SENSPECTRA
An Elastic, Strain-Aware Physical Modeling Interface

by

Vincent Leclerc

Dipl. Cinematographic Studies
Université de Montréal (2000)
B.Sc. Computer Science with Specialization in Computational Arts
Concordia University (2003)

Submitted to the Program of Media Arts and Sciences,
School of Architecture and Planning,
in partial fulfillment of the requirements for the degree of

Master of Science in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2006

Author ____
Program in Media Arts and Sciences
May 12, 2006

Certified by ____
Hiroshi Ishii
Professor Media Arts and Sciences
Program in Media Arts and Sciences
Thesis Supervisor

Accepted by ____
Andrew B. Lippman
Chair, Departmental Committee on Graduate Students
Program in Media Arts and Sciences

SENSPECTRA

An Elastic, Strain-Aware Physical Modeling Interface

by

Vincent Leclerc

Submitted to the Program of Media Arts and Sciences,
School of Architecture and Planning,
on May 12, 2006 in partial fulfillment of the
requirements for the degree of
Master of Science in Media Arts and Sciences

Abstract

Senspectra is a computationally augmented physical modeling toolkit designed for sensing and visualization of structural strain. The system functions as a distributed sensor network consisting of nodes, embedded with computational capabilities and a full spectrum LED, which communicate to neighbor nodes to determine a network topology through a system of flexible joints. Each joint, while serving as a data and power bus between nodes, also integrates an omnidirectional bend sensing mechanism, which uses a simple optical occlusion technique to sense and communicate mechanical strain between neighboring nodes. Using Senspectra, a user incrementally assembles and refines a physical 3D model of discrete elements with a real-time visualization of structural strain.

While the Senspectra infrastructure provides a flexible modular sensor network platform, its primary application derives from the need to couple physical modeling techniques utilized in the architecture and industrial design disciplines with systems for structural engineering analysis, offering an intuitive approach for physical real-time finite element analysis. Utilizing direct manipulation augmented with visual feedback, the system gives users valuable insights on the global behavior of a constructed system defined as a network of discrete elements.

Thesis Supervisor: Hiroshi Ishii
Title: Professor of Media Arts and Sciences

SENSPECTRA
An Elastic, Strain-Aware Physical Modeling Interface

by

Vincent Leclerc

Thesis Committee:

Advisor

Dr. Hiroshi Ishii
Professor in Media Arts and Science
MIT Media Laboratory

Thesis Reader

Dr. John Maeda
Professor in Media Arts and Sciences
MIT Media Laboratory

Thesis Reader

Dr. Daniel L. Schodek
Kumagai Professor of Architectural Technology
Harvard Graduate School of design

Acknowledgements

I would like to thank my readers, John Maeda and Daniel Schodek, for their inspiration and constructive criticism. I also want to my advisor, Hiroshi Ishii for introducing me to his radically original vision of human-computer interaction, for providing me with a framework to be self-critical about ideas and technology, and for giving me the opportunity to learn and share ideas among a community of brilliant thinkers, hackers and makers.

Thanks to the members of the Tangible Media Group who helped me polish half-baked ideas and solve numerous technical issues. I especially want to thank Amanda Parkes who I had the opportunity to collaborate with for the creation of Glume and Senspectra.

Thanks to all the Media lab colleagues who made it all worth it by surprising me everyday, challenging my ideas, debugging technical problems, and by being available day and night. Thanks to the Media Lab sponsors for supporting our projects and without who the Media Lab would not exist.

Special thanks Joey Berzowska for introducing me to this wonderful community, for her moral support and for bombarding me with a million quirky ideas. I appreciated every moment spent with Joey and her team.

Merci à ma famille pour m'avoir encouragé à poursuivre mes études. Merci d'avoir respecté mes choix de vie tout en me conseillant avec votre regard critique et votre experience personelle.

Merci à Corinne pour partager ma vie. Merci de me rappeler quotidiennement les valeurs qui nous ont unies il y plus de 6 ans. Merci pour ton support moral, pour ta stimulation intellectuelle et pour ta joie de vivre rayonnante.

Table of Contents

Chapter 1: Introduction

Model-making using both physical materials as well as computational analysis and simulation provides a methodology to understand and represent the world. Data visualization provides a powerful educational technique as well as an important activity for professionals, particularly in scientific and creative fields such as mathematics, architecture, design, and engineering. In the past, scientists have used clever mechanical instrumentation to study invisible phenomena but the increasing complexity of the studied data has led scientists to mostly rely on mathematical models. Combined with computational power, mathematical models offer potential for extensive simulations to be done extremely quickly and be directly visualized on screen.

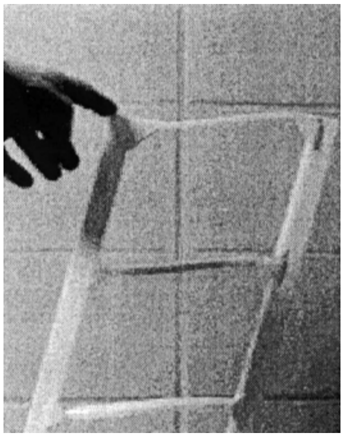

Figure 1: Visualizing lateral force on a physical model made of fixed joints, stiff columns and flexible beams [25].

However, in many of these fields, physical 3D model-making is still preferred over, or applied in conjunction with, on-screen GUI modeling tools such as CAD, as visual *and* tactile aids in the creative and learning process. Physical models allow rapid experimentation with a system to understand its structure and limitations and are much less prohibitive to non-scientists for experimenting with complex data sets.

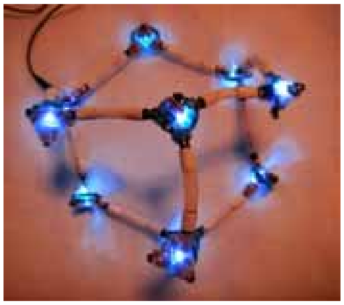

Figure 2: Simple model made of Senspectra primitives.

Computers have proven good at generating rich simulation on physical models but I would argue that their traditional input and output (I/O) modalities are not appropriate to work on such digital models. I will try to show that the semantic gap between these I/O devices and the data they control hinders users’ ability to gain a complete understanding of the systems they work on, and will attempt to provide a better alternative.

Thesis Overview

This thesis describes the design process that led to what became Senspectra, the new interaction modalities it offers and its future.

Chapter 1 introduces the project, describes its general context and sets the thesis framework.

Chapter 2 presents my inspirations behind Senspectra. It gives an overview of my interest in learning with tangible interfaces and my motivation for exploring the materiality that makes these interfaces.

Chapter 3 situates Senspectra in context with previous projects on a technical and conceptual level.

Chapter 4 describes Glume, a 3D physical modeling interface that led to the creation of Senspectra.

Chapter 5 is dedicated to Senspectra, its design, implementation, usability implications and limitations.

Chapter 6 concludes with potential additions to make Senspectra more interesting and application domains that could benefit from a user interface such as Senspectra.

Chapter 2: Motivation

The work I have had the opportunity to pursue with my colleagues in the Tangible Media Group has evolved from a range of interests that revolve around interaction design, cognitive psychology, and fine arts. The following sections put my research into perspective in relation to those fields.

Tangible Media

The major motivation behind my work at the Media Laboratory is Professor Ishii's *tangible bits* vision [24]. Instead of focusing on graphical user interfaces (GUI) to interact with digital systems, Ishii advocates the seamless mapping of digital information to physical objects in an attempt to "embrace the richness of human senses and skills people have developed through a lifetime of interaction with the physical world" [24]. Tangible user interfaces clearly offer new opportunities for novel interactions with computer systems. They take advantage of the physical persistency of atoms to offer hybrid interfaces that live in both the digital and the physical world.

Figure 3: Abacus [23] – Beads in an abacus represent abstract, intangible numbers through physical means but can also be used as controllers to perform arithmetic operations.

One important area of interest for me was the concept of I/O coincidence, where the input and the output of an interface are collocated. A radical example of I/O coincidence is inTouch [08] where users experience collocated mechanical input and output. In this case, users share mechanical motion over distance by moving their hands on the rollers of their respective inTouch device. Whatever motion is input on one device is reflected on another and vice-versa. InTouch removes every boundary between input and output to create a completely seamless interface.

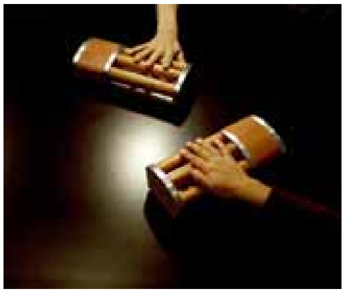

Figure 4: Two users communicate force and motion through their respective inTouch device [08].

Learning with Digital Manipulatives

The eminent 19th century educator Pestalozzi was among the first to recognize the potential of manipulatives as tools to make abstract concepts more accessible to children in the classroom. He developed his doctrine of *Anschauung* (direct, concrete observation) whereby the subject of study must be observed and explored with the senses before using language to synthesize it.

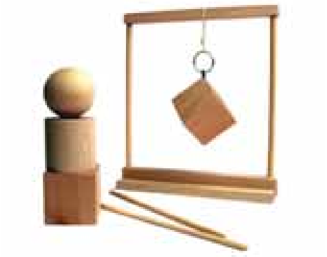

Figure 5: Froebel's "Gift #2" - Sphere, Cylinder & Cube [16]

Building on Pestalozzi's theories, Frederich Froebel, the inventor of the kindergarten, developed a sequential series of *gifts* that consisted in physical objects for children to play with. Froebel specifically designed these objects to help children discover and experiment with common natural phenomena [07]. More recently, informed by Froebel's ideas, Maria Montessori developed teaching tools where physical materials play a crucial role in the education of children. Her manipulatives carefully make use of children's senses to help them learn by exploring and discovering relationships between objects [32].

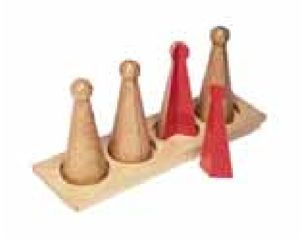

Figure 6: Set of four large wooden skittles representing divisors of 1 whole, 1/2, 1/3 and 1/4 [33]

Mitchel Resnick and his team at MIT Media Lab Life-Long Kindergarten coined the term *digital manipulatives* [40], physical manipulatives augmented with computing capabilities that provide new means for children to explore and experiment with the physical world.

Figure 7: Two elementary-school students experiment with a creature they created with a Programmable Brick [40]

These endeavors resonate well with the recent theories of constructivism, or more specifically *constructionism*, where iterative physical modeling and analysis is at the core of the learning process [36]. Seymour Papert has proposed numerous concrete approaches to constructionism that diffuse computational construction kits into the lives of children specifically to change the context in which they learn mathematical concepts.

Inspired by their work I wanted to contribute to the field by exploring the potential for digital manipulatives to create interfaces for an older audience, namely college students or even professionals. More specifically, I wanted to explore how networks of digital manipulatives can be utilized for simulating invisible behaviors of various natural systems by engaging the users' tactile and visual senses.

John Frazer popularized the idea of using networks of physical building blocks to create digital models more than twenty years ago when he proposed the Three-Dimensional Intelligent Modeling System, a reconfigurable network of cubic building blocks [15]. Many others followed with similar systems that take advantage of the physical immediacy of a digitally augmented construction toolkit. Most of them, however, rely on the physical interface as a mere source of input data, separating the output and shifting it to a conventional display screen. [04, 21, 28, 47]. Furthermore, most toolkits have a very limited set of legal configurations, usually orthogonal, which make it difficult to appropriately model natural phenomena.

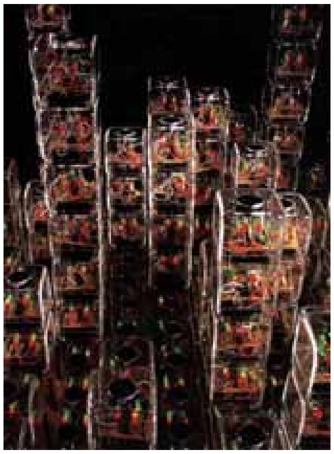

Figure 8: Frazer's Three-Dimensional Intelligent Modeling System [15]

The overarching goal of my thesis is to create a constructive assembly [46] that attempts to eliminate the I/O dichotomy that faces current computationally enhanced physical modeling toolkits. To accomplish this, Senspectra provides a set of control-feedback loops through embedded sensors and LEDs that effectively collocate the physical model and the computer-generated visualizations. The motivation behind this is to help users understand complex physical phenomena quicker and to develop stronger intuitions on the behaviors of the systems they study.

Taking Advantage of Physical Qualities in Materials

Senspectra arose from an interest in exploring the physical properties of the material substance that forms constructive assemblies. The usual conception of digitally augmented constructive assemblies is a set of control mechanisms materialized in their physical *shape*. Not surprisingly, in nature, shape is fundamentally related to function. On a molecular level, it has been shown that the canonical shape of a molecule is closely related to its chemical behavior and reactivity in regards to other molecules [06].

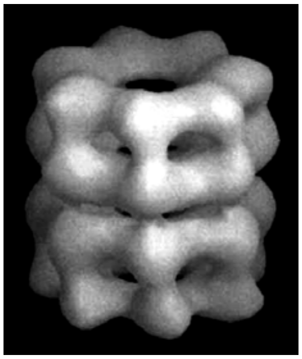

Figure 9: GroEL-nucleotide complex [41]. Proteins interacting with this barrel-shaped complex are reshaped as they pass through its rings.

On a larger scale, it is also clear that the shape of objects informs their users about their function. Psychologist James Gibson introduced the concept of *affordance* in 1966. Gibson's affordances refer to the set of 'action possibilities' for an object and the user's ability to recognize these possibilities [19].

More concerned about product design, Donald Norman refined the term to *perceived affordance* [34] when the user of an object perceives the function of that object by looking at its morphology. For example, a pull plate placed on a door would afford to be pulled and a push plate on the same door would afford to be pushed. Norman says "when affordances are taken advantage of, the user knows what to do just by looking: no picture, label, or instruction needed" [35]. This *visual* affordance plays a crucial role in users' understanding of the behaviors of an object at the very first steps of the interaction process. However, I was curious to see what happens after the visual exploratory phase, when users start engaging with the object and actually manipulate it. Could *tactile* or *haptic* affordances be helpful in guiding individuals in using an object or an interface appropriately?

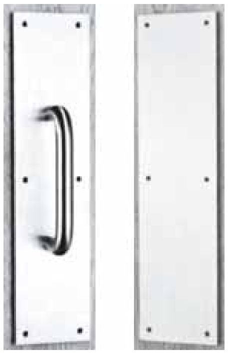

Figure 10: Pull plate and push plate.

“We can often feel what can be done with something – that it is hot enough to fry an egg, sharp enough to slice a tomato” says William Gaver [17]. However, most tangible interfaces and constructive assemblies remain interested in creating toolkits where *visual form informs function*. In Senspectra and Glume’s case, I attempted to broaden the scope towards *materiality informs function*. Specifically, how texture, plasticity and elasticity inform the user about the function of physical interfaces.

Figure 11: The strawberry-cactus.

In nature, beyond their color and morphology, the tactile material qualities of objects and living creatures also inform other organism about their function. The tenderness of berries and their thin, fragile skin invites animals to grab them gently. On the other hand, cacti’s thorns while being visually inviting, clearly do not afford to be picked from a tactile perspective.

Figure 12: Liquid drops or data droplets?

In interface design, materiality connotes a variety of qualities that can be the source of rich sensory experiences and occasion for numerous action modalities. From a tactile perspective, the static quality of rigid objects affords unary or binary controls. Hard objects can be simply touched or pressed in a binary fashion, but they do not afford to be pressed at various pressure levels. Malleable objects however, have a compliant material quality that invites users to a tactile exploration. The physical act of *deforming* a malleable tangible interface can take several meanings. One can, for example, perceive the affordance of squeezing matter or data out of a flexible pipette or dropper when applying pressure to the suction bulb. Similarly, if a user sees a pull plate on a door, engages with it, but the door does not open, the haptic affordance of the handle will suggest that the door is most likely locked.

Summary: Computational Materiality

This thesis centers around what designers call interfacial materials [09], which is defined as materials that interface digital systems to the physical world. Interfacial materials, however deal with the notion of creating skins that bridge the physical world and the digital world. I believe that the concept could be pushed a little further with the creation of materials which are inherently embedded with computational power and not merely be an interface to a dislocated computing system. I decided to treat the concept of tangible media literally and explore how bits can materialize to become atoms in order to create a new physical medium augmented with computational qualities.

As computational systems become more and more integrated in our everyday physical environment and more and more tied to our tangible materiality, I believe it is increasingly critical to develop new physical & digital systems that afford the forms and structures of the natural world and incorporate their materiality into the control-feedback loop.

In consequence, my work in the Tangible Media Group has tried to explore and answer the following questions:

- How can the material properties of a tangible interface inform its control structure?
- How can malleable materials be used to create tangible interfaces?
- How can a set of distributed digital manipulatives create interfaces with new interactive and material properties?

Chapter 3: Background and Related Work

Constructive Assemblies

The concept of constructive assembly refers to a set of building primitives that can be used to model a more complex system. Over the past 30 years, there have been many projects that explored the idea of physically modeling systems with computationally augmented building blocks.

Physical Modeling

Aish's Building Blocks [03] was among the first computational construction kits that allowed a designer to create a 3D CAD model by constructing its physical counterpart using uniform building blocks. By connecting cubes to one another, users could create a physical structure along with its physical counterpart on a computer screen.

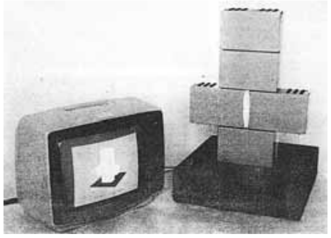

Figure 13: Building Blocks [01]

Determination of a three dimensional form by communication through a set of connected shapes, has been demonstrated by numerous systems which act as physical networks of a digital information topography. John Frazer's 3D Intelligent Modeling System from 1980 [15] set an early precedent as a set of stacking building blocks which send messages to adjacent blocks to determine geometric configuration. This work was followed by projects such as Digital Clay [51] developed at Xerox PARC, a system of rhombic dodecahedrons whose modules have the capacity to sense their own orientation in space with respect to other modules.

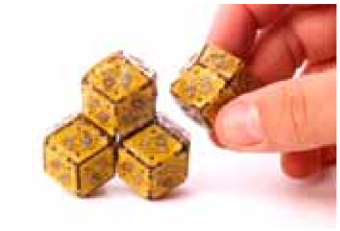

Figure 14: Four Digital Clay primitives [51]

Similarly, the MERL Blocks are a set of building blocks that can describe the morphology of the structure into which they are connected [04]. It uses a distributed networking architecture to relay geometrical information from block to block until it reaches the host computer that renders the structure on a GUI. Users can build 3D models with the bricks and then modify them with GUI-based 3D modeling software.

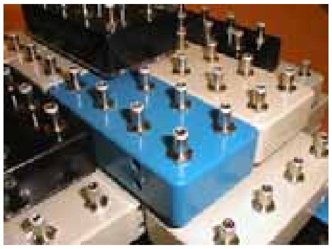

Figure 15: MERL Self-Describing Building Blocks [04].

More recently, MIT's Tangible Media Group gave birth to Triangles [21], a physical/digital construction kit, which allows users to physically manipulate connections and relationships between data elements. By physically connecting triangles together, users can associate digital files and processes to one another. The Triangles system is less concerned with modeling the 3D world, and more interested in seeing how digital processes can be interfaced to with a 3D constructive assembly.

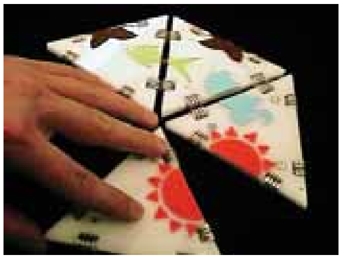

Figure 16: Four Triangles [21].

Coincidental I/O

Topobo [39] is a constructive assembly with the ability to record and playback motion. It is made of interlocking passive primitives and active motors controlled by an embedded microcomputer. With Topobo, users can create creatures and teach them how to move by example, by physically deforming their creation and record the deformations. Topobo shares a lot of commonalities with Senspectra and was a great source of inspiration for the design of the toolkit. Topobo was created to teach children about notions related to mechanical motion in a constructive assembly setting. Its record and playback mechanism introduces the idea of collocated I/O in constructive assemblies much like inTouch [08] did for haptic interfaces.

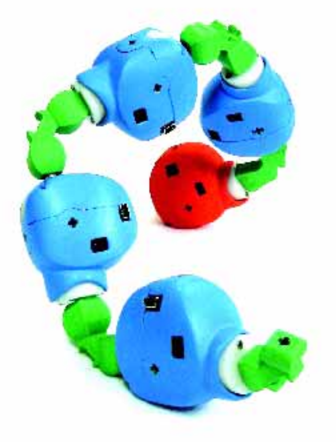

Figure 17: A helix made of five Topobo Actives [39].

Structural Analysis

Analog Tools

Conceptually, Senspectra builds on photoelastic analysis, an optical modeling technique used to give experimental stress analysis based on observation in polarized light of transparent models [30]. Photoelastic analysis gives a view of the overall structural behavior of a clear cast epoxy model of an object or a structure. It is particularly useful when mathematical models of an object are too complex to compute.

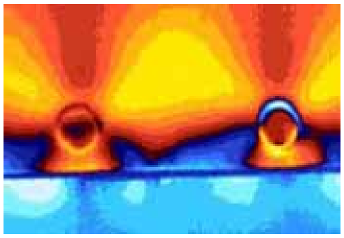

Figure 18: Photoelastic analysis of the stress concentration around rivets [11].

The photoelastic analysis process was developed in 1930 by Coker and Filon [12]. It is based on the property of birefringence exhibited by certain transparent materials. Light passing through a birefringent material experiences two refractive indices. Photoelastic materials exhibit the property of birefringence only under mechanical stress and the magnitude of birefringence at a specific point is directly proportional to the amount of mechanical stress at that point [50]. When looking through a polarized film, one can see the fringe pattern created by a given mechanical stress on a given model.

Tensegritoy [13] is another interesting approach to structural analysis using analog tools. It is different from other physical modeling tools because its edges are not static lines simply defining space, but representations of attractive and repulsive forces in a state of equilibrium. It consists in a set of wooden sticks, plastic caps and elastics to attach the sticks together. Tensegritoy introduces to its users the concept of structures whose morphology is maintained by a continuous tensional network [13]. With this toolkit, users can experience the forces and attraction and repulsion by feeling the tension on the elastic cords and looking at the compression on the wooden struts.

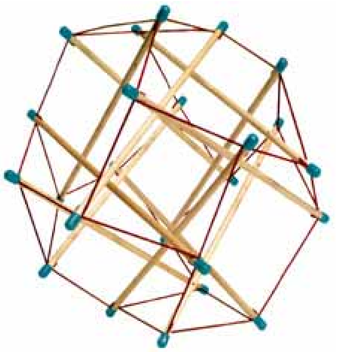

Figure 19: A sphere made with the Tensegritoy.

Digital Tools

Senspectra seeks to couple the physicality of photoelastic analysis with the digital simulation techniques of finite-element analysis (FEA) [27] and on-screen visualizations of structural strain, to provide an interface with direct material manipulation and real-time visual feedback of strain. The finite-element method models surfaces as a mesh of discrete points connected to one another to form finite elements [42]. This method facilitates the use of structural models to analyze the effect of loads on surfaces. FEA models only give an approximation of the behavior of a structure under stress and are very dependant on the meshing algorithms used in the analysis. It is often useful to perform FEA with different mesh models of the same structure to truly reveal its behavior under stress or strain.

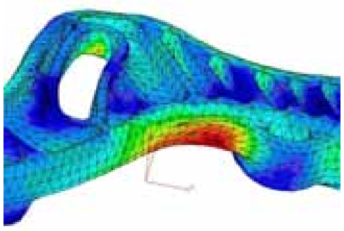

Figure 20: FEA on a pair of pliers using the COSMOS software [38].

Another significant contribution to the field of structural analysis is the digital polariscope. Digital polariscopes have been introduced to structural engineers to dynamically perform photoelastic analysis. They allow the recording of sequential snapshots of the photoelastic analysis of a given model in order to allow its users to review the effect of force over time in the form of a graphical representation of the model. Digital polariscopes introduce the concept of dynamic photoelastic analysis that was previously difficult to obtain with analog tools. Inspired by this technique, the ability to dynamically record mechanical stress over short periods of time soon became a crucial design requirement for Senspectra.

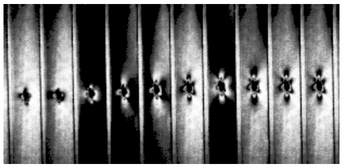

Figure 21: Snapshots of the changes in the stress patterns recorded with a digital polariscope [05].

Distributed Physical Computing

Amorphous Computing

Recently engineers have started exploring the potential of systems with very large amounts of distributed computers. They have been interested in coherent behavior emerging from such systems. One recent example of these endeavors is the work of Bill Butera on what he calls *paintable computing* [10], agglomerates of numerous miniature computing units that can cooperate as autonomous entities part of a larger process.

Figure 22: Gradient propagation on computer-simulated distributed nodes [10].

The discipline is best described by its instigators from the Amorphous Computing effort at MIT CSAIL.

> "The objective of this research is to create the system-architectural, algorithmic, and technological foundations for exploiting programmable materials. These are materials that incorporate vast numbers of programmable elements that react to each other and to their environment. Such materials can be fabricated economically, provided that the computing elements are amassed in bulk without arranging for precision interconnect and testing. In order to exploit programmable materials we must identify engineering principles for organizing and instructing myriad programmable entities to cooperate to achieve pre-established goals, even though the individual entities are unreliable and interconnected in unknown, irregular, and time-varying ways." [48]

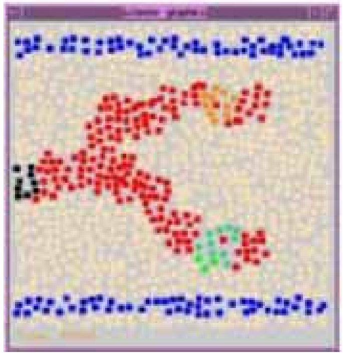

Figure 23: Pattern formation in an amorphous computing simulated visualization [01].

The Amorphous Computing group works on building software infrastructures for distributed environments where clusters of processes can communicate, coordinate and self-organize to solve problems that could not be solved with a single node.

Sensor Networks

The MIT Media Lab Responsive Environments group has done a fair amount of work in distributed physical computing. One of their projects, Tribble [37], shares a lot of commonality with Senspectra. It consists of an array of sensors embedded on a networked surface. Tribble can generate sounds, colors and vibrations depending on how it is manipulated. It functions as a completely decentralized system where there is no central controller. Similarly to Senspectra, Tribble’s behavior emerges from the way its sensory data is digitally coupled to its output modalities. Senspectra however offers a more relaxed infrastructure that is more appropriate for 3D modeling applications.

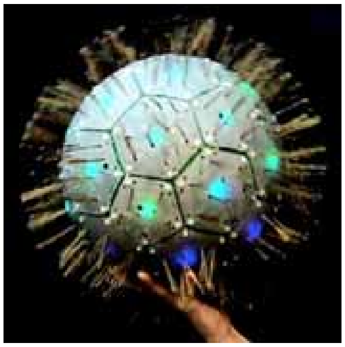

Figure 24: Tribble [37].

Lifton’s Pushpin computer [37] is another project that came out of the Responsive Environments group. Pushpin nodes draw power from the board they are pushed into and communicate to their neighbors through infrared in a decentralized fashion. Each node has a multitude of sensors attached to its surface and can change the color of a full-spectrum LED according to its algorithms. In addition to their peer-to-peer communication scheme, the Pushpins can be controlled by a central process through an IR beacon that globally communicates to all nodes placed on the board. Pushpins can do self-localization by measuring the delays between light flashes and ultrasonic ‘clicks’ in order to get a crude approximation of the current morphology of the system. The Pushpin computer project is generally focused in developing mechanisms for processing various external stimuli in a distributed architecture.

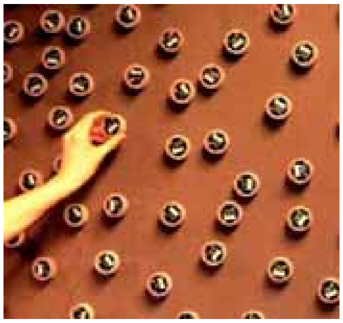

Figure 25: Pushpin computer [37].

Chapter 4: Glume, the Amorphous Constructive Assembly

Glume was my first foray into the realm of interfacial materials. It is a modular and scalable building system with the physical immediacy of a soft and malleable material. Glume explores a unique area of augmented building materials by combining a discrete internal structure with a soft and organic material quality to relax the rigidity of structure and form in previous tangible building block approaches.

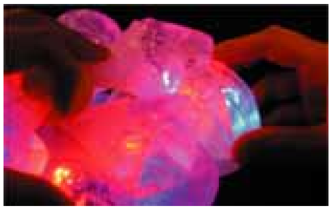

Figure 26: A user manipulating Glume primitives.

Interaction Context

I originally envisioned Glume as a tool for constructing and manipulating models; visualizations and simulations of organic three-dimensional data sets. I wanted to provide an interface with which users could construct 3D compositions with radically amorphous primitives. I strongly believed that such a tool would better represent the plasticity of the things that surround us and would be more appropriate for modeling natural phenomena.

To construct a model, users would combine Glume modules, interlocking and shaping the nodes into place. As the user would build, the system would determine the model's morphology, first defining an origin point at the base of the model and recursively looking at the neighbors of the 'base' node to define a crude morphology. During the construction process, the system would assist the user by providing colored cues within the model by, for example, displaying a color gradient changing from the centre of the structure outwards or a color-coded elevation map of the model. Once the model would be constructed, users would associate it with a predefined semantic model of a specific volumetric map related to the construction.

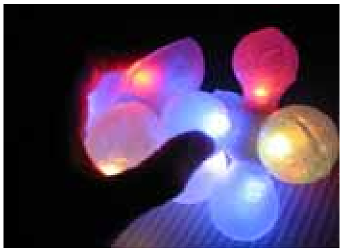

Figure 27: A user grasping two Glume modules.

Hydrology & Flow Propagation

As the first implementation of Glume, we decided to build a prototype set of nodes that would illustrate its interaction capabilities using a hydrology modeling scenario, chosen because it makes use of 3D data sets in irregular structures (such as soil) and requires direct manipulation of spatiotemporal parameters. We also explored other possible applications such as architecture, geodynamics, medicine and archeology.

In a hydrologic assessment situation, users would construct the 3D geologic map of a terrain while taking into account important physical parameters for the simulation such as soil characteristics and land surface slopes. To manipulate a model, users would modify the parameters of a node or a group of nodes by introducing an *object modifier* into the system. Object modifiers could affect a single node directly during the building process, and its color would be adjusted to reflect the change, mapping different properties (such as types or density of particulates) to different colors. If object modifiers were added to an existing model, the system would regenerate its semantic model to reflect the new parameters. In a hydrology model, users could simulate the propagation of a pollutant plume and visualize its effects on the geological map, by placing several object modifiers representing pollutant sites on the surface and simulate the pollutants propagation over time.

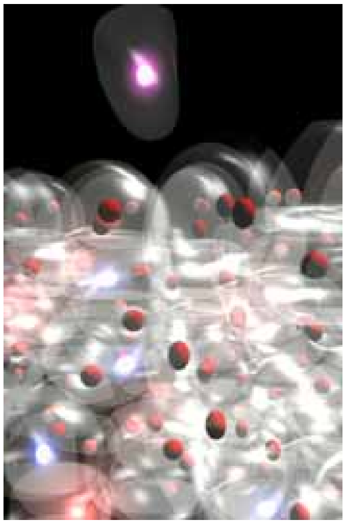

Figure 28: 3D rendition of an object modifier being introduced in a Glume model.

Glume would also enable users to get an extended set of properties associated with a specific node in the physical model using *probes,* which would automatically provide the parameters associated with each node when touched. Finally, users could also *touch* nodes to highlight isovolumes as part of a simulation, for example, touching a 'polluted' node to highlight the propagation of that specific pollutant.

System Design

The Glume system consists of soft and translucent computationally augmented modules which communicate capacitively to their neighbors to determine a network topology and are responsive to human touch.

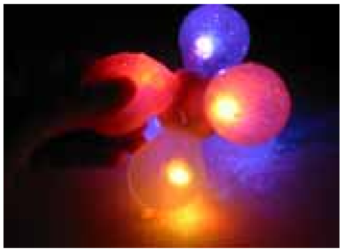

Figure 29: Glume module.

Design Principles

The Glume infrastructure was designed to retain the tactile experience of a soft modeling material, while creating a new identity and extended functionality for the material. To achieve this goal we established guidelines for the physical and digital design of the system:

- Retain a flexible malleable form while incorporating a regular recognizable stacking geometry
- Induce a tactile sensation similar to sculpting with a soft moist material
- Provide translucency to see inside a model
- Allow for distributed or centralized functionality
- Incorporate a touch response as feedback to the user

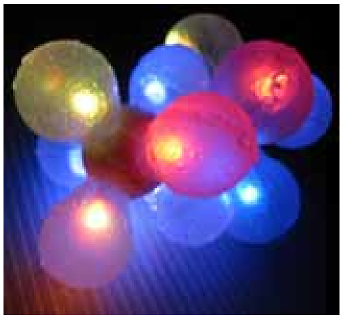

Figure 30: Two interlocking Glume modules.

Current System

An individual Glume module consists of six silicone *bulbs* connected to a central *nucleus*. The bulbs form the bulk of the module and are distributed evenly around the nucleus. I designed the initial prototype as one nucleus per bulb much like the structure of an egg or a cell. Inspired by multi-cellular organisms, I had the intuition that a building block of that form factor would be appropriate for modeling biological systems, but would also provide the organic qualities of artistic mediums such as clay. The original design had the advantage of being amorphous and very sculptable, but for technical reasons such as resolution and power we soon switched to the current system.

Figure 31: R. Sylvatica Eggs [49].

Engineering

Nucleus

The nucleus is the computational core of the Glume infrastructure. It holds the electronics and the lithium-polymer battery. We 3D printed the nucleus shells with a starch printer and reinforced them structurally with cyanoacrylate glue. The circuit board securely attaches to the shell and the battery occupies the remaining space.

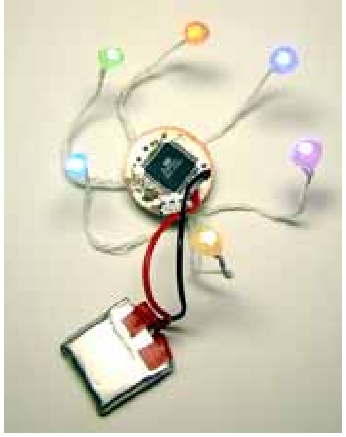

Figure 32: Six full-spectrum LEDs attached to a Glume nucleus.

The circuit board holds all electronic components and connectors for the LEDs and capacitive electrodes. I devised the system architecture in three distinct modules: a pulse-width modulation (PWM) display adapter and six red-green-blue (RGB) LEDs; a frequency-shift keying (FSK) modulator-demodulator and six gel electrodes; and a micro-controlling unit.

The Glume system is driven by an ATMEGA32L AVR 8-bit RISC microcontroller running at 8MHz. It controls the FSK networking peripheral, determines the color transitions, and performs the capacitive sensing. The power source is a 3.7V lithium polymer (Li-Poly) battery that is down regulated to 3.3V. Under normal operating conditions, the battery lasts for more that one hour.

The display adapter consists of two PWM drivers controlled serially by the AVR. Together, they independently drive six full spectrum RGB LEDs embedded in the bulbs of a Glume unit. The MAX6966 PWM drivers are controlled by the AVR through SPI. While the drivers control the color of the bulbs, the AVR can be used to perform more important operations such as capacitive sensing or networking.

The FSK module uses the PWM unit provided by the AVR to modulate two different frequencies into a multiplexer that redirects the signal to one of the six electrodes. The same multiplexer also redirects the incoming frequencies into the input capture unit of the AVR. The transmitter shifts from 31.25KHz (1) to 3.91KHz (0) at a baud rate of 200 bps. Transmission is done by generating a square wave with one general purpose I/O pin of the ATMEGA32L. The signal is distributed in one electrode at a time through an analog multiplexer. The FSK module also acts as a capacitive sensor to determine when a user is manipulating a Glume unit.

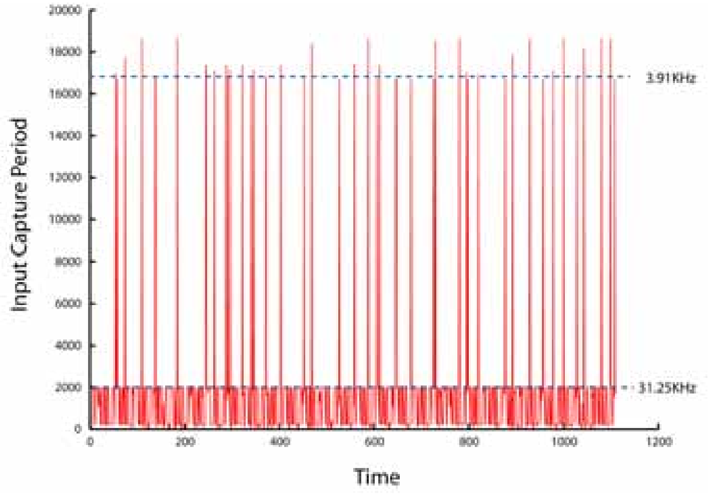

Figure 33: FSK bits decoded by the interrupt-driven timer on the AVR.

Organic Shells

The silicone skin of each bulb has been cast in Smooth-On Sorta-Clear 40, a translucent silicone rubber. It is a platinum curative silicone that cures at room temperature with no shrinkage. It has a shore hardness of 40A and tensile strength of 800psi. The hollow castings were made from molds modeled in Autocad and then printed using a 3D starch printer.

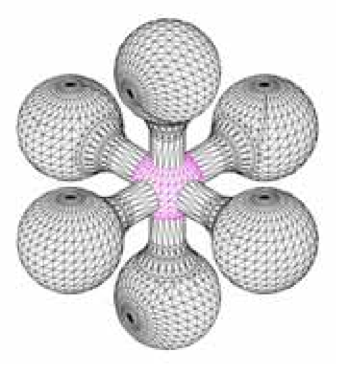

Figure 34: 3D CAD model of a Glume module used for designing the casting molds.

The bulbs are embedded with Softee Protein Styling Hairgel® chosen for its optical clarity and conductive characteristics. The combination of the thin silicone shell and the embedded gel provides the tactile effect that each bulb will retain the shape as sculpted in place by the user.

Figure 35: Hair gel embedded in Glume's bulbs.

Networking

Technically, Glume's networking system builds on developments in distributed and decentralized hardware and software architectures. Zimmerman's Personal Area Networks: Near-field intra-body communication [52] demonstrated how capacitive coupling of charges through the body could allow electronic devices on the body to exchange digital information.

In Glume's case the gel contained in the bulbs serves as the medium for the capacitive coupling electrodes. The gel primarily consists of a solution of Sodium Salt of Carbomer as emulsifier and source of ions for electrical conductivity and Propylene Glycol that provides viscosity and structure.

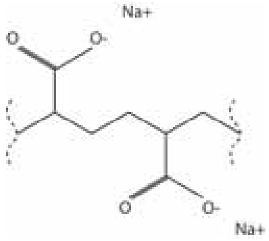

Figure 36: Mobile ions in the Sodium Salt of Carbomer solution.

The gel has a resistivity (ρ) of 7.07 X 10^2 Ωm and each electrode has a capacitance averaging around 30pF. We can calculate the energy consumption as follows:

$$
\begin{aligned}
E &= \tfrac{1}{2}CV^2 \\
&= \tfrac{1}{2} * 30e10^{-12} * (3.7)^2 \\
&= 205 \text{ picoJoules per bit communicated}
\end{aligned}
$$

Capacitive coupling in the Glume system offers significant advantages over traditional wireless short range communication schemes like IR. For constant transmission at 30KHz, the power consumption is about 6.15µWatts. At similar baud rates and distances, IR transmission generally consumes power in the milliWatts range. The bulk of the energy consumption for reception is drawn by the opamp and at roughly 1mWatts. It is thus clear that capacitive communication is ideal for this battery-powered system.

Another compelling feature of capacitive coupling is the short range of the transmission channel. The distance between the transmitter and the receiver must be less than 10mm for the capacitive coupling module to work reliably. The signal strength between two nodes decreases very rapidly and tends to zero when two nodes are separated by ~5cm. Effectively, only immediate neighbors can exchange information. This necessary proximity is crucial for Glume's system architecture where only neighboring nodes can communicate with one another in order to easily determining the topology of the Glume network.

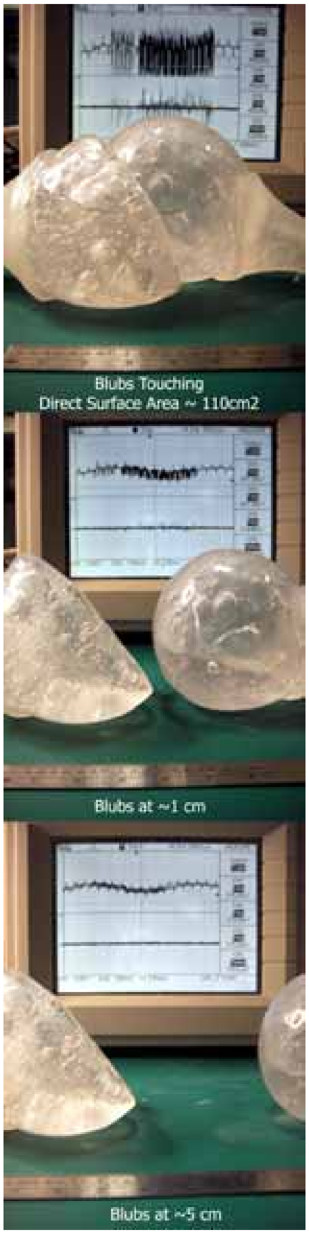

Figure 37: The amplitude of the capacitive coupling signal decreases rapidly as the bulbs are separated.

I had the initial intuition to use frequency-shift keying (FSK) and use two band-pass filters followed by pulse stretchers to filter and amplify the incoming signals. After trying several designs, it prove impossible to create high-order band-pass filters with high gain while keeping the number of components on my board to a minimum. I also looked at synchronous demodulation, but the hardware necessary to achieve proper demodulation demanded too many components and was much too pricy to be used in such a system.

I finally designed a custom communication protocol that could be described as pulse-period modulation. The transmitter basically sends pulses and changes the period between pulses to send a 1 or a 0. It's conceptually the same thing as FSK where the switching between frequencies is done after each single period, but the filtering is done in the firmware.

The incoming signal goes through an analog multiplexer into an inverting high-pass filter. The positive input of the opamp is biased at 1.85V to facilitate the recovery of both phases of the transmission signal. The high-pass filter is a simple first-order active filter with a -3 dB rolloff at 1kHz. It accurately filters out any noise below that frequency. The 20MΩ feedback resistor provides a gain of -1250 effectively saturating the opamp to make the signal long enough to be captured by the ADC.

The ADC measure pulse periods and determines if a packet is being sent or not. There are 10 framing bits and a silence at the beginning of each packet so the receiver knows when to start parsing the packet. The firmware looks at delays between pulses to determine the nature of the data being sent. The algorithm rejects any incomplete packet. A 16 bit cyclic redundancy check (CRC) signature is sent along with each packet to detect any transmission error.

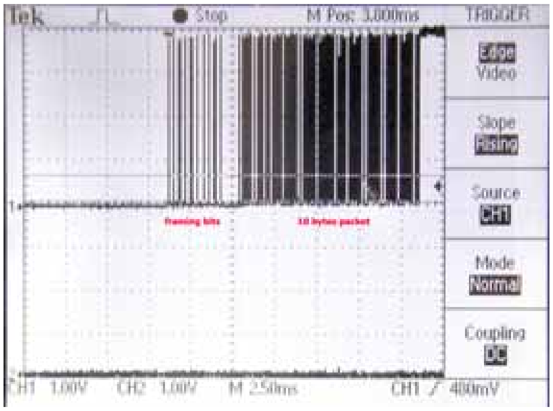

Figure 38: Frequency signature of a packet and its framing bits.

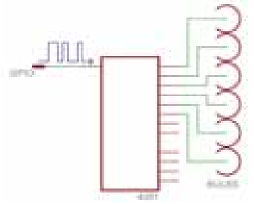

Figure 39: Schematic of the FSK signal multiplexed in the conductive gel inside the six Glume bulbs.

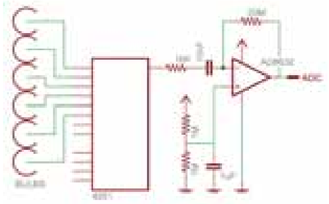

Figure 40: Schematic of the circuitry for receiving, demultiplexing, filtering and amplifying the FSK signals.

Limitations of the Glume Design

The original intent for Glume was to create a system that allowed people to organically design 3D models. We had put a lot of emphasis in creating a very amorphous primitive that would permit organic modeling. During the design process and after experimenting with the current system, several issues became apparent.

Resolution and Depth

The major limitation of the Glume infrastructure is its low granularity. Each bulb measures approximately 6cm in diameter effectively creating pixels bigger than ping-pong balls. This is a serious usability limitation and makes it hard to compete with GUI-based systems for any serious scientific application.

One way to overcome this issue is to fully take advantage of the physicality of the interface, by not only using it as a modeling tool, but also use the model as a controller to more complex parameters in the system after the model is designed. This aspect is hard to introduce in the current iteration of Glume because the only input mechanism available is touch (through the capacitive sensor). The act of touching could be used to select a Glume module to reveal more information about its state or to reveal other modules with a similar state in the model. The major inconvenient of this input modality in the Glume interface is that users can only touch/select modules at the surface of the model. Pixels have a clear advantage over atoms in that situation. One possible solution to the problem would be to introduce remote control interaction techniques to enable users to control physically inaccessible modules.

Regularity

The amorphous physical nature of a Glume module is radically different from its rigid computational structure. I have yet to design a simple data structure that would accurately represent the dynamic morphology of the modules. Currently, the changing morphology of the Glume bulbs is completely invisible to the nucleus. For that reason, the global digital model is inconsistent and incoherent most of the time. I did not succeed in finding a geometric model that would reliably match the network topology of connected Glume nodes. This was in part due to my inexperience in wireless networking, but could be mostly attributed to the fact that the current Glume system has no mechanism to consistently determine the current shape of an individual module.

Gravity

In a perfect world, Glume modules would stick to one another without sticking on human skin much like play dough. This would allow users to compose 3D shapes that are more or less independent of gravity. When we started designing Glume, we quickly switched to an interlocking primitive to overcome the gravity problem. However, even the current modules don’t stack well enough to solve that issue. One possible solution would be to introduce rigid modules that could act as a skeleton in a 3D composition.

Transparency

The silicone that forms the skin of Glume bulbs is not totally transparent. For that reason only the 3 top layers of modules are visible to the user. Future iterations should take that into account by either switching materials or introducing air gaps between the nodes.

Chapter 5: Senspectra

Glume helped me refine the design principles that should define a malleable distributed digital manipulative. When it became clear that Glume had too many technical and usability limitations it was easy to go back to the drawing board to design the new iteration. From my experience with Glume I designed Senspectra, an elastic digital manipulative that retains some of the material qualities of Glume in a more regular primitive.

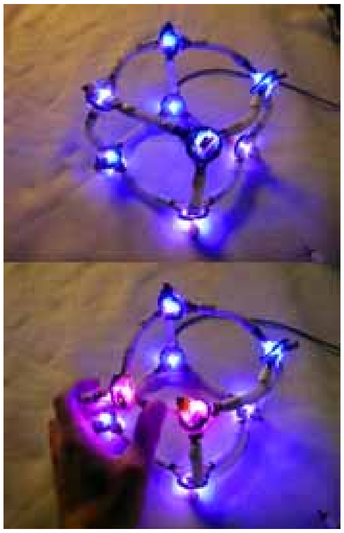

Figure 41: Senspectra structure reacting to the stress induced by a user’s hand.

Senspectra is a malleable interface with which users can define soft geometries and reconfigure them at will. When changing the physical connections in the structure, the system reshapes its digital model to reflect the physical structure and feeds it back into the structure so that each node has a general understanding of the overall state of the system. The nodes can then change their color to reflect some otherwise invisible aspects of their current nature. The direct visual feedback gives users instantaneous cues on invisible parameters in the model. With these new insights, users can refine and analyze their models through direct manipulation of the physical structure.

Interaction Context

While we have built Senspectra as a flexible sensor network platform for visualizing mechanical strain, the original inspiration for the systems derives specifically from frustrations and limitations of linking physical models used in architecture and industrial design to systems of structural engineering analysis, and in educational contexts, to gain an intuition on form and strain. Conversations with professionals and students in architecture and building technology, informed the material and computational qualities of Senspectra, attempting to provide an appropriate toolkit for the design and assessment of structures organic in form.

Structural Analysis

Senspectra's flexible connectors provide a unique material modeling quality, which when combined with the capability of locally measuring and displaying a color based on natural and induced mechanical strain, helps designers and architects to compose structures of non-regular geometries and perform *distributed finite-element analysis* (DFEA) in real-time as they construct their models. Instead of globally analyzing the stresses in a model as traditional FEA tools do, Senspectra's DFEA computes the stresses locally with each individual node integrating the surrounding stresses to obtain a unique local stress vector.

Figure 42: Structure made out of a chemistry modeling kit.

Current FEA tools are effective at showing potential failures in indeterminate structure in the form of digital models. However, there are currently no means for users to seamlessly modify a structure (physical or virtual) and visualize the effect of those modifications in real-time. Senspectra not only provides this real-time FEA functionality, but also collocates the output of the visualization directly on the physical model.

System Design

Design Principles

Senspectra retains most of the design criteria we had established for Glume. The original motivation was to create a modeling interface that is better at representing organic phenomena. It failed at doing so because we put too much emphasis on creating something amorphous without taking into account the low granularity of the interface. Senspectra addresses this issue by offering a more regular constructive assembly while affording organic constructs and control mechanisms. The regularity helps the system understand the morphology of the constructed models and provides users with a limited set of legal configurations that simplifies approximation in the design process.

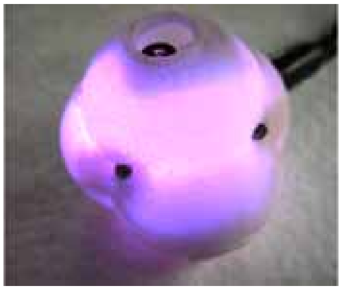

Figure 43: Senspectra tetrahedral light-emissive node.

I decided to offer the user a set of primitives that, at first glance, only allows the creation of regular structures. However, by making the primitives flexible and elastic, the morphology of the regular structure could be altered. The elasticity of the primitives was a crucial element of the system design for two reasons. From my experience with Glume, I was still convinced that malleability of was a critical quality for a constructive assembly toolkit. But more importantly, from a usability perspective, embedded continuous controllers were necessary in this new interface. One of the limitations Glume had was that users could only touch the modules in a discrete fashion, therefore reducing the set of possible interactions considerably. By introducing elastic controllers as part of the Senspectra primitives set, users would be able to change parameters in a continuous fashion.

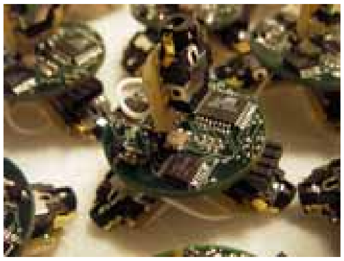

Figure 44: Senspectra Nodes.

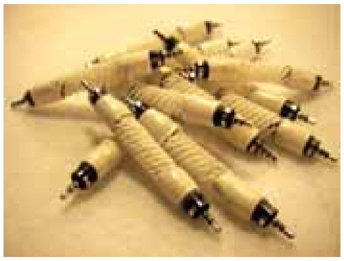

Figure 45: Senspectra Joints.

Another important criterion for Senspectra's design was ease of assembly. I wanted to make the assembly process as transparent as possible so the user could focus on construction rather than assembly. The solution to that was to keep the set of primitives to a strict minimum and design a universal connector that serves as a mechanical attachment as well as an electrical and networking link to other nodes.

One lesson I learned from Glume and various 3D displays is that it is difficult to see through and explore a dense set of volumetric pixels. When the pixels are tangible, things get even harder because there is no non-destructive way to physically access pixels that are not on the surface of the model. Senspectra takes that limitation into account and introduces empty interstices between nodes to facilitate the access and manipulation of 'buried' nodes.

The motivation behind making Glume as a set of amorphous primitives was to represent the natural world more appropriately. This is still an important criterion in Senspectra. However I decided to approach the problem on a different angle and look at natural structures at different scales and base the design of Senspectra on these observations.

Designing an interface based on these principles led to a set of 2 primitives that form the current version of Senspectra. Together, these primitives abide by the criteria I had established and resolve most of the limitations I had encountered with the Glume system. Senspectra *nodes* are made of a hard translucent shell and are embedded with full spectrum LEDs controlled by a microcontroller and several other electronic components. They have 4 isoangular connections in a tetrahedral configuration in which a *joint* can be introduced to mechanically connect nodes to one another. The joints are made of a hollow

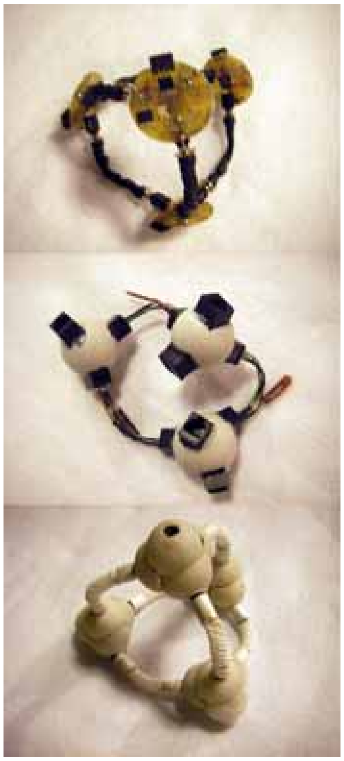

Figure 46: Evolution of the Senspectra toolkit. Top: First system with no strain sensors; Middle: Second system with resistive bend sensors on the joints; Bottom: Current system.

silicone tube and can sense the angle at which they are flexed. They communicate that information directly to the nodes they are connected to as well as doing power distribution and networking. Their radial connector freely rotates on the nodes they are attached to. Users can connect nodes with joints to create structures that retain plastic properties that natural and artificial structures would exhibit. The following section explains the design decisions that were taken for making these 2 primitives.

Light-Emissive Nodes

Senspectra nodes are shaped in the form of a tetrahedron. I explored many candidates for the morphology of the nodes and tetrahedrons seem to be the most appropriate for several reasons. Tetrahedral and triangular structures are present in nature at many instances. Evolution seems to show that this type of configuration makes the most efficient use of material for cellular structure such as insect wings, leaves, muscle filaments, the protein shell of some viruses, etc. At a larger scale, insects also use triangulated patterns in construction. The honeycomb walls of a bee hive are a classic example. At a smaller scale many crystals are arranged in a tetrahedral sturcture. Atoms of carbon form bonds to one another in this fashion to make diamond. The same process happens with silicon atoms in semi-conductors.

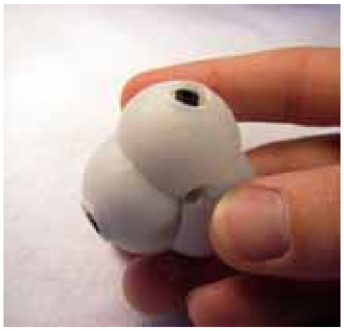

Figure 47: Senspectra node.

Another compelling reason for using tetrahedrons in Senspectra was that a tetrahedral bond is the simplest geometrical unit to form 3D shapes. For all these reasons, tetrahedrons seemed to be the perfect for a primitive in a semi-*regular* constructive assembly made of fixed and rigid nodal points.

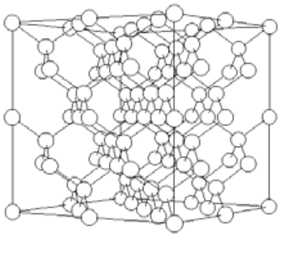

Figure 48: Diamond crystal lattice inscribed in a cube [44].

Flexible Joints

An important consideration I had when designing Senspectra was to keep some of the malleability that we had managed to obtain with Glume. The main criterion was to create an interface that has a regular internal structure, but that is malleable and elastic. From the beginning, it seemed more appropriate to keep rigid nodes, but introduce elasticity in the joints. I explored different materials, looking for tubing that would have a high elasticity and a high threshold before experiencing permanent plastic deformations. Silicone tubing not only has this property, but its high elasticity gives it the desirable quality of not kinking when bent - even at 180 degrees.

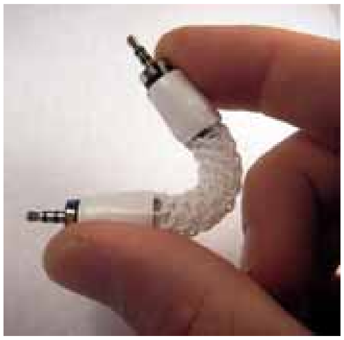

Figure 49: Senspectra joint.

The sensing mechanism I used for the bend sensor inside the silicone tubing forced me to place the wires around the tube. I decided to braid the wires used for networking and power distribution in a pattern that can be stretched and keep a consistent mechanical response to stress.

Another important usability concern was to find a simple way for users to connect nodes and joints together without having to focus on the assembly process. The connector used in Senspectra joints is fully radial freeing users from having to align or orient the joint connector with the node connector. The radial connector also allows nodes to freely rotate around the joints they are connected to without inducing extra mechanical strain on the structure.

Graphical User Interface

While we have observed the physical immediacy and direct manipulation characteristics of Senspectra to be engaging and intuitive as an interface, one prevalent limitation of the current Senspectra system is resolution and scale. In order to address this issue, we have developed an initial system to interface Senspectra with an on-screen representation, allowing for interpolation to provide a greater resolution of a model. The system is connected to a PC and once the network topology is determined, every nodal connection is fed into a simulated annealing model [18] that generates a virtual 3D representation of the physical structure on-screen.

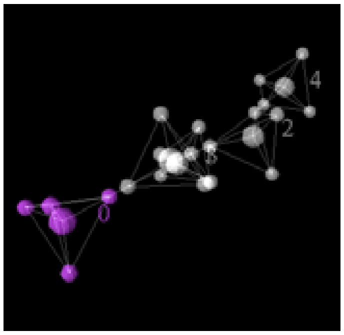

Figure 50: Screenshot of a 3D model generated by the Senspectra GUI.

The GUI is currently stripped down to a minimum. It shows the physical geometry as a structure of graphical nodes connected to one another and floating in empty space. Users can click on a node to reveal its ID and see what neighbors are connected to it. Users can also reset the model to regenerate the geometrical model of the physical structure. In the rare case where it does not yield to the appropriate model, users can introduce vibrations in the 3D model to force it to realign into a more stable geometry.

We have also provided means for users to navigate the 3D space around the model using a keyboard.

Engineering

Joints

The joints are used to mechanically connect nodes. They are made of silicone tubing and eight conductors wrapped around the tube in a diamond braid. This braiding technique is commonly used for making ropes where flexibility and elasticity is of importance. I decided to apply it to the Senspectra joints to maintain a consistent flexibility and elasticity throughout the designed structures. The braided wires serve for power distribution, peer-to-peer networking, and the sensing of the bending angle of the joint.

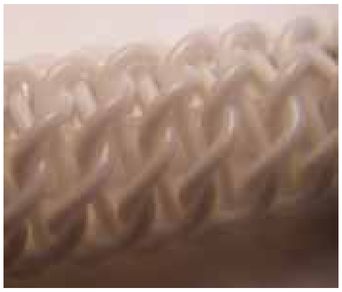

Figure 51: Braided wires around Senspectra's joints.

The tips of the joints are made of two radial connectors similar to headphone connectors. They allow free rotation of the connected nodes freeing the users from having to twist the joints to connect two nodes together. The free rotation also reduces the undesired mechanical stress in the designed structures that is induced by the physical limitations of the toolkit itself.

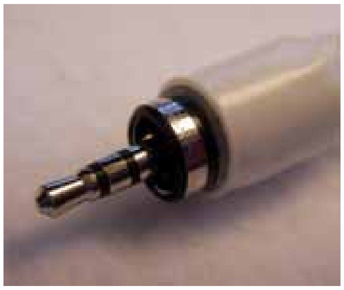

Figure 52: Radial connector.

The silicone tubing within the joints serves as Senspectra's omnidirectional bend sensing mechanism. It uses a simple and inexpensive optical occlusion technique measuring the intensity of infrared light coming from an LED on one end of the joint with a matching phototransistor placed at the other end. When the joint is straight, the intensity of the infrared light is at its maximum. As the joint bends, the tubing occludes the light to a point where the phototransistor cannot detect any infrared light emitted from the LED. The advantage of using this method over traditional resistive bend sensing is that the joints can bend in any direction and give consistent readings.

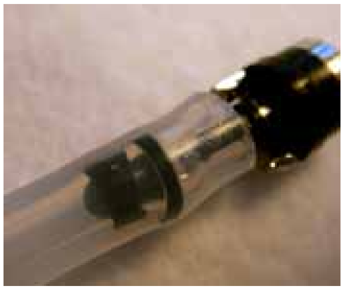

Figure 53: Phototransistor embedded in the silicone tubing that forms Senspectra's joints.

Nodes

The nodes are made of a 3D printed ABS plastic shell embedded with a surface mount PCB. The shell is rigid and translucent to protect the electronics and act as a light diffuser for the embedded full-spectrum LEDs. The shell has four holes for connecting joints to the node. The connectors are respectively separated by 109° angles to form a perfect tetrahedron.

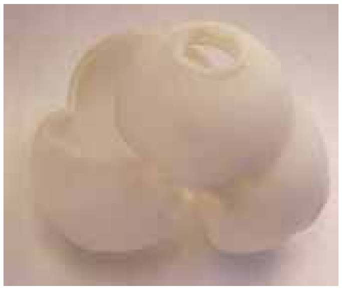

Figure 54: Senspectra Node translucent ABS plastic shell.

The heart of the circuitry is an 8-bit RISC AVR ATMEGA8 microcontroller running at 8MHz. It controls the color of the 2 RGB LEDs through a custom made timer-driven pulse-width modulation algorithm. I decided to eliminate the PWM drivers that we used in Glume to reduce the price of the nodes and to reduce their circuit board area to a minimum. The algorithm I wrote uses an 8-bit hardware timer on the AVR and can control the brightness of up to 16 LEDs independently.

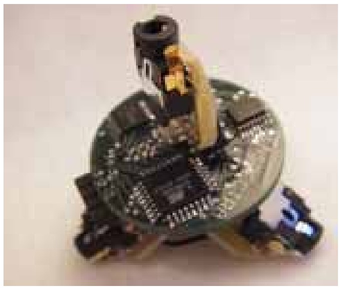

Figure 55: Senspectra node assembled circuit board.

The AVR also calibrates the signal coming from the bend sensors through a simple voltage follower. I had to pull-up the signal coming from the connectors to prevent false readings when no joints are connected. The pulled-up signal then goes through an operational amplifier and is read by an analog-to-digital converter on the AVR.

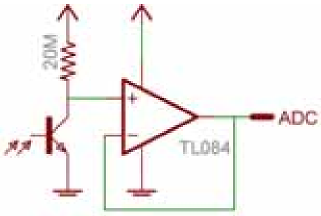

Figure 56: Schematic of the sensing circuitry for the omnidirectional bend sensor.

The microcontroller communicates with neighboring nodes through a UART that is multiplexed on four channels. The multiplexer is protected by 4.7KΩ inline resistors to prevent failure of the analog-to-digital converter when connecting and disconnecting joints. The UART runs at 500kBps.

Sensing

The stress analysis is done locally by every individual node. On the image on the right, if you look at node B for example, every few milliseconds it computes its local mechanical strain by adding the force vectors S_{AB}, S_{BC} and S_{BD}. The force vectors come from the optical-occlusion bend sensors embedded in the joints attached to the node. The computed value is then normalized and is mapped to a color value that gets read by the LED controller. When someone applies pressure on the structure, the structurally weakest nodes turn red informing the user where the weak points are in the structure.

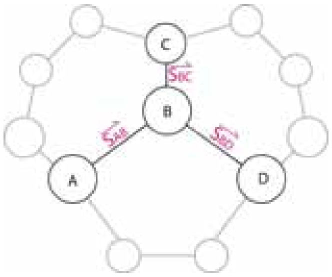

Figure 57: Distributed finite-element analysis model.

Networking

In order to send a packet, a node first sends a *request to send* to its neighbor and waits for a *clear to send* response from the recipient. The request to send contains the length of the packet to be sent so both nodes know when transmission is complete and can resume their activity.

A packet contains its length, its destination address, the address where it came from, a unique ID generated by the sender, a payload type, the payload itself and a cyclic redundancy check signature for error detection.

Table 1: Senspectra networking packet structure.

Byte 1	Packet Length (n)
Byte 2	Destination Address
Byte 3	From Address
Byte 4	Packet ID
Byte 5	Payload Type
Bytes 6..n-2	Payload
Bytes n-1..n	Cyclic Redundancy Check (CRC)

There are currently 6 different payload types that perform different functions. *Ping & Pong* are used for network discovery, *Request Strain & Send Strain* is used for exchanging sensor data between nodes, *Set Color* is used to set a node to a specific color and finally *Ack* is used by a node to acknowledge that it received a particular packet to the sender.

Table 2: Payload definition.

Payload Type	Byte value
Reserved	0x00
Ack	0x01
Ping	0x02
Pong	0x03
Request Strain	0x04
Send Strain	0x05
Set Color	0x06

On a higher level, Senspectra's networking algorithms were designed to accomplish three specific tasks: peer-to-peer communication, topology discovery and centralized control of individual nodes by a host computer.

The peer-to-peer networking scheme is used for communication between neighboring nodes. It is used by individual nodes to agree on sensor readings but can be extended to propagate control messages or requests in a viral manner. This layer is critical to the entire networking architecture because it is the only available mechanism for nodes to communicate with one another. There is no common bus in Senspectra so for non-neighboring to communicate with each other, packets have to be routed using a dynamic routing table.

This topology discovery uses the peer-to-peer layer to map the topology of the network from any point in the structure. It uses a ping/pong mechanism whereby ping requests propagate virally throughout the structure and pong replies return to the sender using the shortest path possible recorded by the virally propagated ping. This functionally is currently implemented in the G*host node* (GUI host node) that connects the constructed structures to a PC, but could be invoked by any node in a completely decentralized fashion.

The topology discovery allows the creation of a dynamic routing table by the computer. The computer can use this table to quickly address individual nodes, requesting specific sensor data or commanding a transition to a particular color value.

Simulated Annealing

The Ghost node directly feeds data to Senspectra's GUI-based simulation model through USB. In the model, every physical node is a thermally excited particle connected to its physical neighbors by springs. As the temperature drops in the overall system, the particles reach a geometrical equilibrium that resembles the physical model created by the user. The virtual model can then be used to run computationally intensive simulations and feed the results into the physical nodes to have them display appropriate colors.

We have seen rare cases of ambiguity when mapping the network topology into a 3D geometry. We therefore have provided the user with GUI tools to obtain the right geometry. The *Excite* function injects thermal energy into individual nodes. This causes the nodes to vibrate thereby augmenting the chances that they would reconfigure into a lower thermal minimum model that would better represent the physical model. In the case the excite tool fails to generate the appropriate model, the S*huffle* function reinitiates the simulation where all nodes regain a random position in the virtual space and slowly stabilize.

Usability Implications

Collocated Visualizations of Otherwise Invisible Parameters

From the informal user testing I did, people recognize several benefits in the ability to have visualizations on the models they constructed. Most users appreciated the direct reaction of their models to mechanical stresses and iterative design seemed to be encouraged by the real-time collocated simulations. While the collocation of the input and the output seemed to elicit wander and encourage reflection, I believe that the instantaneous reaction of the models to manipulation was also a crucial factor in the interaction process.

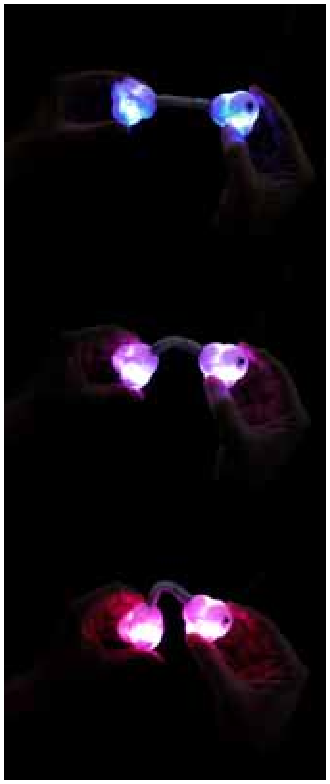

Figure 58: Simple Senspectra construction that changes color showing the mechanical stress between two nodes.

Incorporating Materiality in the Control-Feedback Loop

While Glume focused in making a soft modeling interface, we did not take advantage of the malleability of the medium to its full potential. The softness of the Glume units was *digitally meaningless*. This is something I was aware of when designing Senspectra and it reflected in the control mechanisms implemented in the flexible joints. In Senspectra's case, the malleability quickly became one of the most important interaction modalities. Generally, neophytes are quick at understanding the relationship between applying forces to Senspectra models and their change in color. In those regards, the haptic affordance of Senspectra seems very successful.

Limitations of the Senspectra Design

Resolution & Density

Senspectra experiences the same resolution problems I had encountered with Glume. Although I managed to reduce the physical size of the Senspectra nodes, the density of the nodes in a structure was decreased significantly due to the size of the joints.

While Senspectra suffers from the low resolution of its output modality, users can still obtain valuable information from deforming their constructions. It would certainly be more enlightening to have a modeling interface with a higher granularity, where users can create intricate models that possess finer sensing and output mechanisms.

As silicon footprint gets smaller, I believe that Senspectra-like interfaces will appear in our everyday lives and make use of limited computational power densely distributed across new materials and interfaces. Butera's 'paintable computer' or Senspectra's *sculptable computer* have huge potential as architectural substrates, human-computer interfaces and expressive artistic media.

Centralized Control

The current centralized control mechanism that uses the Ghost node presents potential scalability issues. As the amount of nodes in a construction grows larger, there could be a networking bottleneck that would cause the performance of the system to decrease substantially. One viable, but complex solution would be to redesign the system with a hierarchical architecture where multiple Ghost nodes get connected between the central computer and *branch* nodes. The branch nodes could then route the information to *leaf* nodes. Such a networking scheme would create a topology where nodes are at a maximum three hops away from the central computer. It would, on the other hand, create a usability nightmare, since the network would no longer be transparent to the users. Users would need to create structures made of Ghost nodes, branch nodes and leaf nodes simply for networking purposes and put a major part of their attention trying to network their creations rather than focusing on more important aspects of the construction.

For this reason, I believe that a decentralized architecture remains the most viable solution for the Senspectra system. I do not think the centralized control mechanisms should be removed from the system. However the central computer should not have to address individual nodes, but rather send high-level commands that propagate virally in the constructions to indicate to the nodes what function they should perform.

Chapter 6: Future Work

Senspectra currently suffers from the fact that we have not yet manufactured enough primitives to truly experiment with building complex structures to iteratively explore how modifications to the structures affect their response to mechanical stress.

Nevertheless I am proposing in the following sections how some additions to the Senspectra interface could reaffirm its potential as a 3D modeling interface.

I also suggest several application domains that could benefit from a tangible interface in the likes of Senspectra.

Senspectra Additions

Other physical controllers, different than the current modeling primitives, could help users to better interact with the Senspectra system and could potentially lighten the role of Senspectra's GUI to offer a more tangible interface. Such additions could offer more sophisticated input modalities in the form of binary or continuous controllers that could be mapped to specific functionalities in the system.

Similarly, probing tools could be a good solution to the resolution issue that currently faces Senspectra. Users could probe specific areas of their models to obtain high-resolution data on that region in the model.

Input Devices

Object modifiers could be introduced in Senspectra much the same way I had envisioned with Glume and could be used as physical handles to digital parameters. Currently, Senspectra users interact with their models by assembling/disassembling primitives and by deforming them. Introducing objects modifiers in the system would allow them to perform more complex tasks such as the selection of nodes. A *nodes selector* tool could be used as such to then enable users to assign specific material properties to a subset of nodes in a construction and subsequently visualize the effects. The selection process would have to use different metaphors than the traditional 2D selection metaphor used by most GUIs. A node selector in the form of a paintbrush or a flashlight would probably be more meaningful in Senspectra's case.

Object modifiers would also be helpful to freeze the state of current structural stresses in a model so users can spend time analyzing them without applying constant force on the structure. This *snapshot* tool could simply take the form of a push button embedded in one of the nodes in the structure. That way, users could easily toggle between an exploratory mode and a frozen mode directly from the physical model.

Another potential candidate for Senspectra would be a *record & playback* functionality with which users could record the stresses encountered by the model when deforming it at high speeds of in intricate ways and then playback the visualizations on the model at various speeds. Such functionality could be implemented in the form of a record button much like the snapshot tool and a playback jog wheel that would enable the speed and direction of playback.

Probing Mechanisms

Coupled with the new input modalities, *probes* could potentially alleviate Senspectra from the complementary GUI by offering users a set of specialized probing mechanisms that interface to the model directly. A probe could serve as a high-resolution interface to obtain detailed information on the state of a specific node. Probes would certainly benefit from having a high resolution display embedded in them to facilitate the visualization of the various parameters. They could take the form of a handheld pointing device coupled with a display or take the form factor of a magnifying glass to emphasize the zooming metaphor.

Application Domains

During Senspectra's design process, I developed several interaction scenarios in which its interactive potential could be used at its best. I ended up focusing on structural analysis as a first proof of concept, but I believe the following application domain could equally profit from a constructive assembly such as Senspectra.

Elastic Stability

The ability to visualize the resonance of a structure made of Senspectra primitives would be an easy iteration on the current system that takes advantage of sensing and output modalities already implemented. Users could record the stresses generated by high frequency oscillations and flutter in a structure they created and then 'playback' the recordings in the form of visualizations on the structure itself at lower speeds. It would take advantage of the collocation of the model and the visualization and would offer an accessible interface for people to experiment in an iterative way.

It is crucial in most constructions to have an accurate estimation of the frequency ranges that may cause free oscillation. Dynamical loads such as wind or moving vehicles which have their frequencies' range close to the ones of the construction can be precarious. If the frequency of a dynamic load coincides with one of frequency ranges of the construction's proper oscillations, then a danger of resonance appears which may cause the collapse of the construction [29]. Senspectra could be a good modeling tool to visualize micro and macro oscillations in a structure and play them back at various speeds for later analysis.

Figure 59: The collapse of the Tacoma Narrows Bridge in 1940 due to wind-induced vibrations.

Thermophysics

The system architecture of Senspectra makes it well suited for viral message propagation in a decentralized fashion much like cellular automata. The propagation of messages in a model made of Senspectra primitives is well suited for applications that require the flow of parameters through a 3D medium. One such example is thermal conductivity where the transmission of heat through a structure depends on the materials it is made of. The morphology of the structure is also important in analyzing thermal conductance because of the radiation or the diffusivity effect [45]. Senspectra nodes could easily change color dynamically to show its users how heat propagates through their construction over time.

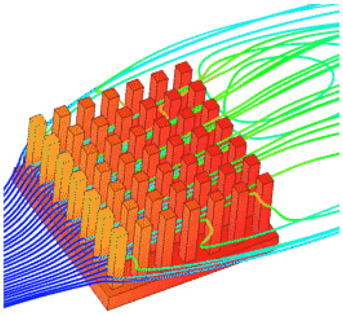

Figure 60: Heat dissipation simulation using the ACUSIM software[02].

Another important subject in thermophysics is the structural reaction of a material when faced with changes in temperature. Again Senspectra could be a good modeling tool for visualizing these changes with dynamic color gradients and iterative design could be a helpful mechanism in learning how to overcome issues related to heat transfer and dissipation in industrial design.

Solid-State Physics

A solid-state physics simulation tool could be a good application that shows the potential of Senspectra for modeling systems at different physical scales. The crystal lattice of silicon is made of a regular arrangement of atoms in a tetrahedral conformation. Senspectra nodes could be used to create a regular crystal lattice as a tool for chemists and physicists to visualize the flow of electrons in the lattice.

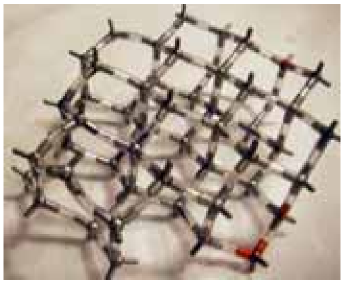

Figure 61: Plastic model of the atomic structure of a silicon crystal.

An absolutely pure silicon crystal is an insulator. When impurities are introduced in the crystal lattice (called *doping*), minute amounts of free electrons become available for electricity to flow through it. The impurities can generate electron surplus (n-type doping) or electron deficits (p-type doping). By combining p regions and n regions in a silicone crystal one can create semiconductors with drastically different behaviors.

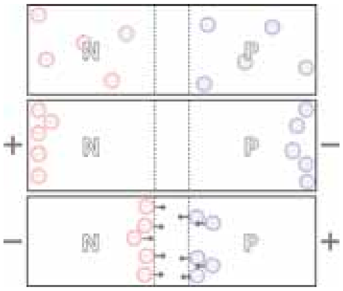

Figure 62: Diode. Top: no voltage applied; Middle: reverse bias (no current flow); Bottom: forward bias (current flows freely).

Users could *dope* parts of the model with various agents using the *nodes selector* tool and modify the nature of atoms at specific areas on the lattice. This would cause the model to become more or less conductive and conform into a diode configuration or an n-channel transistor, MOSFET, JFET, etc. Users could then apply a difference of voltage potential to the Senspectra model at various locations and directly visualize the flow of electricity and the transfer of charges in their model.

Protein Folding

The understanding of how proteins fold in space still remains an unsolved problem in computer simulation. Scientists have yet to find appropriate algorithms that accurately model how atoms in a protein attract and repulse each other. Current fold-recognition algorithms often produce only partially correct models where regions in the models remain misfolded [26]. Experienced chemists and biologists however have a general idea of how a protein will arrange itself in space. Senspectra could serve as a bridge between unassisted computer simulations and chemists' experience in a scenario where chemists would model a protein made of an extended set of Senspectra primitives and arrange the protein in space according to what they think the morphology should be. The Senspectra centrally-controlled simulation engine could show in real-time the interactions between the individual atoms and change their color to indicate to the users high and low energy points and give them insights on how they should modify the morphology of their protein to reach maximum stability.

Figure 63: Cartoon diagram of the folding of a protein solved by X-ray crystallography [26].

This is a form of human-based computation where the computational process of finding the protein morphology employs humans to perform parts of the workload. The algorithms take advantage of human capabilities to solve problems based on their experience coupled with their acute visual and tactile senses [43]. Following this description, I believe a modified version of the Senspectra modeling kit would have huge potential to help people better solve and understand the protein folding problem. Senspectra's sensing and output modalities would be very appropriate for molecular-based systems where the measurement of forces between atoms is relevant and simple visual cues can help its users significantly in finding the appropriate configurations for their system.

Art

I believe there lies potential beneath Senspectra for it to be used as an expressive medium. Senspectra creations could become light sculpture onto which people can program light patterns and sequences by bending its joints and recording pushes and twists on the sculpture that get translated into colorful animations.

A simple record and playback functionality could be easily implemented with object modifiers. Users could record light animations on their creations and playback single recordings or superpositions of recordings on the creation.

Another interesting feature for Senspectra would be to extend its primitives set with *plastic joints* as opposed to elastic joints. Plastic joints could be malleable just like their elastic sibling but would hold their shape when being deformed. This would introduce the concept of continuous controllers with 'plastic' memory in the Senspectra interface. Users could make deformable creations out of nodes and plastic joints that would have a structural memory.

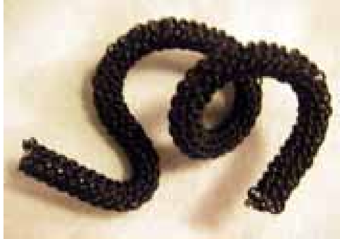

Figure 64: Plastic Joint prototype. This cable is embedded with titanium wire, a ductile metal that retains its shape when deformed.

Closing the Loop: Digital Materiality

Senspectra showed that the physical properties of a constructive assembly can be modified by users to control digital parameters in an interface. Senspectra only touched a small subset of possible interactions with malleable computing media. There definitely is space for extensive exploration with the contextual meaning of deforming objects and I hope to see more projects exploring these ideas in the future.

Looking ahead, how could digital processes control the physical properties of digital manipulatives? Past projects such as Senspectra or Topobo [39] have managed to make manipulatives embedded with computational power that change color or move in space depending on certain external factors. These projects unfortunately have a low granularity which makes them appear more as a set of discrete nodes rather than continuous media. Materials that exhibit different plastic properties under specific circumstances such as magneto-rheological fluids could be an interesting first step to create computationally-embedded materials with dynamic malleability.

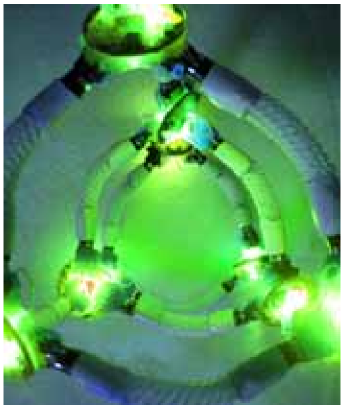

Figure 65: An architectural-scale structure could be made of Senspectra-like primitives and offer new types of interaction with more immersive interfaces.

This digital materiality could be applied to other types of human-computer interfaces. By integrating these technologies into building materials, one could foresee several new ways with which humans could interact with large-scale, immersive digital systems. But most importantly, by embedding computational power into such materials, the materials themselves would serve multiple digital/physical purposes. These reactive, self-conscious materials could truly be the dawn of the computational ubiquity praised by so many of us.

References

01 Abelson, H. & al. “Amorphous Computing”. *AI Memo 1665*. (1999)

02 ACUSIM Software. © 1994-2006.

03 Aish, R. “3D input for CAAD systems”. *Computer-Aided Design*, 11(2):66-70 (1979)

04 Anderson, D. & al. "Tangible Interactions and Graphical Interpretation: A New Approach to 3D Modeling", *ACM SIGGRAPH* (2000) pp. 393-402

05 Asundi, A., Sajan, M. “Dynamic Visualization using a Digital Camera”. Asian *Society of Visualization*, Chiba, Japan. (1994)

06 Bhushan, N. and Rosenfeld, S. “Of Minds and Molecules: New Philosophical Perspectives on Chemistry”. Oxford University Press. (2000)

07 Brosterman, N. “Inventing Kindergarten”. New York: Harry N. Adams Inc. (1997)

08 Brave, S., Dahley, A., Frei, P., Su, V., and Ishii, H. “inTouch”. *Conference Abstracts and Applications of SIGGRAPH '98 , Enhanced Realities* (1998), ACM Press, pp. 115.

09 Brownell, B. “Transmaterial”. *Princeton Architectural Press*, New-York (2006) pp. 9-10.

10 Butera, B. “Programming a Paintable Computer”. *PhD Thesis*, MIT Media Lab (2002)

11 Calver, G. & al. “Applications of Modern Automated Photoelasticity to Industrial Problems”. *Insight: Journal of The British Institute of Non-Destructive Testing*, Vol 44, No 4, pp224-228 (2002)

12 Coker, E. and Filon, L. “A Treatise on Photoelasticity”. Cambridge University Press, London. (1931)

13 Design Science Toys Ltd. “Tensegritoy” Tivoly, NY, 1994.

14 Eddaoudi, M., Kim, J., Vodak, D., Sudik, A., Wachter, J., O'Keeffedagger, M., Yaghi, O., “Geometric requirements and examples of important structures in the assembly of square building blocks”. PNAS (April 2002) pp. 4900-4904

15 Frazer, J., “An Evolutionary Architecture”. *Architectural Association*. (1995)

16 FroebelUSA.com © 2004.

17 Gaver, W. “Technology Affordances”. *ACM SIGCHI*. (1991)

18 Gershenfeld, N. “The Nature of Mathematical Modeling”, *Cambridge University Press*, 1999, ISBN 0521570956. pp. 162-164.

19 Gibson, J. “The Ecological Approach to the Visual Perception of Pictures”. *Leonardo*, Vol. 11, No. 3. (1978)

20 Gillet, A., Sanner, M., Stoffler, D., Olson, A., "Tangible Augmented Interfaces for Structural Molecular Biology". *IEEE Computer Graphics and Applications* (2005) pp. 13-17

21 Gorbet, M., Orth, M., Ishii, H., "Triangles: Tangible Interface for Manipulation and Exploration of Digital Information Topography". *Conference on Human Factors in* Computing systems (1998)

22 Haneman, D., "Surface Structures and Properties of Diamond-Structure Semiconductors". *Phys. Rev. 121*, 1093–1100 (1961)

23 Hart-Davis, D. © 1998.

24 Ishii, H. and Ullmer, B., "Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms". *Conference on Human Factors in Computing Systems* (March 1997), ACM Press, pp. 234-241.

25 Kellogg, R. "Demonstrating Structural Behavior with Simple Models". Published by the Author. (1994)

26 Kosinski, J. & al. "A "FRankenstein's Monster" Approach to Comparative Modeling: Merging the Finest Fragments of Fold-Recognition Models and Iterative Model Refinement Aided by 3D Structure Evaluation". *PROTEINS: Structure, Function, and Genetics,* 53:369–379 (2003)

27 Kurowski, P., "Finite Element Analysis for Design Engineers". *SAE* (2004) p.10

28 Lertsithichai, S., Seegmiller, M., "CUBIK: A Bi-Directional Tangible Modeling Interface". *Conference on Human Factors in Computing systems* (2002)

29 Leschenko, A. "The Fundamental Structural Mechanics of Elastic Systems: Oscillations and flutter". *URSS*, Moscow, Russia. (2006)

30 Mark, R. "Experiments in Gothic Structure". Cambridge, MA. *MIT Press* (1982).

31 Merkle, R., "Molecular building blocks and development strategies for molecular nanotechnology". *Nanotechnology 11* (2000) pp. 89-99

32 Montessori, M. "The Montessori Method". New York: Frederick Stokes Co. (1912)

33 Montessori N' Such, Inc. © 1997-2005.

34 Norman, D. "Affordances, Conventions and Design". *Interactions 6*, ACM Press. (1999)

35 Norman, D. "The Design of Everyday Things". New York, Doubleday. (1988)

36 Papert, S. and Harel, I. "Constructionism". Ablex Publishing Corporation. (1991)

37 Paradiso, J.A., Lifton. J., and Broxton, M., "Sensate Media - Multimodal Electronic Skins as Dense Sensor Networks". *BT Technology Journal*, Vol. 22, No. 4, October 2004, pp. 32-44.

38 PATRIOT Engineering Company. © 1979 – 2006.

39 Raffle, H., Parkes, A., Ishii, H. “Topobo: A Constructive Assembly System with Kinetic Memory”. *Conference on Human Factors in Computing Systems* (2004).

40 Resnick, M., Martin, F., Berg, R., Borovoy, R., Colella, V., Kramer, K., and Silverman, B. “Digital Manipulatives: New Toys to Think With”. *Proceedings of CHI ’98*, ACM Press (1998)

41 Roseman et al. “The chaperonin ATPase cycle: mechanism of allosteric switching and movements of substrate-binding domains in GroEL”. *Cell 87*, 241-251. (1996)

42 Schodek, D. “Structures” - 3rd edition. Prentice-Hall. (1998) pp. 541-543

43 Sims, K. “Artificial Evolution for Computer Graphics”. *Computer Graphics*, 25(4), SIGGRAPH'91. (1991)

44 Tautenhahn, L. © 1999.

45 Touloukian, Y. & De Witt, D. “Thermophysical Properties of Matter data series”. *IFI/Plenum*. (1970)

46 Ullmer, B. and Ishii, H. “Emerging Frameworks for Tangible User Interfaces”. IBM Systems Journal, Vol. 9, Nos. 3&4, 2000, pp. 915-931.

47 Watanabe, R., Itoh, Y., Asai, M., Kitamura, Y., Kishino, F., Kikuchi, H., "The Soul of ActiveCube - Implementing a Flexible, Multimodal, Three-Dimensional Spatial Tangible Interface". *ACM SIGCHI International Conference on Advances in Computer Entertainment Technology* (2004) pp. 173-180

48 Web document. http://swiss.csail.mit.edu/projects/amorphous/

49 White, J. “Rana Sylvatica Eggs”. © 2006.

50 Wikipedia contributors. “Photoelasticity”. *Wikipedia, The Free Encyclopedia*, http://en.wikipedia.org/w/index.php?title=Photoelasticity&oldid=49278096. (2006)

51 Yim, M., et al. “Rhombic Dodecahedron Shape for Self-Assembling Robots”. *Xerox PARC SPL TechReport P9710777*, 1997.

52 Zimmerman, T., “Personal Area Networks, (PAN): Near-Field Intra-Body Communication”. *IBM Systems Journal 35*, pp.609-618 (1996).