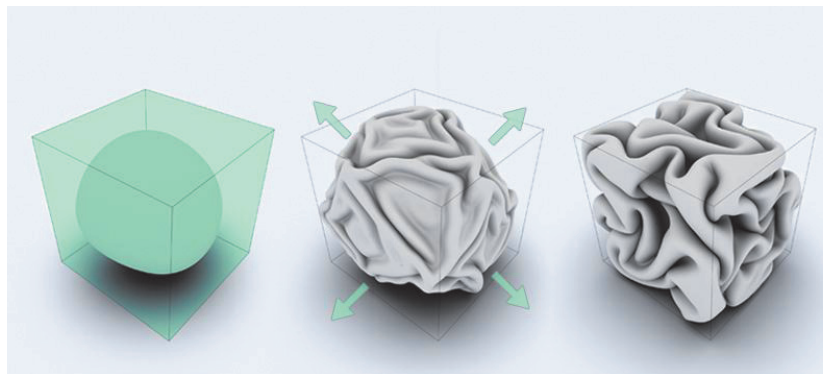


◀ Figure 4. Malleable surface conforming to an underlying rigid structure (left to right): initial condition, collision due to shrinkage and negative internal pressure

Bounded Growth

Bounded growth is similar to the shrink-wrap process involving both an interior and an exterior shape. However, in this method, we reverse the relationship of these shapes and the surface area of the envelope is increased while contained within a boundary constraint.

To achieve this, a surface made up of cloth like material is placed inside a closed rigid bounding container. The rest length of the surface in a given direction is gradually increased until the surface begins to collide with the enclosing container and with itself. Over time, corrugations, bends and folds can occur to accommodate the increased surface area of the surface inside the container. This method could also be combined with some changes in material properties to allow sharp angular folds to develop (see Figure 5).



► Figure 5. Growing surface bounded by an enclosure (left to right): initial condition, collision due to expansion and final shape

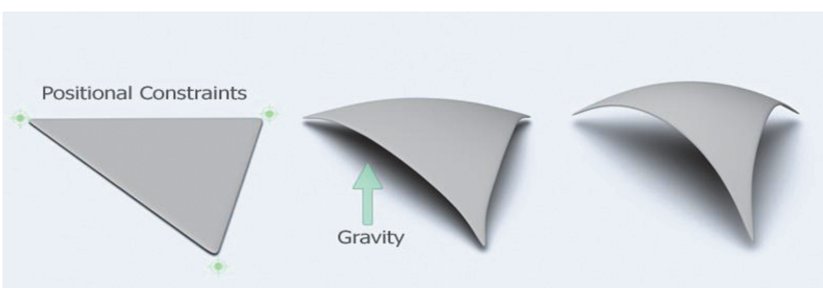
3.2. Equilibrium

Equilibrium is the tendency for a system to achieve a stable balance between internal influences within that system. For instance, in designing fabric or grid shell structures designers aim to achieve an equilibrium position under the influence of loads by using computational methods such as dynamic relaxation. Relaxation is essentially a natural process that minimizes the potential energy in a system as that system tends towards equilibrium. The design of the British Museum Roof exemplifies this method by iteratively solving for the propagation of forces between all the nodes in the system [23]. Dynamic relaxation is typically applied when the overall form has already been fixed. A physics-based approach, however, opens up the possibility of using multiple sets of constraints with properties that would allow behaviors such as tension or compression to emerge as a form finding mechanism. Generally, the initial system is not in equilibrium before the simulation is started. After simulation begins many physical changes can be observed as elements in the system interact and change to achieve equilibrium. Observed changes in the system can also be captured during the process as starting points for other processes. The simulation can be run until convergence or until a final state of equilibrium is achieved. In the case where a valid equilibrium state cannot be found, the simulation normally oscillates between different states in perpetuity.

During simulation, designers can also interact with the elements of the simulation changing the outcome and the possible states of transition. These changes may provide a vast number of design variations. Below we describe a number of key methods based on the notion of equilibrium.

Gaudi Paradigm

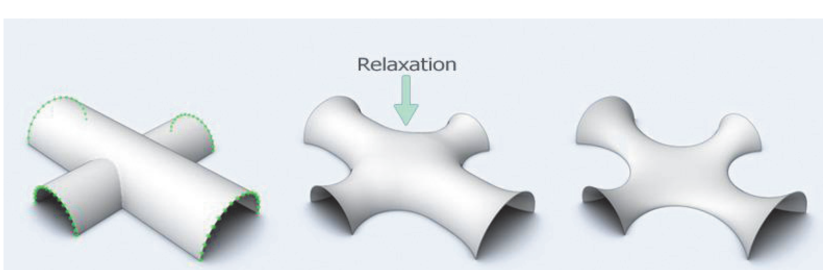
This paradigm refers to a classic method of structural form finding where form is defined through a translation of gravitational force. Antonio Gaudi's hanging chain models are the best known examples of using this scheme in which a building is modeled in tension under reverse gravity to define the form of the compression structure (under normal gravity). While this method has been previously explored, by making multiple physical models, a similar set up can be created as a real-time simulation [20]. By applying positional transform constraints to vertices of a planar surface and raising them to a given height during simulation, a tent like structure will emerge. Similarly, groups of nodes can be constrained to form creases of various shapes. As tension propagates through the fabric under motion, waves can form in the cloth until gravity and damping dissipate them allowing the system to reach equilibrium. In the example below (see Figure 6), a triangular piece of virtual cloth, that is pinned at the corners, stretches under the effect of reversed gravity. Varying material properties such as stretch, shear, rigidity and the bending between surface sub-elements can change the shape and nature of the resulting structure.



◀ Figure 6. Gaudi Effect (left to right): negative gravity stretches a surface

Minimal Surface

When a catenary curve is rotated about an axis, it creates a minimal surface area for the bounding circle called a catenoid. This can also be approximated using cloth and gravity under our solver. The structure shown in Figure 7 was



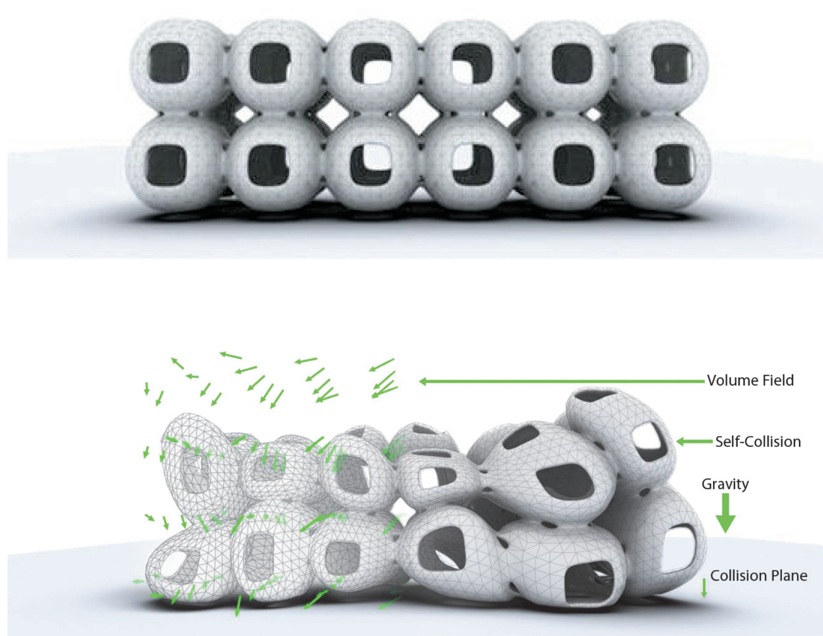
◀ Figure 7. Surface minimization (left to right): rest length reduction created by intersecting, merging and smoothing two open cylinders. Positional constraints are added to the end annuli of the cylinders. The rest length is then scaled down for all elements of the material, putting the entire surface in tension and allowing it to shrink. Sufficient stretch sub-steps are used in the simulation to avoid excessive non-uniform deformation.

3.3. Freeform Finding Using Interacting Elements

Previously, we discussed collision and equilibrium separately. We now examine more complex scenarios where these classes interact and, furthermore, volumetric or logic-based constraints are involved in the simulation. A constraint-based conceptual design process can further be extended as the designer sees fit. In addition to each method described in previous sections, we can combine various methods to allow more complex behaviors to emerge. In the example below, a set of spheres with cut out areas are initially placed in a grid pattern. Using particle dynamics, a volumetric varying torsional force field is applied to the particles which are the nodes of simulated cloth. The simulation adds material properties and realistic deformation by colliding with a fixed ground plane. The interplay among all the internal material forces, collision and the torsional force field cause the entire structure to deform almost organically with dramatic effect.

As the force field dissipates, the form settles to a stable state. These force fields could represent certain contextual conditions that are not strictly physical. Thus, simulation can be used not only to generate forms but also to produce the 'spatial coding of information' [24].

▶ Figure 8. Interacting elements (top to bottom): deformation through an interplay between the internal material forces, collision and torsional force

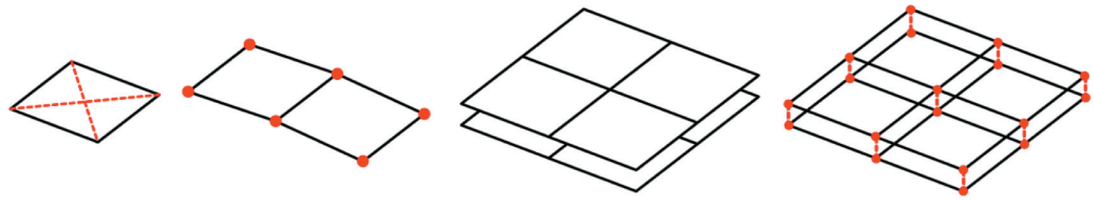


4. Embedded Rationality

Freeform architecture based on doubly curved surfaces is technically difficult and costly to directly realize as a physical artifact. Panelization is a technique to enable such a surface to be constructed from a series of smaller, simpler components. There is a considerable advantage if the panels are planar, since this enables the panels to be made from a standard material such as glass. Conventional planar panelization using triangular facets can be fitted to complex surfaces, but at each node six panel edges must be connected, which introduces additional fabrication complexity [25]. Furthermore, triangle meshes do not support offsets at constant distance in a multilayer structure [26]. By using quadrilateral panels we can simplify the connections. However it is non-trivial to define the set of planar quads (PQ) for a given surface, where each set of four adjacent panels meet at a common point (or structural node). In addition, because the sheet material (such as plywood) has a defined thickness, it is also important that the offset quads of each four adjacent planar quads also intersect at a common point [25]. Thus the full definition of the implementation constraint is that the design surface has to be decomposable into Planar Offset Quads (POQ).

In this example we explore a freeform surface design driven by a POQ mesh principle. While this class of surface has been previously explored as a mathematical optimization (post-rationalization) of a fixed surface [26, 27] we are interested in exploring POQ meshes as a guiding principle of dynamic surface generation. Instead of approaching POQ meshes as an optimization problem, we embed their rationale within a flexible and iterative design process. Therefore, a freeform surface is defined as an emergent set of relationships among simpler components.

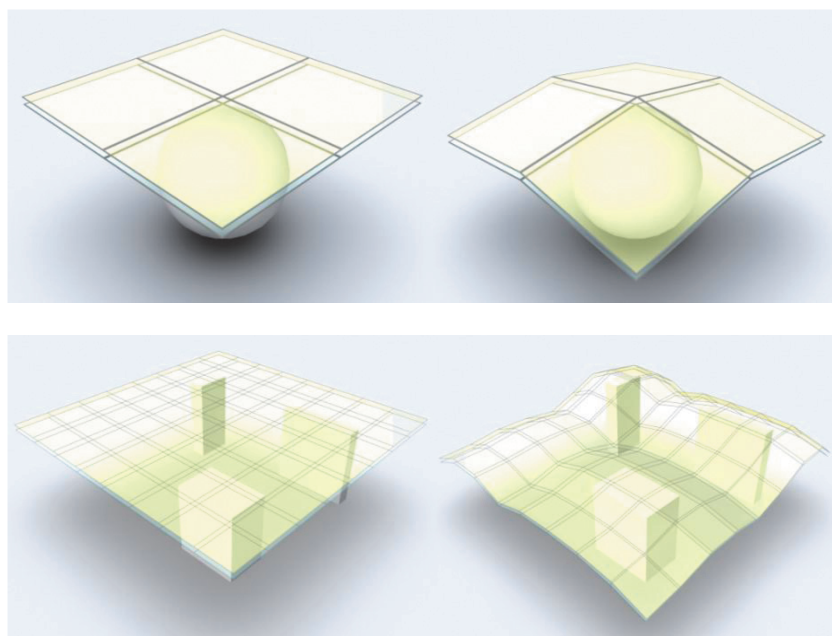
We begin the process by establishing the surface as a simulation of singular panels. By simulating the actual panels our system guarantees a constant offset within a numerical tolerance using collision between surfaces and constraints between points. As mentioned, material properties such as stretch, bend or shear are all formulated as constraints. Therefore, we apply the principles of the POQ mesh as constraints that define the inherent properties of panels. To assure planarity, each panel is essentially treated as a 3-simplex shape where the angle of two faces (bend) is minimized through cross bracing, see Figure 9 (a). After setting the material property of the panels we establish a set of relationships among the panels in order to define the overall behavior of the surface system. These relationships are defined through two sets of constraints. One set of constraints welds all the panels together while allowing each panel to pivot around its border, see Figure 9 (b). After offsetting the surface, Figure 9 (c), the second set of constraints is applied as a distance constraint between the surface and its offset, thereby emulating the thickness of the panels. The distance constraint allows the offset surface to slide while maintaining a constant offset value from the original surface, see Figure 9 (d).



▲ Figure 9. (left to right). (a) Planarity (build PQ face) by adding bend constraint. (b) Coincident vertex constraint to build "surface" from PQ face. (c) Creating an offset PQ mesh. (d) Creating POQ mesh by adding distance constraints

► Figure 10. (top to bottom): A basic POQ mesh interactively draped on a collision object. Using POQ mesh to interact with larger surfaces

Once these connections have been established, we can manipulate the surface, either through pushing and pulling of nodes, or with other collision methods described earlier in this paper. Figure 10 illustrates the result of our simple set up consisting of 4 panels draped on top of a collision object. Unlike a typical simulation process which requires a well-defined model to converge at an optimum solution, we present a stable numerical model aiming at a more iterative progression but with fast results. These results represent light-weight conceptual models that can be further refined in the later stages of design.



4.1. POQ Analysis

As described above, the behavior of our POQ system is defined through a set of interacting constraints. This behavior attempts to maintain the properties of a real manufacturable panel. In terms of geometry, there are three aspects of a panel's shape that must be maintained within a minimum acceptable level of error or tolerance. First, the top and bottom elements must be planar. Second, the distance between the top and bottom elements over the area of a panel element must remain consistent according to a specified value. Finally, the interfaces between the panels or the cutting edges of the panels must also be planar with no gaps between panels.

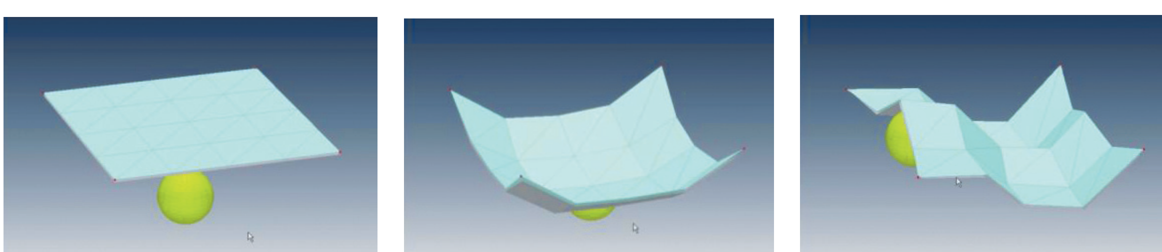
The units in our physics-solver are all non dimensional values and must be mapped to real world values for comparison. For our purposes, we have chosen various tolerances, all less than +/–2 mm for every meter of length.

In order to determine the quality of the solution provided by the embedded rationality system, the sources of possible error need to be considered. In overview, there are three categories for sources of error:

1. Numerical and convergence errors with respect to solving system constraints.
2. Numerical errors due to scaling of the input geometry and digital representation errors.
3. Iterative error or convergence errors as the geometry system changes.

Given the scope of this paper, we will only consider the iteration error as it pertains to the geometrical layout of the panel. Iterations in this sense correspond to solved or converged states as calculated by the physics-solver, more akin to frames in an animation than calculation iterations. For every one of these system iterations, the physics-solver will have iterated many times over all calculations to satisfy the solver's own internal convergence criteria and error minimization schemes.

To study these potential errors, a simple model is simulated to demonstrate a sequence of real-time interactions with POQ mesh within our framework. The model consists of a 4 x 4 grid arrangement of panels with all four upper surface corners fixed in space, see Figure 11 (a). These four anchor points essentially represent a design constraint by fixing the free-form surface at four imaginary posts. A uniform force in the downward



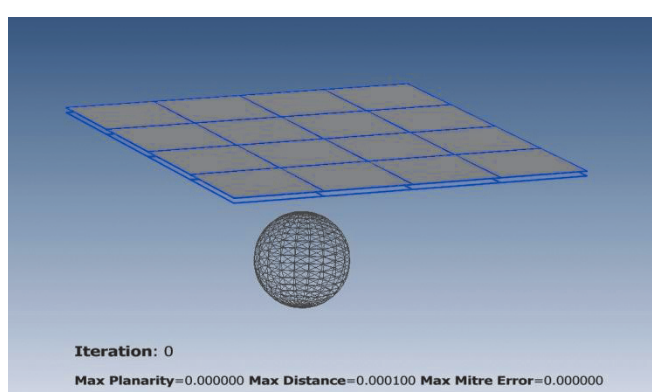
▲ Figure 11. (left to right): (a) A POQ mesh set up in initial state. (b) POQ mesh is relaxed under gravity. (c) Interactive manipulation of POQ mesh using a passive collider

direction is given to simulate gravity and a slightly off centre sphere is used as a passive collider over which the surface will drape, see Figure 11 (b). Once the drapery has reached a stable state, we begin to manipulate the surface directly in real-time by using the sphere as a collision object, see Figure 11 (c). Referring to examples in previous sections, we can combine additional constraints and forces during the simulation as a method for affecting the POQ surface. For example, we can apply the gravity in a negative direction while adding wind drag and torsion to the overall system. Furthermore, we can stop the simulation at any given time in order to sample the geometry as a possible design alternative.

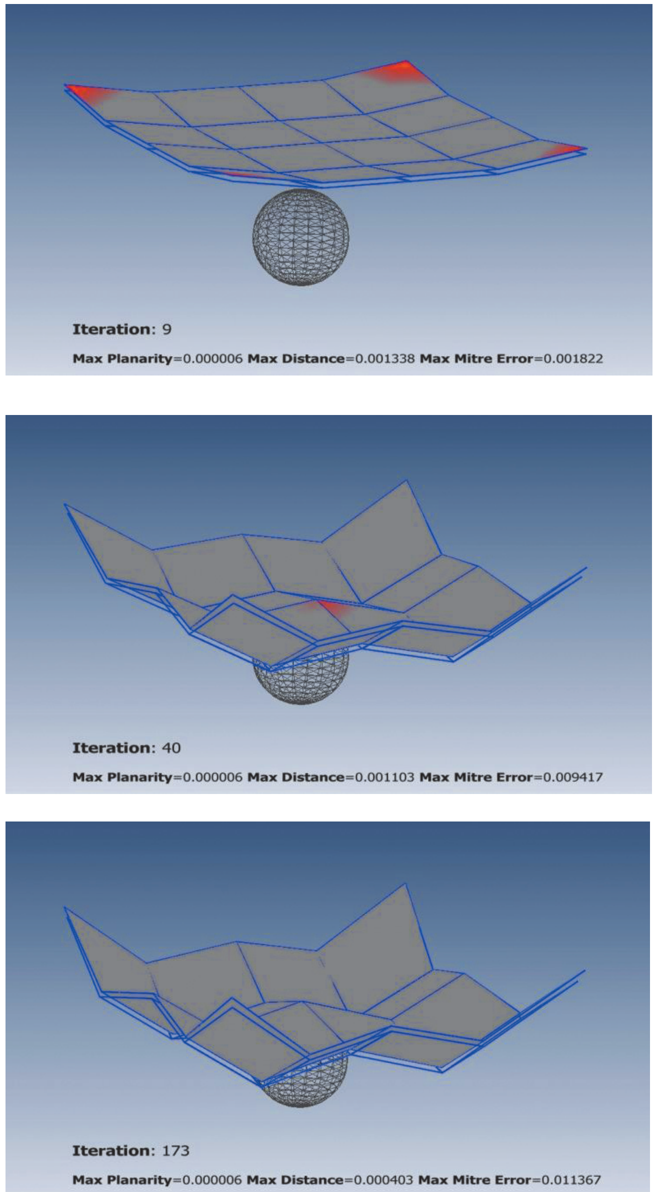
To evaluate the geometric errors mentioned earlier, we need to perform a set of three simple calculations. First, we measure the planarity of both the top and bottom panels by calculating and normalizing the face of each of the triangles that compose each quadrilateral. The dot product between these normals will yield a proportion of the length of the unit normal proportional to the cosine of the angle. The error is one minus the dot product of the unit vectors which is a proportion of the length scale. To determine errors in panel thickness, we measure the distance between each corresponding top/bottom pair of triangles that compose the panel. Then, the error is the distance minus the target thickness of the panel and is in units of length. The connecting interface between each panel or miter joint between them should be planar as well to facilitate manufacturability. Because of the way the connections between panels are specified and due to the nature of the links interconnecting the panels, we will not evaluate the error between connections and treat the interface between panels as a single miter joint. Using the two topmost vertices of an edge of a panel and a single vertex of the corresponding edge on the bottom panel, we fit a plane. To measure the deviation of the miter joint, we calculate the distance of the remaining point from the plane. This distance yields an absolute value in units of length as a measure for twist in the miter joint.

During simulation, maximum error values over the entire panel system are computed and displayed numerically at the bottom of each display frame for every iteration. To visualize the location of the errors, a gradient color value is applied through texture shading. Values of error are linearly interpolated from the vertices across the face of each triangle. In the shaded images, values within tolerance are displayed in grey color. Values above tolerance are shaded in red with the maximum shade corresponding to the maximum specified tolerance (see Figure 12).

In this example, the panels are .8 x .8 length units with a thickness of 0.01. For panel thickness, the acceptable tolerance was specified as 0.00005 units of length while the maximum was set to be 0.001 units of length. For



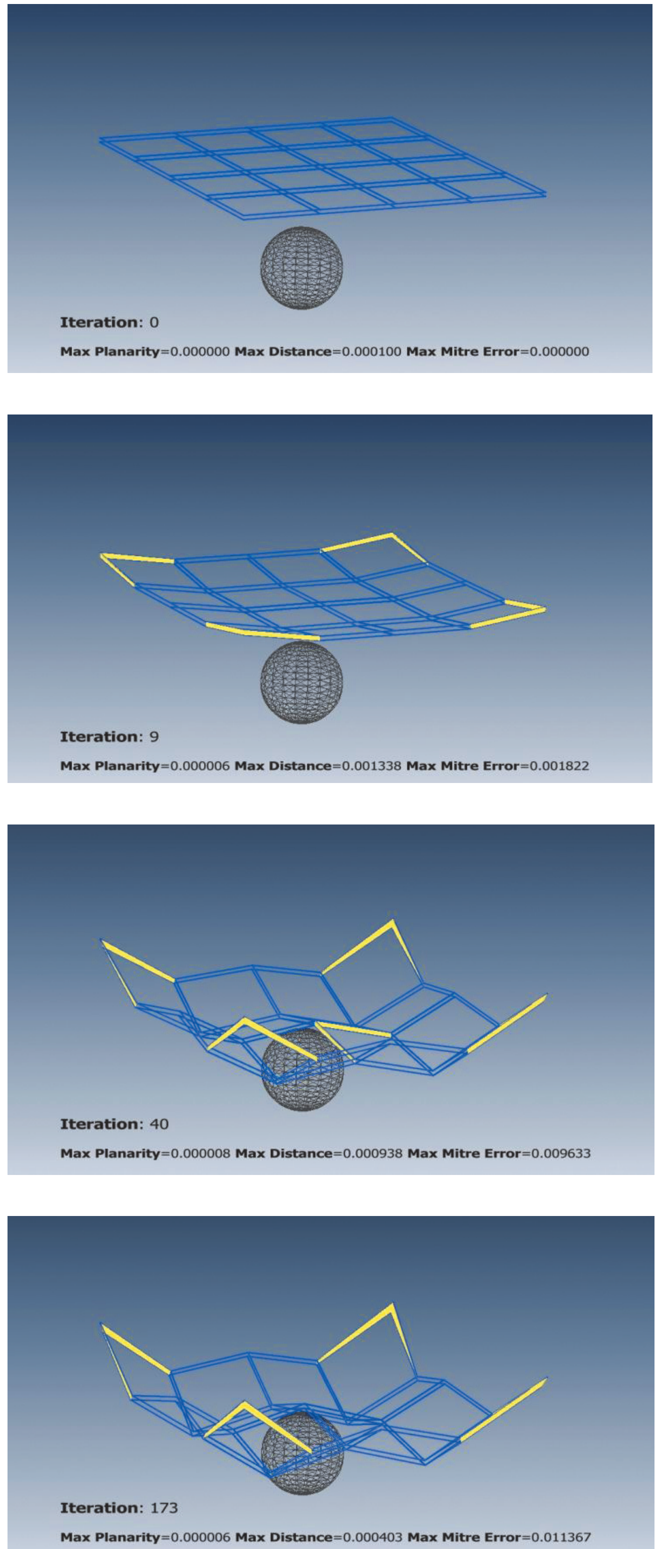
► Figure 12. Visualization of planarity errors (shown in red)



◀ Figure 12. *(continued)*

a 1 meter square panel, these tolerances correspond to 0.06 mm and 1.25 mm respectively. At iteration 173, the maximum distance error between panels was 0.000403, which corresponds to an offset of .5 mm for a 1 meter panel.

Miter error tolerances were specified as 0.001 units of length. This corresponds to a deviation of 1.24 mm for a 1 meter panel. By examining the images in the series showing miter error (see Figure 13), we can see that it is predominantly the outside edges that fall outside of tolerance. This is due to the fact that only the top vertices of the panels were pinned in space. By pinning both sets of vertices, top and bottom, this error can be



▶ Figure 13. Visualization of miter errors (shown in yellow) avoided and similar results obtained. Even in this case, the miter error for a 1 meter panel corresponds to approximately 14.2 mm. The planarity error of the top surfaces range between 0.000006 and 0.000008 throughout the iterations but are never truly zero. The deviation is very slight and corresponds to 0.006 and 0.008 mm for a 1 meter panel.

In general, the presence of errors is transient and temporary. The system naturally relaxes to a good solution if it is able to find one. Errors that are out of tolerance in both panel thickness and panel joints appear initially after the panels collide with the sphere. Over multiple iterations, these errors disappear. If a transient state of the panel system is desired, we believe that the system can be put in a state where relaxation could occur thereby reducing or eliminating the errors with only minor shape variations. For arrangements where there is no solution, the solution generally oscillates between two saddle states in perpetuity, with similar values of error. This error can be evaluated and if found to be too large, parts of the POQ mesh can be manipulated with extra positional constraints or manipulated interactively and massaged into a system that can achieve convergence.

In our example, convergence is obtained after approximately 173 system iterations. Panel thickness and planarity errors are within tolerance on the interior of the panel. The overall behavior of the surface can be characterized as a balance between precision and the degree of freedom. By pinning both sets of corner vertices, both top and bottom, we believe that all tolerances can be met.

5. Conclusion

Simulation has already allowed architects to pursue novel approaches to a design problem. However, the idea of constraints-based simulation as a dynamic framework for form-finding introduces new possibilities in how we can rationalize our design explorations. A primary goal in the development of our approach has been to provide a high-level framework that can unify various design constraints into a single model of generative design. Given the flexibility and generality of our framework we hope to explore additional classes of real-world constraints as part of an approach to simulation-based form-finding. In contrast to previous methods, we have presented a system that circumvents geometrical rules and abstraction by using a unified constraint framework for dynamic simulation. Furthermore, we have demonstrated interactivity as a key component of our simulation framework where resulting effects can be produced in real-time. We have shown an instance of our approach by directly modeling the physics of the panel component, while establishing our fabrication constraints as embedded rationality and the genesis of form exploration. Unlike typical analytical simulation which aims at an optimum solution based on a well-defined model, our framework presents a stable numerical model providing dynamic iterations, flexibility and faster results.

References

1. Fischer, T., *Enablement or Restriction?*, CAAD Futures Conference, 2007, 585–598.
2. Schlueter, A. and Bonwetsch, T., *Design Rationalization of Irregular Cellular Structures*, International Journal of Architectural Computing, Vol. 06, 2008, 197–211.
3. Whitehead, H., Laws of form, in: Kolarevic, B., ed., *Architecture in the Digital Age-Design and Manufacturing*, Spon Press, New York and London, 2003, 82–100. Jos Stam. (2009).
4. Fischer, T., *Rationalising Bubble Trusses for Batch Production*, Automation in Construction, Vol 16(1), 2005.
5. Kilian, A., *Design Exploration through Bidirectional Modeling of Constraints*, PhD Thesis, Massachusetts Institute of Technology, 2006, 54–55.
6. Stam, J., Nucleus: Towards a Unified Dynamic Solver for Computer Graphics, IEEE International Conference on Computer-Aided Design and Computer Graphics, 2009, 1–11.
7. Alexandroff, PS., *Elementary Concepts of Topology*, Dover Publications, New York, 1961.
8. Arnold, VI., *Mathematical Methods of Classical Mechanics*, Springer-Verlag, New York, 1989.
9. Hairer, E., Lubich, C. and Wanner, G., *Geometric Numerical Integration*, Springer-Verlag, Berlin, 2002.
10. Oxman, R., *Performance-Based Design: Current Practices and Research Issues*, International Journal of Architectural Computing, Vol. 06, 2008, 1–17.
11. Burry, J, Felicetti, P, Tang, J, Burry, M and Xie, M., *Dynamic Structural Modeling*, eCAADe Conference, Denmark, 2004, 312–317.
12. Terzopoulos, D., Platt, J., Barr, A. and Fleischer, K., *Elastically Deformable Models*, SIGGRAPH,1987, 205–214.
13. Shea, K., Aish, R. and Gourtovaia, M., *Towards Integrated Performance-Based Generative Design Tools*, eCAADe, Austria, 2003, 553–560.
14. Schein, M. and Tessmann, O., *Structural Analysis as Driver in Surface-Based Design Approaches*, International Journal of Architectural Computing, Vol. 06, 2008, 19–39.
15. Kirsh, D., and Maglio, P., *On Distinguishing Epistemic from Pragmatic Actions*. Cognitive Science. Vol. 18, No. 4, 1994, 513–549.
16. Burry, M., Between Intuition and Process: Parametric Design and Rapid Prototyping, in: Kolarevic, B., ed., *Architecture in the Digital Age-Design and Manufacturing*, Spon Press, New York and London, 2003, 149–162.
17. Aish, R., *From Intuition to Precision*, The 23rd eCAADe Conference, Portugal, 2005, 10–14.
18. Harada, M., Witkin, A. and Baraff, D., I*nteractive Physically-based Manipulation of Discrete/continuous Models*, SIGGRAPH, Los Angeles, 1995
19. Gleicher, M. and Witkin, A., *Drawing with Constraints, The Visual Computer*, Springer-Verlag, New York, 1994, 39–51.
20. Kilian, A., *Linking Digital Hanging Chain Models to Fabrication*, Proceedings of AIA/ACADIA, Cambridge and Toronto, Canada, 2004, 110–125.
21. Allmer, A., *The Architecture of Gigantic Drapery*, The Richard B.Fisher Center for the performing arts, Bard College, theory, Vol. 11, 2007, 265–275.

22. Ophir, Y., *Go with the Flow: Particle Systems for Program Modeling in High-Rise Buildings*, International Journal of Architectural Computing, Vol. 06, 2008, 172–196.
23. Williams, C., The Analytic and Numerical Definition of the Geometry of the British Museum Great Court Roof, in: Leach, N., Turnbull, D. and Williams, C., ed., *Digital Tectonics*, Wiley-Academy, United Kingdom, 2001, 78–85.
24. Franken, B., Real as Data, in: Kolarevic, B., ed., *Architecture in the Digital Age-Design and Manufacturing*, Spon Press, New York and London, 2003, 122–138.
25. Cutler, B and Whiting, E., *Constrained Planar Remeshing for Architecture*, Graphics Interface, Montreal, Canada, 2007.
26. Pottman, H., Schiftner, A. and Johannes, W., *Geometry of Architectural Freeform Structures*, ACM Symposium on Solid and Physical Modeling, Internationale Mathematische Nachrichten, Vol. 209, 2008, 15–28.
27. Zadravec, M., Schiftner, A and Johannes, W., *Designing Quad-Dominant Meshes with Planar Faces*, Eurographics Symposium on Geometry Processing, Vol. 29, 2010.

Ramtin Attar, Robert Aish, Jos Stam, Duncan Brinsmead,
Alex Tessier, Michael Glueck and Azam Khan

Autodesk Research
210 King Street East, Toronto,
Ontario, Canada M5A 1J7

Ramtin.Attar@autodesk.com