Speculative Structures

Reanimating latent structural intelligence in agent-based continuum structures

Joshua M. Taron
Laboratory for Integrative Design, University of Calgary, Canada
josh@synthetiques.net

Abstract. The potential afforded by the open search spaces of both agent-based models and evolutionary engines have given architecture yet another set of computational tools to play with, yet more often than not and with some cause, they are used in isolation from one another. This research explores the set of techniques and results of having combined swarm formations, FEM software and an evolutionary engine within a parametric modeling environment such that they induce materially intelligent and structurally viable swarmed formations. A set of protocols are developed for grafting these formations into the already-built environment, treating it as a resource to be accessed and exploited toward the production of novel morphogenetic results and architectural possibilities. Keywords. Interoperability; morphogenetics; evolutionary computation; swarms; FEA structural analysis.

INTEGRATIVE VS. INTEGRATED INTEROPERABILITY

Interoperability and Building Information Models (BIM) have become nearly synonymous terms as Industry Foundation Classes (IFC) have become a means for integrated design solutions. More specifically, data-rich IFC objects have established a standard so that information is not lost when moving between platforms. This integrated solution creates a problem however by excluding data sets that cannot read or generate IFC objects. Without having any problem with IFC progress and development, there is another line of investigation that I will call Integrative Interoperability that does not concern itself with IFC language and instead focuses on techniques of communication between software that might otherwise remain disconnected. This difference borrows from Branko Kolarevic’s (2008) disciplinary integrated vs. integrative distinction and applies it to software methodology. So rather that dealing with a defined and closed integrated model such as IFC, integrative interoperability speaks to the generative capacity to yield complexity by fluidly navigating across different software territories to discover and create a process, technique, or a product that is qualitatively new. Additionally, rather than framing integrative techniques as fighting the tide of disintegration, integrative interoperability partners with and exploits disintegration between multiple software as a fertile territory for new morphogenetic design opportunities.
The foundational premise of the research lies in integrating multiple and otherwise disparate data sets into a single morphogenetic, structurally viable assembly; or rather a discrete act of architecture. Integrative interoperability becomes a significant tool under such a model because it frees design from the totality of any single given software environment and lets each program read and generate information that it is specifically geared for. For example, 3D modeling environments can directly access and exchange information with multiple other software platforms so that the user can get spatial, aesthetic, structural and material feedback.

The advantages of integrative interoperability exponentially increase within evolutionary environments when virtual populations are culled and bred relative to explicit performance criteria. This not only enables design to engage with increasingly complex problems but it also allows new morphological and material behaviors to emerge in the process. For architectural design, this means that the meta-structure of the information networks becomes synonymous with defining performance criteria itself. As cities become more complex, our ability to provide equivalently complex building solutions inherently evolve with them provided we have integrative methods for capturing, integrating and producing emergent data sets.

STRUCTURALLY INTELLIGENT SWARMS
Most, if not all, architectural projections of swarm-generated buildings are imaged as static instances of otherwise dynamic processes. While architects are usually interested in building in general, this raises questions as to what information within the swarm continues to actively contribute to the development of any given project. The fact that a swarm becomes frozen at a single moment of its existence limits its behavior in the context of its “native” animate environment. However, the complexity of that instance does not dictate a single solution (and in fact resists single solutions), but rather it opens up a set of solutions given discrete constraints such as materiality, structural sizing and loading. This research accesses and makes use of latent information within “frozen” instances of a swarm in order to discover new structurally informed morphologies – in effect reanimating swarms in ways that would otherwise be impossible or undesirable in their original agent-based environment(s). Furthermore, the scalability of swarms should be of particular interest to architectural design as increasingly complex strategies of integration are clearly necessary if design is to sustain the metabolisms of urban environments (Weinstock 2008). This is consistent with various others’ calls for computational approaches to design that make use of planetary and even cosmologically scaled populations (Bratton and Metahaven 2011; Keller 2011).

At first glance, snapping a moment out of an active agent-based space might seem undesirable, but it actually opens up an opportunity to overcome (or at least bypass) a fundamental problem presented by swarms when attempting to integrate them with analytical software. Critical to a swarm’s functionality as a design tool is its ability to remain relatively computationally inexpensive. If agents within a swarm have to perform overly complex sets of individual calculations, models either become too slow and inefficient to run given a fixed number of agents or the number of agents must be reduced, thus decreasing the intensive capacity of the model. In other words, a swarm’s intelligence is directly related to how many agents it is able to sustain (more is typically better) and how efficiently it can develop solutions (faster is better). Augmenting flows of information should ideally yield more productive results rather than negatively disrupting the otherwise fluid circulation of information.

One instance of inefficiency and computational expense within swarms can be found when attempting to form feedback loops between active agent-based models and analytical software, in this case finite element analysis (FEA) software. The intended purpose of forming such a connection lies in the ability to incorporate structural performance into a swarm’s behavior. But structural calculations are computationally expensive as structural sizing,
materiality, connections and loading must all be accounted for. This is only compounded when dealing with large numbers of structural segments. Perhaps a more critical question lies in what structural logic any instance of a swarm should assume at all as addressed in previous investigations [1] that attempted to compress agent-based patterning into surface logics in order to integrate them into a single material construct.

With so many parts and such a range of complex interconnections, it becomes apparent that a uniform approach would likely fail to make use of the latent structural intelligence that a swarm has to offer. This is made more difficult given that in many cases, structural solutions are beyond intuitive or conventionally determinable means. By separating structural calculations and spatial swarming from one another, their own efficiencies can be maintained and productively mobilized against one another. However, in order to enable an evolutionary morphogenetic design environment, a method for advantageously [re]connecting them without sacrificing efficiency is needed. This is achieved by making use of Geometry Gym tools in Grasshopper, grafting swarm formation into architectural assemblies and running them through FEA-driven evolutionary feedback loops. This yields solutions that explicitly express latent structural intelligence of swarm formations in partnership with the already-built environment.

**Evo-morphogenetic structure**

The meta-structure of these investigations is defined through the explicit relationships between a series of different software platforms. Specifically, the work establishes integrative interoperability between initial non-parametric 3D geometries in Rhino, agent-based models in Processing, parametric environments in Grasshopper (including the Galapagos evolutionary engine) and structural analysis software (SAP2000 FEA) (Figure 1).

In geometric terms, the framework allows one to manipulate a series of points in Rhino which are actively linked to an agent-based environment (Processing) where those points function as particle emitters in a pheromone-flocking particle swarm optimization model. Particle courses are tracked and interpolated as NURBS curves in Grasshopper. This geometry is parametrically associated with material properties and structural sizing and iterated through the Galapagos evolutionary engine in a manner that mobilizes individual structural members against the global structural performance of the entire assembly. Evolutionary results are then sent back to the initial Rhino environment and evaluated for any other criteria (not addressed in this work).

By nesting an evolutionary feedback loop within the larger morphogenetic framework, specific attributes emerge without threatening the structure as a whole. The necessity for hierarchies (or at least their
resilient stability) within morphogenetic assemblies is often something that is overlooked despite the crucial role it plays in enabling emergent behaviors to express themselves (Hensel et al 2006). Given the dynamic nature of both material(s) and geometries, resilient network structure acts as an attendant “not in the sense of a spectator that simply observe a process as it unfolds, but rather as a constant or point of reference in relation to which variation is assessed” (Deleuze 1981). The framework then becomes a mechanism to produce variations that aggregate to form novel and identifiable behaviors. In this case, the framework aims at finding and fostering structural and material viability within geometric formations that otherwise lack such intelligence.

**Sensitivity-driven bending**

The first attempts at reanimating swarms focused on two related orders of interconnection needed to achieve structural viability. In the case of post flocking simulations, particles tend avoid one another and thus particle courses fail to come into contact with one another. In addition to the course curves, a connective layer of curves is produced by means of a proximity mesh through the swarm much the same way webbing connects chords to one another within a truss (Figure 2).

Alone, the proximity mesh serves as a minimal solution within the excessive redundancy of the swarm. However, the conceptual and aesthetic dissonance between these two layers is seen as undesirable and opens up a new line of questioning that focuses on how the minimal solution of the proximity mesh might be accessed so that the swarm might induce deviations from what otherwise ultimately operates as a single linear span between 2 points. Using a proxy geometry that approximates the swarm components, the geometry is broken down into differential lists in order to provide a parametric framework.

The first level of subdivision differentiates between agent path curves and proximity mesh curves. A point set is then distributed through each agent path curve based on a fixed frequency of length identifying pinned connection points for the proximity mesh curves (Figure 3).

These points are capable of shifting as the assembly deflects. End points of the agent path curves constitute a separate point set that serve as fixed connection points that will not shift as the assembly deflects. Each agent path curve is assigned a value for structural size with straight segments spanning from point to point within each respective curve. In the interest of limiting evolutionary variability within the high number of proximity mesh segments, curvature, structural sizing, and curve segmentation are compressed into a single parametric component (Figure 4).

The logic of the component is as follows: straight proximity mesh segments are restructured as single span 3-degree curves resulting in a set of 4 CV’s per curve. While the end points remain connected to their respective agent path curves, the remaining CV’s are each attracted to a nearest point condition found within a differentially scaled set of agent path curves. This process distorts the curves as to provide a way for the swarm, through its internal proximities, to produce locally driven curvature. Omni-directional curvature differentiates in magnitude on a per segment basis within the proximity mesh as it distributes through the agent path curves. The relative magnitude of induced proximity mesh curvature is managed through a single sensitivity value. As this value increases, so does relative curvature which in turn drives structural sizing and segmentation. The combination of agent path curve structural sizing and proximity mesh sensitiv-

![Figure 2](image-url)  
*Course connection diagram.*
While sensitivity-driven explorations focused on individually sizing every member in the swarm assembly, this round of testing takes a step back in order to articulate the performative advantages of replacing larger scale structural members with swarm-based assemblies that use smaller/lighter structural members. Inspired by Huang and Xie's evolutionary topology optimization of continuum structures that use displacement constraints (Huang and Xie 2010), we began by developing a parametric definition that would allow a single span beam to be evaluated for a specified deflection value (SDV) through SAP2000. By minimizing the absolute value between the deflection result and a SDV in Grasshopper, evolutionary iterations through Galapagos would yield a specific structural size that would approach the SDV as shown in (Figure 6).
Having established the necessary computational framework for our own evolutionary structural optimization, we began applying swarm assemblies in place of the singular structural member in an attempt to drive the size of the structural members down while maintaining the ability to achieve a range of SDV’s given an axial load of 1kN to put the assembly into compression. All members in the assemblies were assigned a uniform value for size and employed a simple proximity mesh to constrain the otherwise disconnected swarm courses to one another. Initial results are shown in (Figure 7). The results of these tests exhibit a desired correlation between lighter members and higher SDV. However, the tests raised questions over the intensity of the proximity mesh and the effect they were having over achieving deflection. Toward this end, the proximity mesh was sorted into a list that measured their lengths and put them into sets representing increments of 25% of the total population. The tests were run again to achieve the specified deflection values of .001, .01 and .1 of the overall beam span. In this series the population of connecting members was culled by 25% increments beginning with the longest members so as to understand the effects of decreasing structural frequency. Results that used only the shortest 25% of the population failed to manage the axial loads and as such their results were discarded noting the threshold for failure.

Two unexpected behaviors were expressed. First, even with the full population of proximity mesh structural segments in place, the longest members contained the highest stress loads. This was suprising in that we expected stress either to appear toward the middle of the span or in areas where other connections were not being made. Secondly, the test demonstrated the intelligence to size up the structural size of the members in order to achieve a deflection value that had previously relied on more parts throughout the assembly. We were not surprised to see shorter members demonstrate higher stress levels given the absense of additional structural members.

**Stress-driven Branching**

While the previous tests were inspired by subtractive methods in order to arrive at a structural equilibrium, efforts were made to develop a bidirectional
system where structure could not only be removed through a hard-kill process similar to those used in BESO methods (Huang and Xie 2010) when members fail to meet minimum stress levels but also added in order to target high stress areas and locally distribute their loads. Toward this end we modified a series of grafting protocols (Taron 2012) whereby agent-based morphologies structurally integrate into otherwise normative wall assemblies (Figure 8).

Given the presence of higher stress levels in longer members developed in the SDV tests, the branching grafts intend on distributing stress throughout the entire assembly. Additionally, these curve networks were translated into continuum meshes that simultaneously achieve dimensional depth through manifold volumes while minimizing the length of any structural member to the edge of a given face. FEA stress analysis on a series of increasingly complex formations reveal the successful distribution of stress through the assembly while maintaining constant loads (Figure 9). This is a promising discovery as a particular form can achieve specified loading and reduce structural sizing without having to revert to minimal geometric form. In other words, redundancy has the capacity to produce its own modes of efficiency that operate as an alternative to ‘conventional’ form-finding methods.

By iterating bi-directional branching through FEA informed evolutionary loops, an initial swarm formation can grow and decay such that it efficiently grafts into an existing structure and actively participates in distributing loads through the entire as-
assembly. Because the evo-morphogenetic framework remains intact, additional forces and geometries can be added or subtracted thus allowing the assembly to search for new equilibria (Figures 10 and 11).

CONCLUSIONS AND FUTURE WORK
Evolutionary morphogenetic tactics demonstrate real purpose for developing latent performance attributes in complex assemblies including swarm formations. While much discussion continues to take place revolving around the usefulness of swarms in architecture, this work articulates the potential value for any complex assembly subjected to evolutionary iteration when integrated with performance analysis software thus enabling novel solutions and morphologies to affect architectural objects and discourse.

Work has already begun toward fabricating these assemblies at a number of scales and with a range of materials and connection strategies. Presently the work has focused on populating planar surfaces but will likely yield new problems and opportunities when deployed through more spatial (multi-orientation, multi-surface) assemblies. Additionally the research would benefit from urban analysis that identified derelict or abandoned structures that could be revitalized and reprogrammed through these tactics. Rather than thinking of archi-

Figure 9
Stress distribution through curve network (above) and continuum structure (below).

Figure 10
Hard-kill stress distribution branching sequence.

Figure 11
Stress-generated continuum Structure assembly render.
tecture as always a new discrete object, these methods reposition it as an always already integrated part of the already built environment.

REFERENCES
