Material-based Design Computation An Inquiry into Digital Simulation of Physical Material Properties as Design Generators

Neri Oxman and Jesse Louis Rosenberg

Material-based Design Computation An Inquiry into Digital Simulation of Physical Material Properties as Design Generators

Neri Oxman and Jesse Louis Rosenberg

The paper demonstrates the association between geometry and material behavior, specifically the elastic properties of resin impregnated latex membranes, by means of homogenizing protocols which translate physical properties into geometrical functions. Resin-impregnation patterns are applied to 2-D pre-stretched form-active tension systems to induce 3-D curvature upon release. This method enables form-finding based on material properties, organization and behavior. Some theoretical foundations for material-computation are outlined. A digital tool developed in the Processing (JAVA coded) environment demonstrates the simulation of material behavior and its prediction under specific environmental conditions. Finally, conclusions are drawn from the physical and digital explorations which redefine generative material-based design computation, supporting a synergetic approach to design integrating form, structure, material and environment.

1. Introduction

1.1. Background

"What does a brick want to be?" In his philosophical explorations the architect Louis Kahn proposed that buildings were not inert configurations of form but "living organic entities" [1]. By reinterpreting the tectonics of structural processes Kahn sought to postulate a universal ordering system whereby function had to accommodate itself to form insofar as form was the result of a profound and timeless understanding of the task it had to support. Let us replace the word "brick" with the notion of "material", and the concept of "task", or "function", with that of "performance". Now let us speculate on how does a material perform? Moreover, is there a way in which we could *predict* material behavior and organization within a given context? How do we *find* material form?

Pursuing further this strand of logic, the notion of *form-finding* attributed to the great Frei Otto [2] follows from Kahn's conviction of a predetermined search for material form, but produces perhaps a more refined account with regards to the synergetic relationship between performance and material integrity. In much of Otto's work, specifically in his early experiments from the 50's, types of structures are systematically classified and form is the result of load application. Of particular interest in this respect are Otto's membrane structures and pneumatic structure experiments that promote the formation of "minimal surfaces" which optimize structural loads.

This present research seeks to build upon design work based on physical form-finding and to extend this experimental research tradition to the inquiry into what the implications of such experiments may be *when translated from the physical to the digital realm*. How, and indeed when, does *digital matter* transcend its representational value and acquire ontological, operative and, even, generative validity for the designer in his creative search for formal, structural and material integrity.

1.2. Problem definition

Current CAD applications, including associative modeling software packages, appear frequently to promote generative approaches to design [3]. Rather than treating the computational media merely as an "output station" prior to production, the designer is now able to establish parametric relationships between features, methods and/or functions in a way which supports design processes of an exploratory nature [4]. However, this liberation which seems to be manifesting itself across the board throughout the continuous phases of the design process is currently mainly driven by geometrical constraints. Generative performative modeling approaches have been introduced which engage principles of engineering with from-finding [5]. And yet, even when integrating performance factors and tools that are significant in determining architectural form [6], material organization and behavior are already predetermined design constraints; predetermined factors. Form finding, in the digital realm, is thus restricted to the relationship between structure and geometry (and/or fabrication); it does not generally incorporate, and/or support, the expression of material properties, organization and behavior.

1.3. Aims and objectives

This work seeks to establish a *synergetic* approach to design whereby material organization and behavior, as they may appear in the physical world, may be integrated into digital tools for design exploration. The approach is based on the premise that *material, structure, and form* can become inseparable entities of the design process which relate to *matter, performance and geometry* respectfully. Beyond this theoretical significance, the goal of the experiments presented here is to effectively link the simulation/computational techniques across adjacent scales of physical behavior so that microscopic level physics and mechanisms are incorporated into the description of properties and behavior at the mesoscopic (micro-structural) level, and beyond that, in order to suggest descriptive attributes even at a macro scale. The paper presents some first steps in the creation of a digital simulation tool developed in *Processing* (JAVA environment) which is informed by material behavior and targeted towards material-based design generation.

1.4. Organization

The paper is organized in four major parts: the first chapter includes a problem definition statement; in the second chapter the theoretical framework upon which the design experiments are based is formulated; the third chapter describes and demonstrates a set of computational tools that have been developed to enrich and potentially to redefine the content and characteristics of current CAD applications, and, finally, the forth chapter concludes with speculations on directions for a paradigmatic shift in design towards what has been demonstrated as material-based computation.

It is clear that beyond the ever growing capabilities of digital media to generate form there is a predominant need to be able to model and simulate material properties and behavior in the process of finding and evolving form. The experiments reviewed and the tools developed here focus on specific materials and behaviors, but are suggestive of a more general approach to material computation in design.

2. State of the art and problem definition

2.1. Integrated digital and physical form-finding

Some original work has been carried out which promotes the development of digital modelers in Java (and other object-oriented languages) which aim at emulating physical modeling. Among recent examples is Killian's modeling of Gaudi's hanging chain models [7]. This work specifically demonstrates the way in which fabrication strategies may potentially be linked to real-time form-finding simulation in that it seeks to establish an iterative design process which allows for physical and digital integration. Furthermore, Killian has succeeded to incorporate material properties (and behavior) within the framework of real-time digital form-finding while, in addition, strategizing fabrication for physical mock-ups.

Physically-based cloth simulation has been around for more then a decade and still challenges the community of computer graphics when confronted with the need to accurately describe the properties and behavior of physically informed deformable surfaces [8].

Most applications of physically-based cloth simulations challenge both the realistic representation of material behavior and the speed of modeling. This work is targeted towards the former, more than it is attempting to promote a robust number-crunching application. Nonetheless, many energy models (models which incorporate dynamic forces as drivers for material conditioning) were developed for research in physical based modeling. These models are loosely classified into three categories: continuum models, mass-spring models, and particle models [9]. Continuum models which simulate material behavior, such as "Folding&g" consider the equilibrium of a given body subjected to external forces, and are mostly modeled with FEM [10]. Mass-spring models use mesh vertices as an underlying behavior system, and particles systems (such as the one demonstrated by this work) use geometrical primitives which are assigned certain physical properties individually. Some interesting work for fast energy-based surface wrinkle modeling has been presented by Wang, Wang and Yuen [9] which uses a governing curve-driven technique to interpolate surface wrinkles. Other models have been developed for modeling, simulation and behavior predictions in material science for the optimization of processing parameters which are based on computer models with aspects of artificial intelligence such as neural nets [11].

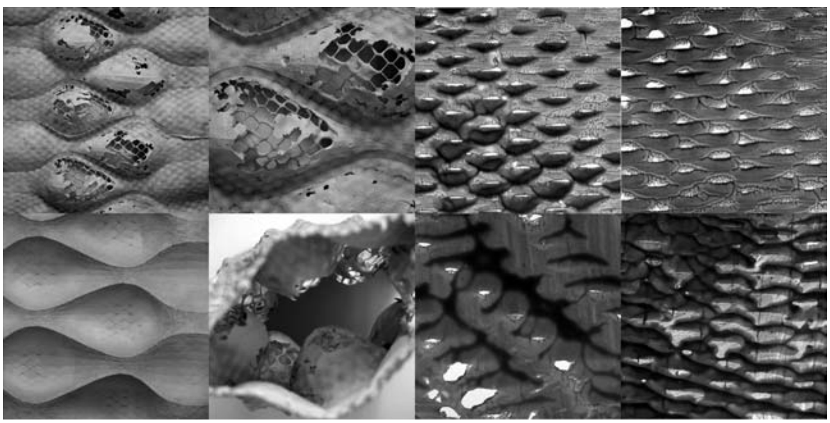
Regarding the forces which are being simulated, most models consider internal: stretch, shear, and bend, and external: gravity and air-drag forces [12].

2.2. Material computation

Material microstructures play a significant role in the mechanical efficiency of natural materials [13]. The organization of cells or fibers determines its mechanical behavior and may be altered to promote efficiency. This present work examines the relationship between geometrical features and material attributes. Given a specific material system, it explores the relationship between the curvilinear development of form, given its loading conditions and the material's ability to stretch. The "curviness" represents the system's geometrical attributes, and its "stretchiness" relates to its material attributes, specifically its elasticity. This exploration has been carried out simultaneously in the physical and the digital realms so as to be able to simulate material behavior.

3. Introductory work: physical experiments

Early studies focused on exploring the relationship between "stretchiness" and "curviness" using physical models. These explorations confirmed that by creating *material composites*, operating as one united organizational entity (as opposed to *material assemblies* operating individually), flexible fabrics may be globally modulated to induce curvature. Figure 1 below demonstrates the early physical explorations of material differentiation in elastic membranes constrained by wire meshes and local resin impregnations (Figure 1, right and left images respectfully). In both studies, the elastic membrane is coupled with additional (locally constraining) material to form a composite which redistributes tension and strain.

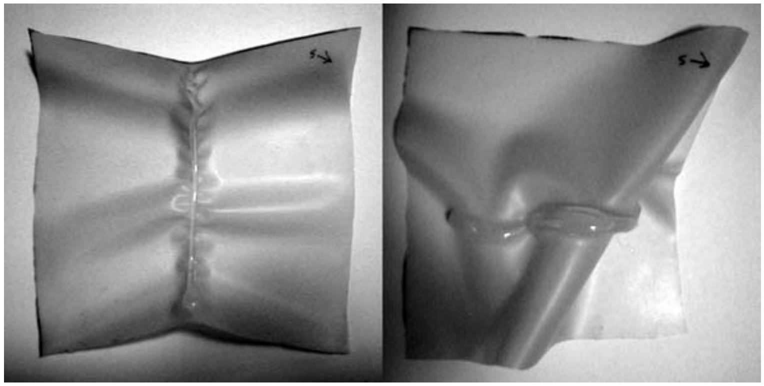


◀ Figure 1: Early physical explorations of material differentiation in elastic membranes

3.1. Physical experiments

As the research seeks to establish the relationship between "curviness" and "stretchiness", strategic decisions were made with regards to material selection. The physical experiments were executed in three groups, each exploring different aspects of the same phenomenon and growing in complexity. Elastic membranes made of rubber latex were selected as a working material.

Experiment 1: The initial experiment (Figure 2) demonstrates the behavior of an elastic membrane when upon pre-stretching, local resin impregnation is introduced to promote non-homogenous material distribution within the membrane (upon release). This phenomenon is in some ways analogous to that of a funicular shape which is one similar to that taken by a suspended chain or string subjected to a particular loading. The impregnated resin is equivalent to "lines of constraint or hardness" which force the membrane to remain at its initial (pre-stretched) length when released, and as a result induce curvature upon release.



► Figure 2: Resin impregnated latex-membrane experiments. Left: Impregnation applied perpendicular to stretch. Right: Impregnation applied parallel to stretch

In this experiment, the latex membrane is stretched in one direction and the resin ("line of force") is introduced, once parallel to the stretch, and again, perpendicular to the direction of the stretch.

Figure 2 illustrates the basic experiment: in the left-hand image, impregnation is applied **perpendicular** to stretch direction and results in local folds along impregnation after release. In the right image impregnation is applied **parallel** to stretch direction and results in global curvature of the fabric after release.

Experiment 2: The next series of experiments (Figure 3) demonstrate the relationship between the resin impregnated patterns applied to the pre-stretched membrane and the resulting curvature produced upon release.

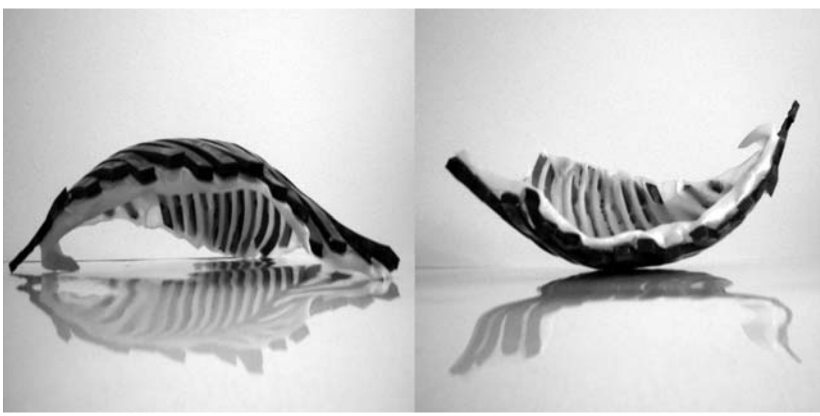
► Figure 3: Resin impregnation process using CNC fabrication methods

In this experiment (Figure 3), the latex membrane is homogeneously stretched on a wooden frame (left and right images illustrate bottom and top configurations of frame); laser-cut chipboard with impregnation pattern is attached to the membrane and resin is applied into the "negative cutting" pattern. Once the resin has dried out, the chipboard is removed and the latex membrane is released from the frame.

Establishing such relations between the impregnation pattern, the direction of stretch, and the resulting geometry would assist in predicting the induced three-dimensional form based on the two-dimensional pattern.

Two models were experimented with which illustrate such relations. The first employed an elliptical pattern and the second, a hyperbolic pattern, both of which proved to be indicative of essentially two different forms of overall global curvatures respectively: the first, synclastic curvature (bowl-like, positive curvature), defined by the property of a surface or portion of a surface for which the centers of curvature of the principal sections at each point lie on the same side of the surface. And the second – anticlastic curvature (saddle-like, negative curvature), defined by having the property of a surface or portion of a surface whose two principal curvatures at each point have opposite signs, so that one normal section is concave and the other convex. By forming a composite material (resin impregnated latex) the impregnated pattern promoted a locally non-homogeneous behavior within this material system.

Figure 4 illustrates the result of such an experiment: the non-homogeneous resin pattern was applied to a homogeneously pre-tensioned latex membrane. Once the stress is released, the membrane deforms completely, based on the 2D impregnation pattern, to form a 3D structure with global synclastic curvature.



◀ Figure 4: Resin impregnated latex-membrane physical experiment (left and right - frontal and dorsal elevation views) demonstrating curvature induced by pre-stretching

4. Simulation system description and implementation

The experiments illustrated here were developed as parallel investigations in the physical and digital realms. Physical experiments were executed which defined the criteria for simulation and further generation of form, based on material behavior under extrinsic loads as determined in the material experiments. The computational tools were developed in the *Processing* environment, an open source programming language developed by Ben Fry (Broad Institute) and Casey Reas (UCLA Design, Media Arts), which enables, among other things, particle system animation.

The term *particle system* is loosely defined in computer graphics. It has been used to describe dynamic simulations of groups of moving objects. A particle system is composed of one or more individual particles. Each of these particles has attributes such as mass, position, and velocity, that directly, or indirectly, affects the behavior of the particle or ultimately how and where the particle is rendered. Often, particles are graphical primitives such as points, or lines, but they are not limited to this. With the Particle System API it is easy to create a group of particles, and then describe the components of the particle effect using *actions* like gravity, bounce, etc. Actions are applied to the particle group at each time step, and then read back as a vertex array or as geometry instances.

In this demonstration the aim was to compute material behavior in two modes: the "top-down" mode promoted the creation of a constraint space emulating the physical system explored. This so-called "space" is defined by limited degrees of freedom (DOF) defined as geometrical constraints driven by physical material attributes. The "bottom-up" mode promoted the application of local, and at certain cases, incremental constraints which were systematically introduced to the "constraint space" to fit specific physical manipulations; more on this later. Before introducing the computational model let us discuss some basic principles of the system at hand.

4.1. Digital implementation and tool development

In the process of converting the physical findings to digital representations, which may potentially become tools for material behavior simulation and prediction, it is essential to determine the way in which 'material behavior' is coded as geometrical features, methods and functions [14]. Both typical cases introduced in the previous section include the perpendicular-to-stretch application of resin impregnation (which resulted in pleating-like cresses formed in the direction of the stretch in the process of compensating for the shortening of the membrane after release with local folds/curves) and parallel to the stretch (which results in a general synclastic and/or monoclastic curviness of the surface as a result of the resin which forces the membrane to fold upon itself upon shrinkage and stress release).

In computational terms this phenomenon of shrinkage vs. stretching must be accommodated by highly articulated "material solvers" that are capable of mimicking material behavior and translating physical properties into geometrical attributes. The computational tool developed for material emulation and material-based form-generation is based on the creation of a *particle-spring engine* built in Java. The fabric is modeled as a set of interconnected particles connected by springs (particles are represented as points, and springs are represented as lines connecting between those points).

Basic physics teaches us that the force applied on a body equals its mass times an acceleration factor:

(1) F (force) = M (mass) * A (acceleration). From this:

(2) A = F/M

We also remember that the force equals the stiffness of the body times its distance:

(3) F (force) = K (stiffness) * D (distance)

The force of a spring therefore equals the stiffness of the spring time the distance from its equilibrium position:

(4) $F = - K * (X - X_0)$

(5) δ (stress) = *E* (modulus of elasticity) * ξ (strain)

A mass *particle p0* – at a given point in time *tn* – has four attributes: its mass - m (defined as a scalar), its position – x, its velocity – v, and its acceleration – a, all being vectorial dimensions. The velocity represents the first derivative of position with regards to the time dimension, whereas the acceleration denotes the second derivative of time. A *spring*, *s0,* connects between two given particles, P0 and P1. It has three attributes which describe it: Its current length – l (scalar), its reference length – l(0) (scalar), and some constant – k (constant). It also has a damping factor – d – which resists motion, v, so the system reaches a stable equilibrium. Given a configuration of any two particles (p1, p2), a spring (s0) at a given time t(n):

A mass particle *p1* – at a given point in time *t (n)* - has:

M	*mass*	*scalar*
X	*position*	*vector*
V	*velocity*	*vector*
A	*acceleration*	*vector*

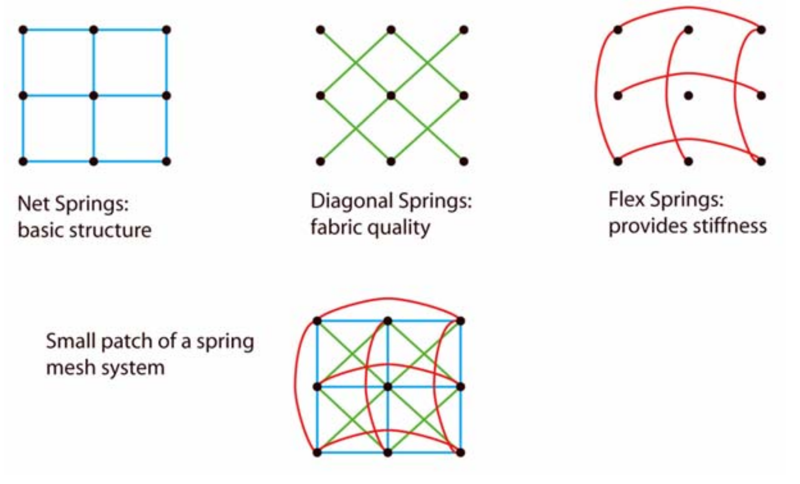
A spring *s0* has:

p0, p1	*two particles it connects*	
L	*current length*	*scalar*
L(0)	*reference length*	*scalar*
K	*spring constant*	*constant*

The Processing algorithm determines the position of p1, p2 at time t(n)+1 = t(n)+ h and so forth, using a Runge-Kutta solver. A regular grid of mass particles includes 3 types of springs and configurations, modeling 3 internal stresses (stretch, shear and bending) with their respective coefficients (*kstrech, kshear, kbend*).

Let's now move on from curves to surfaces and the general theory behind springs and particles: For any given two dimensional surfaces, let us assign the familiar geometrical parameters: U, V and T (U and V describe our coordinates in "paper space" and T stands for Time). If we hold U and V constant, the resultant function describes the journey of points in time; if we hold T constant – the resultant function describes the journey of points in space. Thus the first constants (U,V) allows us to trace the *velocity* of any given point P, and the other constant (T) allows us to describe the *stretching* of the surface at any given point P. We are now beginning to tie time and space as independent parameters in our design space.

Simulating the stretch fabric surface (rubber latex membrane):

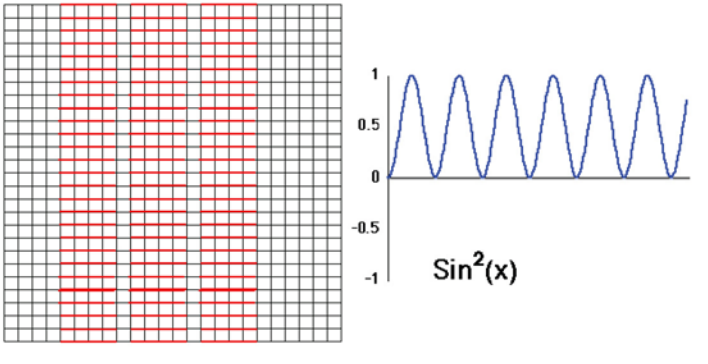


► Figure 5: Composite image illustrating the stretch fabric simulation logic. Three underlying computational structures were modeled as the initial mesh: (a) net springs provide for the basic structure (stretch); (b) diagonal springs mimic fabric behavior (shear); (c) flex springs provide for additional stiffness (bend).

In terms of the computational system itself - three major components exist which allow for the digital emulation of physical material behavior (Figure 5).

Deforming the mesh: extrinsic vs. intrinsic deformation:

Mesh deformation is carried out either by introducing extrinsic deformation (adding accumulative amounts of springs to the mesh locally) or by introducing intrinsic deformation (modifying the characteristics of the springs themselves across the surface). As the membrane strongly resists stretching motions while being comparatively permissive in allowing bending or shearing [8], the simulation had to account for the intrinsic forces as much as the extrinsic ones.



► Figure 6: Increasing the length of red regions driven by a Sin^2(x) curve introduces intrinsic deformation to the mesh.

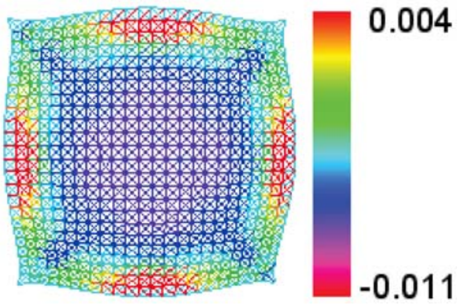
Figure 6 illustrates a sin function which informs the way in which the mesh deforms. The degree of change may potentially be modified by applying varied mathematical formulas.

Controlling the degree of mesh deformation by modifying the Gaussian curvature of the mesh:

The Gaussian curvature of a surface at a point is the product of the principal curvatures at that point. The tangent plane of any point with positive Gaussian curvature touches the surface at a single point, whereas the tangent plane of any point with negative Gaussian curvature cuts the surface. Any point with zero mean curvature has negative or zero Gaussian curvature. The Gaussian curvature of a polygon around any vertex is the angle defect of that vertex:

$$(6)\ AD = 2\pi - \xi\varphi(i)$$

Where $\varphi(i)$ are the angles of the triangles that contain the current vertex.



◀ Figure 7: Gaussian curvature representation indicating areas of positive and negative curvature in the mesh.

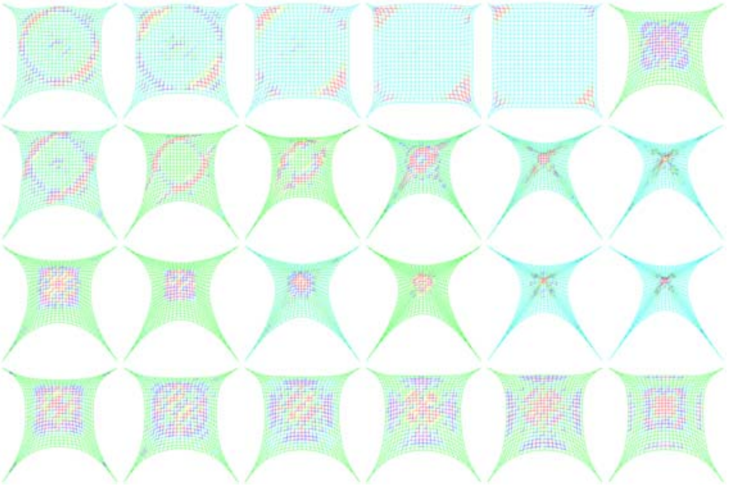
Figure 7 illustrates the color-coded Gaussian curvature in a generic mesh model: Any points on the surface with curvature values between the values specified by the user will be displayed using the corresponding color. For example, points with a curvature value half way between the specified values will be green. Points on the surface that have curvature values beyond the red end of the range will be red and points with curvature values beyond the blue end of the range will be blue. A positive Gaussian curvature value means the surface is bowl-like (synclastic or positive curvature). A negative value means the surface is saddle-like (anticlastic or negative curvature).

5. Tool demonstration

Experiment 1: Basic Simulation

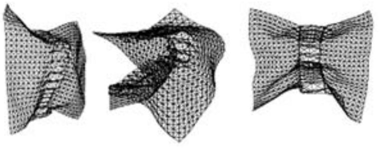
The mesh variables include the Spring Constant (K) indicating the strength of the fabric, the gravity force (G) within the modeling environment, the time step (T) for each increment, and the damping factor (D). Figure 8 illustrates consecutive instances of mesh behavior in stretch. The images below are color-coded according to stress conditions: red indicating tension; blue indicating compression.

► Figure 8: Composite image showing 24 consecutive instances of one mesh model simulating the stretch-fabric behavior.



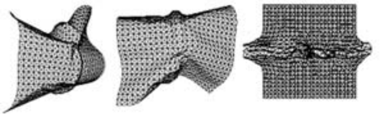
Experiment 2: Extrinsic stretch emulating resin impregnation ***perpendicular*** to stretch

► Figure 9: Digital simulation of resin impregnation which is applied perpendicular to stretch.



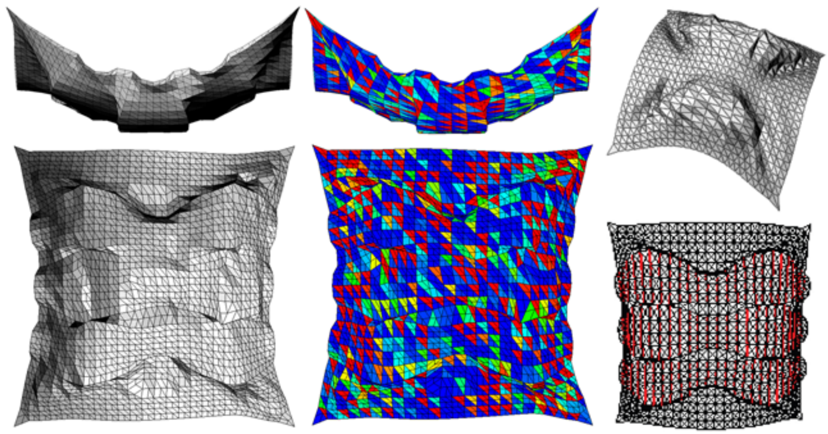
Experiment 3: Extrinsic stretch emulating resin impregnation ***parallel*** to stretch

► Figure 10: Digital simulation of resin impregnation which is applied parallel to stretch.



The above two experiments (experiments 2 and 3) demonstrate the digital simulation of the initial physical experiments as shown in Figure 2. The top images illustrate the digital output of a stretch simulation which is applied to a resin-impregnated mesh where the resin is applied **perpendicular** to the stretch direction and results in local folds along impregnation after release. Bottom: impregnation is applied **parallel** to stretch direction and results in global curvature of the fabric after release. Compare with Figure 2 illustrating the physical equivalent.

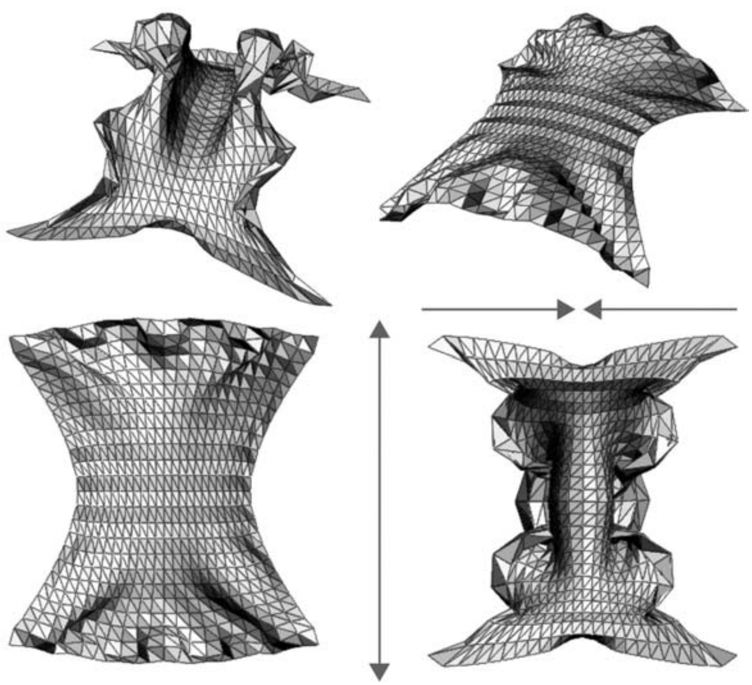
Experiment 4: Intrinsic Mesh Properties Sin^2(x)



◀ Figure 11: Digital Simulation of Intrinsic Forces in Mesh.

This experiment illustrates the behavior of the mesh when constrained by intrinsic forces. Negative curvature is introduced, the result of which is an anticlastic global surface curvature. The bottom right image (Figure 11) illustrates (in red) the springs which are added to counteract the stretch.

Experiment 5: Shrink Fabric: Mapping Gaussian Curvature as Negative Curvature

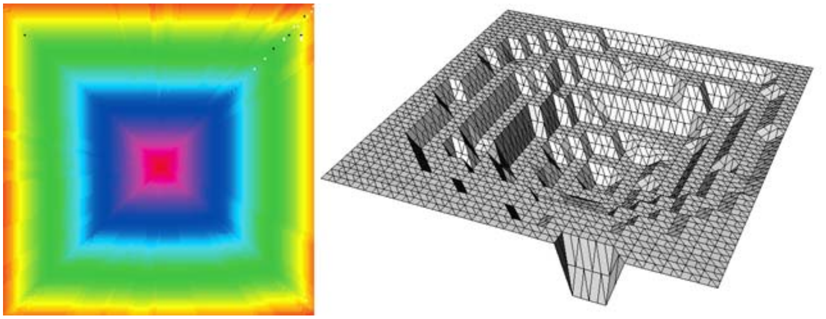


◀ Figure 12: Digital Simulation of Applying Intrinsic Forces in mesh to Induce Negative Curvature.

Anticlastic surface curvature is produced when constraining (Figure 12) the central part of the mesh by maintaining spring length. The springs in the

periphery elongate locally to cater for the tension introduced in the fabric, promoting the global negative curvature and the anticlastic form.
Experiment 6: Generating 3-D structure based on a 2-D curvature representation

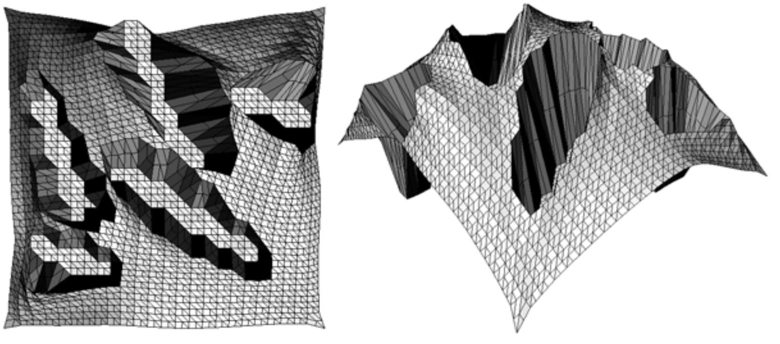
► Figure 13: Reverse-engineering method to produce 3-D structures based on Gaussian curvature map.



Based on material attributes which have been processed into the computational protocols, the tool is now able to generate in addition to its ability to simulate: given a specific *two-dimensional representation* of a Gaussian map, the tool will generate the 3-D membrane and expose it to any given "environmental" conditions (gravity, damping etc). The predictive value of the model is yet to be refined. The final aim would be to predict the resin pattern and distribution needed in order to achieve any given surface geometry.

Experiment 7: Prediction of material behavior under the introduction of topological changes within mesh

► Figure 14: Topological changes introduced within the material system while material behavior is being predicted by computational tool.



The introduction of a cut or a hole (size and distribution are controlled by the tool user) illustrates change in kind, rather than change in degree within the material system at hand. The tool is designed to predict stress distribution around the area of the cut as a gradient of stresses across the springs comprising the mesh.

6. Summary and conclusions

◀ Figure 15: Three representations of the two physical experiments (far left images) and their digital simulations as illustrated by the tool

The tool presented here is one of many potential applications in design computation which promotes the integration of material properties and behavior alongside associative modeling environments which allow for structural optimization. In doing so, it seeks to define a synergetic approach to generative design processes by linking the material world to that of geometry and promoting design explorations of an iterative nature bridging physical and digital modeling. Beyond its value as an operative tool for design, this exploration raises some interesting realizations regarding the computation of material qualities.

6.1. Finding-form by controlling anisotropic material organization

◀ Figure 16: Physical model of resin-impregnated rubber-latex membrane

Isotropic materials are materials which exhibit properties (such as velocity of light transmission) with the same values when measured along axes in all directions. Anisotropic materials are those which exhibit properties with different values when measured in different directions. The level or degree of "property differentiation" may be determined across different scales of investigation. When examined both in the local (molecular material composition) scale and the global scale (defined herein as building scale), the rubber-latex membranes exhibit homogeneous properties of behavior in all directions. The act of applying resin strips over the surface, given a specific pattern arrangement, creates a material composite (resin impregnated rubber-latex) which acts in a heterogeneous manner when stretched and released. In other words, the resin patterns inform stretch and strain paths across the surface which transform its internal organization and contribute to its anisotropic behavior. The resin basically introduces external forces on the material causing it to deform. This process informs the way in which the computational model works.

6.2. Predicting 3-D material behavior based on 2-D impregnation patterns

The experiments illustrated above demonstrate how through the modulation of 2-D impregnation patterns it is possible to induce curvature within rubber-latex pre-stretched membranes. Experiments in other disciplines have been carried out which support the relation between 2-D mappings and 3-D manifestations of curvature [15]. Such, for instance, is an experiment which proves that the elliptical and hyperbolic patterns are characteristic of periclinal (natural) folds and can be used to classify structures according to different curvature attributes: Elliptical patterns indicate structures with synclastic curvature whereas hyperbolic patterns are indicative of anticlastic curvature [15]. These findings support conditions in which 3-D elastic form may be generated by 2-D impregnation patterns which induce specific local and global curvatures within the stretchable surface. This implies perhaps that certain rapid-fabrication techniques, such as CNC resin-injection, may support automated fabrication of membrane structures based on 2-D digital "impregnation patterns".

6.3. The *homogenization process*: integrating physical data in digital protocols

The process of translating physical material behavior into digital particle-system-based entities requires matching mathematical and geometrical equivalents to physical phenomena such as stress, strain, gravity etc. This is where a structural, or rather organizational, hierarchy is needed for the simulation (see figure 7 illustrating simulation logic). The notion of "homogenizing" physical parameters which are not of the same order of magnitude and/or do not register, or measure, with the exact same unit conventions, promotes an approach for "calibrating" these orders such that they are comparable to one another. For example, strain – generally conceived of as a physical material property, may also be referred to as the geometrical expression of deformation caused by the action of stress on a physical body. Strain therefore may express itself as a change in size and/or shape. The deformation of the mesh object is defined by a tensor field, i.e., this strain tensor is defined for every point of the object. This field is linked to the field of the stress tensor by the generalized Hook's law. Given that strain results in the deformation of a body, it can be measured by calculating the change in length of a line or by the change in angle between two lines (where these lines are theoretical constructs within the deformed body). The effect of gravity upon the stretchable mesh is also measured in length units. So in order to generalize the problem of "physical computing" it is therefore crucial to ask and inquire what are the physical parameters which are later translated into geometrical entities measured by a systematic method for quantification.

6.4. Computational strategies for emulating physical behavior of globally-modulated systems

The experiment illustrated in this paper demonstrate a system of materials (resin-impregnated rubber-latex composite) which is globally modulated across its entire surface area (by pre-stretch, impregnation and release) to promote both global and local curvature. Two strategies in general were instrumentalized to simulate the material behavior across both local and global scales. The first strategy was to differentiate the forces of stretch within the mesh based on a given mathematical formula which was integrated into the script; the second strategy was to locally add springs as needed to simulate different degrees of stretch once force is applied in a non-uniform manner. These two strategies promote two very different approaches to the computation of material properties. The first strategy appropriates the mesh in its entirety and seeks to treat it as a *continuous* geometrical entity modulated by a global formula; the second promotes *discreteness* of material elements, or components, which do not necessarily go hand in hand with the way in which the material itself is described physically. It is probably worthwhile emphasizing that the closer the computational strategy is to the material logic, the more inherent – and thus better – it becomes. Membrane structures seek continuous solutions whereas component-based systems would promote a more discrete strategy with regards to computation of material properties.

6.5. *Reverse-engineering*: predicting physical behavior in digital form

In order for this computational tool to become a generative medium for the designer it must allow for an iterative, open-ended process of simulation, evaluation and generation. Some of the experiments shown here (specifically, experiment 6) have demonstrated such capacity in that the designer is able to draw out his own Gaussian-curvature map (or any other map which could potentially be linked to physical performance) and the material-system, at hand, would automatically be generated to suit these criteria. This feature has great significance for the future development of the specific tool illustrated here and/or any other tool developed in the framework of the larger theory and practice of material computation.

Acknowledgements

This work has been carried out by the authors' collaboration in the framework of an MIT course directed by Dr. Dennis Shelden, Chief Technology Officer of Gehry Technologies. The authors wish to thank Dennis for his informative and inspiring seminar. The authors also wish to thank Yanni Loukissas, teaching assistant, and Dr. Axel Kilian for their invaluable input.

The Java programming for the tool developed here were executed in Processing, a 3D programming environment developed for the design community by Benjamin Fry and Casey Reas at the MIT Media Lab, Cambridge, USA.

References

1. McCarter, R.; *Louis I. Kahn,* Phaidon Press, New York, 2005.
2. Otto, Frei, *Finding Form,* Edition Axel Menges, Berlin, 1995.
3. Shea, K., Aish, R. and Gourtovaia, M., *Towards Integrated Performance-Based Generative Design Tools,* Digital Design 21th eCAADe Conference Proceedings, 2003, pp. 553-560.
4. Hesselgren, L., and Oxman, N.; *Folded Plate Roof Research Project* in Achim Menges, "Instrumental Geometry", Hensel, M., Menges, A and Weinstock, M., Techniques and Technologies in Morphogenetic Design, Architectural Design, March/April, 2006, pp. 52-53.
5. Burry, J; Felicetti P.; Tang, J.; Burry, M, and Xie, M.; *Dynamical structural modeling: a collaborative design exploration* International Journal of Architectural Computing 3(1), 2005, pp. 27-42.
6. Oxman, N.; *"Performative Morphologies: The Vertical Helix"*, in Hensel, M. and Menges, A., Differentiation and Performance: Multi-Performance Architectures and Modulated Environments, Hensel, M., Menges, A and Weinstock, M., Techniques and Technologies in Morphogenetic Design, Architectural Design, March/April, 2006, pp. 66-67.
7. Kilian, A.; *Linking Digital Hanging Chain Models to Fabrication* ACADIA Fabrication: Examining the Digital Practice of Architecture, Proceedings of the 23rd Annual Conference of the Association for Computer Aided Design in Architecture and the 2004 Conference of the AIA Technology in Architectural Practice Knowledge Community, 2004, pp. 110-125.
8. Barraf, D., and Witkin, A.; *Large Steps in Cloth Simulation,* In SIGGRAPH Computer graphics Proceedings, Annual Conference Series, pp. 43-54.
9. Wang, Wang and Yuen, *Fast energy-based surface wrinkle modeling* in Computers & Graphics, issue 1, 2006, pp 111-125.
10. Peña and Catalán, *A Computer Program for the Simulation of Folds of Different Sizes under the Influence of Gravity,* in Computers & Geosciences, issue 1, 2004, pp 33-43,
11. Malinov and Sha, *Software products for modelling and simulation in materials science,* in Computational Materials Science, issue 2, 2003, pp 179-198.

12. Romero et al, *Parallelization issues of a code for physically-based simulation of fabrics* in Computer Physics Communications, issue 3, 2004, pp 188-202.
13. Gibson, L. J.; Ashby, M., F.; Karam, G., N.; West, U., and Shercliff, H., R.; *the Mechanical Properties of Natural Materials. II. Microstructures for Mechanical Efficiency,* Royal Society Publishing, London, 1995, pp. 123-140; 141-162.
14. Hübsch, P. F.; Middletona, A. J.; Knoxb, J; *Computation of the elastic properties of a resin impregnated wire mesh by the means of a homogenisation procedure* Medical Engineering & Physics, Elsevier Science Ltd 22(7), 2000, pp. 521-524.
15. Lisle, R., J, *Dupin's indicatrix: a tool for quantifying periclinal folds on maps,* School of Earth, Ocean and Planetary Sciences, Cardiff University, Cardiff CF10 3YE, UK. Geological Magazine, Cambridge University Press (CUP) 140(6), 2003, pp. 721-726.

Neri Oxman, PhD Candidate and Presidential Fellow
Computation Group, Department of Architecture, **MIT**
and
Jesse Louis Rosenberg
Department of Mathematics, School of Science, **MIT**
Address and E-mail:
Apt. 815, 70 Pacific St., Cambridge, MA 02139, USA

neri@mit.edu, nerioxman@gmail.com