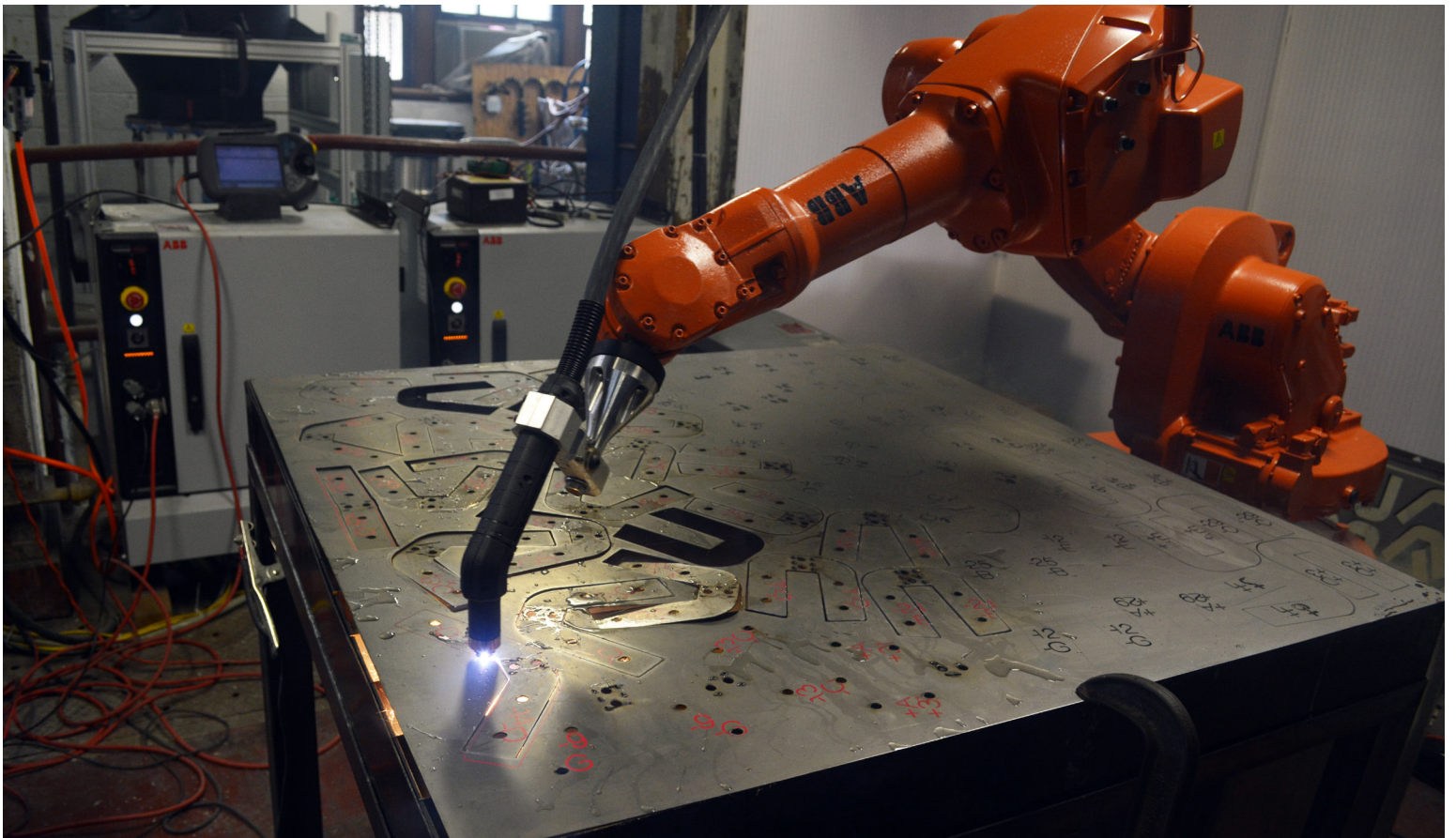


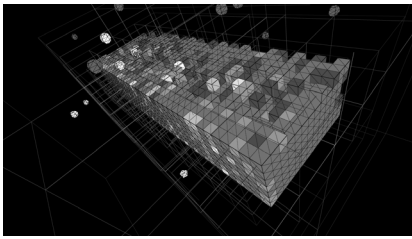
MICHAEL JEFFERS & JORDAN PARSONS
RECURSIONISM

05-09-2013

**OUR THESIS IS THE
DEVELOPMENT, STUDY AND
INTEGRATION OF A SYSTEMIC
COMPUTATIONAL METHOD
WITH DIGITAL FABRICATION.**



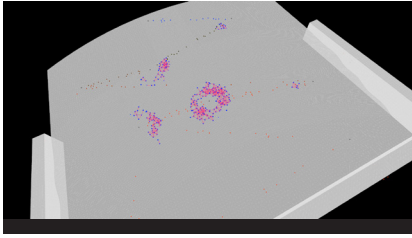
**SYSTEMIC COMPUTATION IS
THE CONTROL OVER THE
RELATIONSHIP BETWEEN
DISCRETE COMPUTATIONAL
FUNCTIONS AND SUBSYSTEMS.**



thesisWindErode

WindErode was an early attempt to make a tool which dealt with low level data, a point cloud, to form it and imbue more information into via the simulation of a natural process. The script simulates additive and subtractive forces on a given point cloud based on wind direction and values.

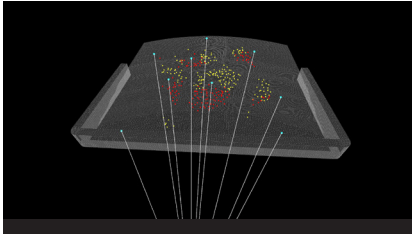
Environment: Java/Processing
Category: Point Cloud
Input: Points
Output: Points



thesisCirculation

Circulation acts on a point cloud with a high level abstraction of human movement behavior. It has a series of destination points, and agents who move between those points, pushing the base point cloud around based on the generated circulation path.

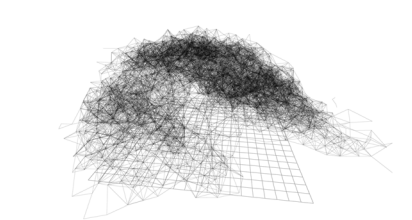
Environment: Java/Processing
Category: Point Cloud
Input: Points
Output: Points



thesisProgram

Program has a series of locations which act as points of interest, and a system of agents who act as audience members or users. The user agents move inside the point cloud pushing it and themselves around while the cloud collapses around them. The system continues this process until it finds a balance between space and enclosure.

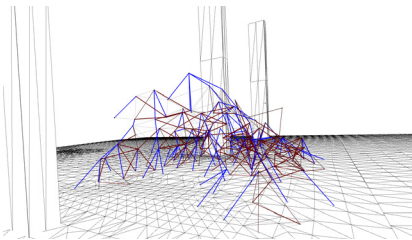
Environment: Java/Processing
Category: Point Cloud
Input: Points
Output: Points



thesisConnectionMatrix

Connection matrix takes a finalized point cloud as a text file input, and given a minimum and maximum connection radius, the program generates a brute force connectivity matrix between all the points, it exports a unique set of connections as a simple pairing of points.

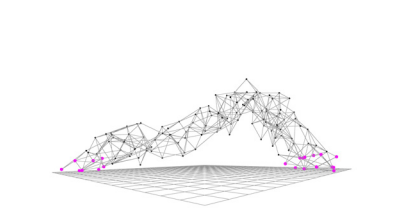
Environment: Java/Processing
Category: Structure
Input: Points
Output: Raw connectivity



thesisWeightingSystem

The weighting system takes the raw connectivity from connection matrix and gives every connection a value based on angle tolerances, force transfer and planar angle. The system then removes the connections with the lowest values, and repeats the test. This process continues until a state of equilibrium is reached at which point the connections are exported again for future use.

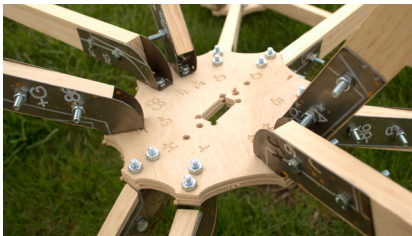
Environment: Java/Processing
Category: Structure
Input: Raw connectivity
Output: Evaluated connectivity



thesisPhysics

Physics acts as a simple spring based test, it allows us to see quickly and easily if a given connectivity may fail. It is not a perfect simulation, and can easily be confused, but it is a safeguard, and would ideally be more integrated into the weighting system.

Environment: Java/Processing
Category: Structure
Input: Evaluated Connectivity
Output: Visual test of stability



thesisJoinery

Joinery takes the refined connectivity and generates geometry, labels and several different text outputs from that. It is the core of our current process. It uses several steps to handle joint generation, and relies heavily on the thinning that is done earlier in the weighting system. Its outputs are used to cut the plates, fingers, render images, and generate labels.

Environment: Grasshopper/C#
Category: Fabrication
Input: Connections & Points
Output: Connection export, final joint geometry, meshes.



thesisSolar

Solar works through the refined connectivity file, and finds triangles bounded on all three edges by a valid connection. This list of triangles is useful for prefabrication and potential surface mounting. This program then evaluates each triangle for solar exposure at a given location and timespan.

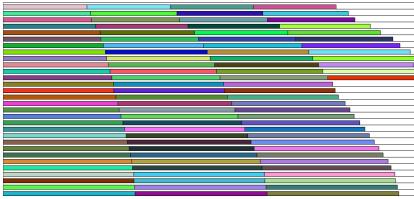
Environment: Java/Processing/Grasshopper
Category: Surface
Input: Connections & Points
Output: Triangles w/ Weighted values



thesisLabels

This program generates a series of QR coded labels that are intended for each connection in the system. The labels have QR codes, which link to an image of the connection in place in the whole structure, the connection ID, the node and finger ID of each end of the connection.

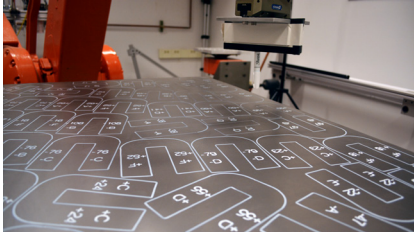
Enviroment: Java/Processing
Category: Fabrication
Input: Connections
Output: Printed labels, QR Codes & GIF renders.



thesisBinPacker

BinPacker uses a sorted bin packing algorithm to prepare the connections for milling. It attempts to pack them onto the smallest number of boards possible and then generates the cut geometry to mill the connectios.

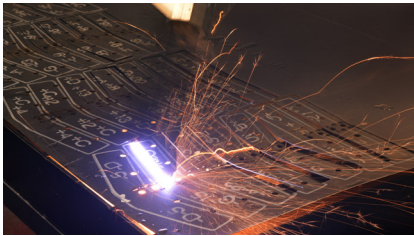
Enviroment: Java/Processing/Grasshopper
Category: Fabrication
Input: Unsorted Connections
Output: Sorted Connections & CNC Cut Geometry



thesisRobotPainter

Using output from joinery, RobotPainter uses Grasshopper and HAL to generate RAPID files. The IRB 4400 then uses these files and an oil paint marker to draw out finger labels and profiles on a sheet of steel, so that when plasma cut, the steel parts are labeled and can be tracked.

Enviroment: Grasshopper/HAL/RAPID
Category: Fabrication
Input: Cut Geometry
Output: Painted steel sheets for plasma cutting.

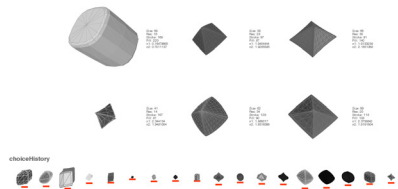


thesisPlasma

Using output from joinery, Plasma uses Grasshopper and HAL to generate RAPID files which are used by an IRB 1600 to plasma cut sheets of 20GA cold rolled steel to form fingers for the structure.

Enviroment: Grasshopper/HAL/Rapid
Category: Fabrication
Input: Cut Geometry
Output: Robot cut files.

thesisDecisionMaker

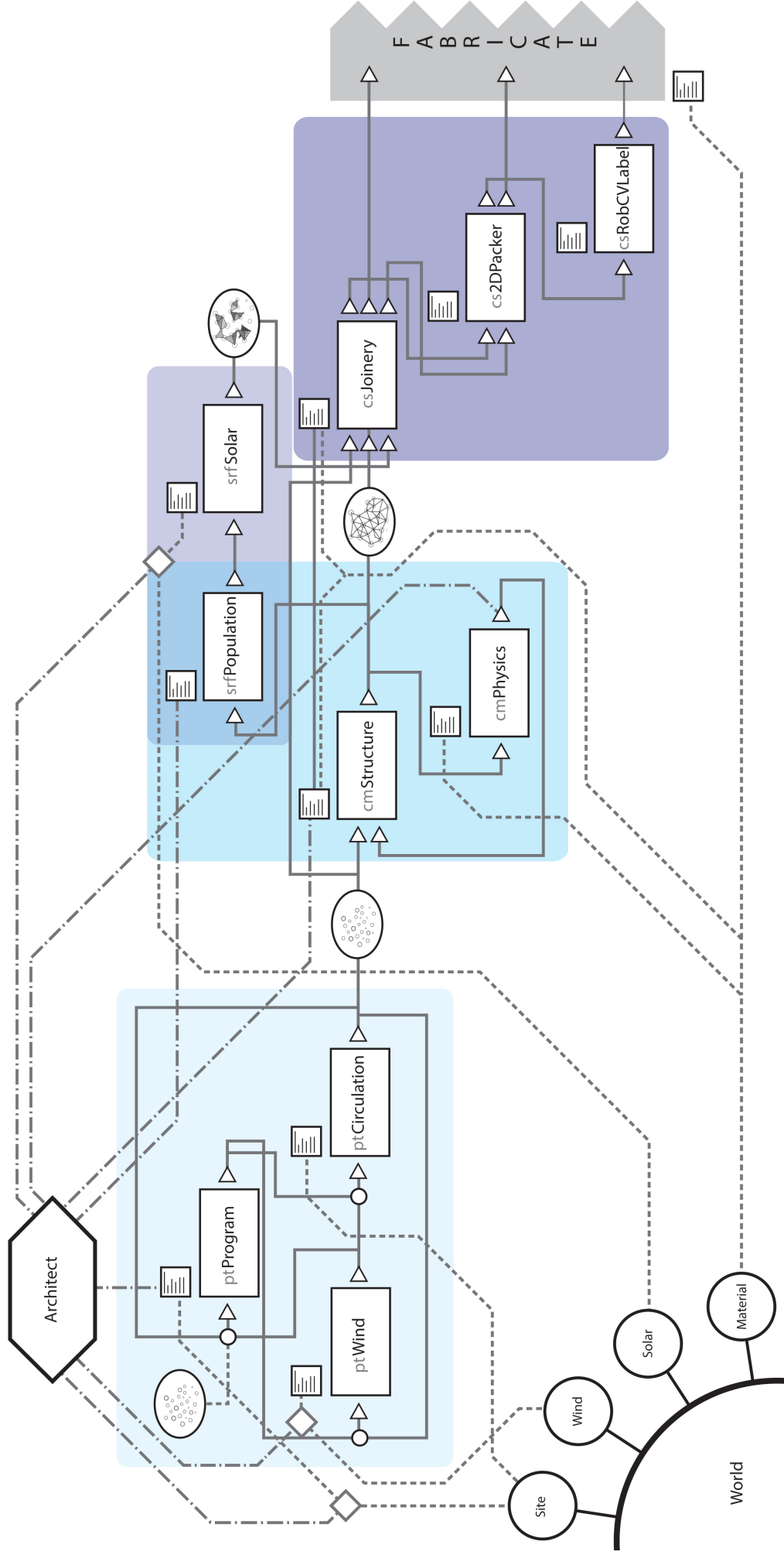


thesisDecisionMaker

DecisionMaker was an attempt at aesthetic evaluation through predictive selection. It represents a brief detour, and the tip of a much larger discussion about the quality of spaces generated via computation and the quantification of qualitative information.

Enviroment: Java/Processing
Category: Qualitative Testing
Input: Meshes/User Choice
Output: ?

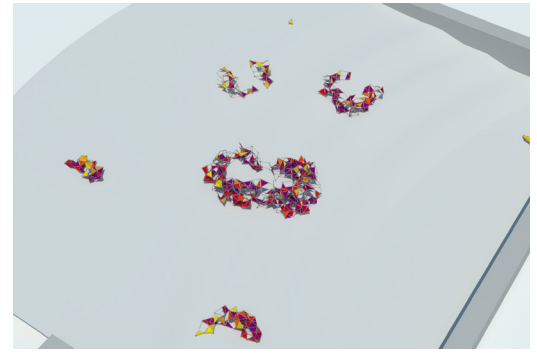




FIRST GENERATION SYSTEM DIAGRAM

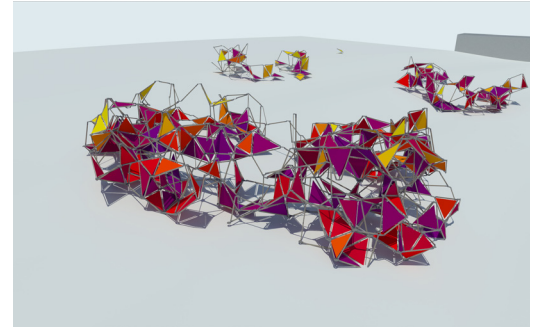
Instantiation as Architecture

The built instantiation is experienced independently of the logics of its generation. This disconnect leaves the instance in an ambiguous state between architecture and object. Altering qualities and factors in its siting, scale and tectonic execution could reposition it in a more architectural state.



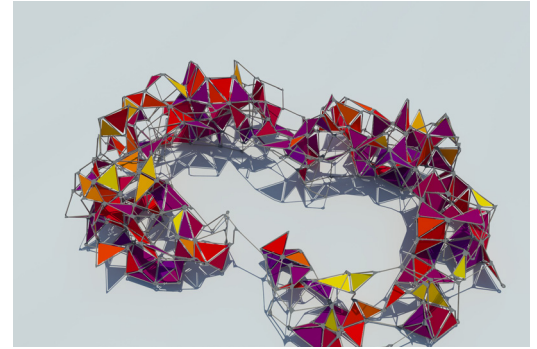
A Priori vs Systems

Systemic computation preserves the connection between intent and execution. Erroneous results force refinement of the system, and not the resultant form. Whereas traditional design practice amalgamates conflicting logics into a single object, obscuring the connection of intent to execution.



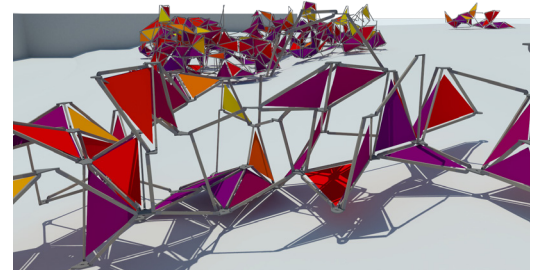
Craft and Authorship in Computation

Craft in computation is based on knowledge of low level functionality in involved processes. This knowledge and the computational logic employed in the execution of the system transfers the authorship of the designer to any resultant instantiations if the design intent and logic are congruous.



Construction Process

Systemic computation engaged with digital fabrication allows for the explicit linking of digital and physical information. Construction logics and nomenclatures emerge based on this relationship, controlling the degree of on site decision making.



First Generation System

The toolkit was designed as an open framework that can accept and interfaces with a multiplicity of tools, types and functionalities allowing for the incorporation of additional data sets, conditional gates, analytic and evaluative subsystems. It can also be understood as a component of a larger framework that operates across different scales, engaging with higher level logics.

Systemic and Non-systemic Computation

Non-systemic computation exists as the realization of a priori forms through the use of discrete computational functions without consideration for the direct interplay and communication between them. Systemic computation arises through the direct articulation and acknowledgement of the relationship between discrete computational functions and subsystems.

