**Computational ADU Design: An Evolutionary Approach**

Informal preface:

The exact methodology that I am following en route to the results (both of which will comprise the largest chunk of my research paper) is still in flux. As such, I am writing informally before and after the dividers this week, which signify the bounds of the writing to actually be included in the paper. This is a means to translate informal work to formal work once the method is straightened out and the results become more clear.

1. *Introduction*
   1. *Housing crisis*

The United States is amid an unprecedented housing crisis, stemming from issues outside of housing, such as neoliberal cuts to social benefits spending and the increasing privatization of essentially every aspect of American life. The result is a rising population of unhoused peoples and the inability for many to become a homeowner or to even afford minimum monthly rent in many cities. Solutions to this dilemma are neither straightforward nor definite. However, as designers and architects, this problem can only be addressed at the symptom-level, while advocating at the root cause.

[expand slightly here]

* 1. *ADU as solution*

Seattle, Washington; Portland, Oregon; and Vancouver, British Columbia are exploring the use of accessory dwelling units (ADUs) as one method to combat surging housing prices. Use of ADUs is an effective means of increasing housing density without replacing single family housing zones with new multi family residential construction. Additionally, ADUs are often designed to be rented, generating supplementary income for the homeowner. Ten detached ADU (or DADU) designs are pre-approved by the City of Seattle and are available online to entice homeowners. However, a 2019 city survey shows that there are calls for an increased focus on sustainability and cost in the construction of ADUs(Seattle 2019). These out of the box designs do not offer the scalability or energy efficiency that an ADU designed specifically per site offers.

[expand here]

* + 1. *Existing ADU caveats*

Homeowner’s associations and other local organizations have historically fought back against any proposed density increases through zoning or other method. However, Seattle and the other aforementioned cities in the Pacific Northwest have succeeded in allowing for the construction of ADUs in recent decades. The pre-approved designs are free, but require payment of ~$1000 [check this] for approval and come with one significant downside of many other pre-designed structures- a lack of contextual design and individualization.

* + 1. *Computational tool as fix*

This research explores whether the use of genetic algorithms via shape grammar methodology can effectively optimize DADU plans to site context, increases building performance or further encourages construction. The proposed methodology begins by reading example site data from the city of Seattle including building and vegetation context from the Seattle GIS. Next, a genetic algorithm explores the design space for a viable floor plan solution based on a fitness function. This fitness function evaluates individual designs according to predefined traits. Traits to evaluate include window to wall ratio, insulation depth/type, ventilation strategy, and shading technique. Locating the correct combination of traits to minimize (or maximize) which results in a higher performance DADU is the desired outcome. Resulting designs will be analyzed and compared via energy performance simulation. The end-goal is to develop a computational tool using the aforementioned system to conduct automated site analysis, as well as parametric generation of DADUs.

1. *Literature Review*
2. *Methods*
   1. *Multi-objective search and optimization*
      1. *Genetic algorithms (via Grasshopper + Galapagos)*
   2. *Energy simulation (via Ladybug + Honeybee)*
3. *Results*
4. *Conclusion*
   1. *Future work* 
      1. *Benefit of a pure python tool*

Looking forward, moving on from the Grasshopper platform gives the benefit of non-reliance on developers to maintain the software in which the tool depends. Creation of such a tool in a singular programming language (in this case Python) further offers the ability to quickly and easily run the tool on a high variance of devices. In turn, this theoretically increases the rate of adoption by lowering the requirement to use the design tool. Additionally, Python is used within Rhino/Grasshopper (or a flavor thereof), provides many useful math and geometry libraries, and has options for injection into a web app (Flask and Django).

* + 1. *Limitations of Galapagos/Grasshopper workflow*

Utilizing Galapagos and Grasshopper offers a ‘sandbox’ environment to begin to understand genetic algorithms and to arrive at tangible design solutions faster than using a home-brewed algorithm. Galapagos offers many fewer input parameters and a much narrower scope in which to define the fitness function. In its out-of-the-box form, Galapagos only accepts number sliders as input and can only optimize integers and floats- in reality, there is not means in which to define a true fitness function, only numerical values to target.

1. Bibliography

Li, Li. 2012. “The Optimization of Architectural Shape Based on Genetic Algorithm.” *Frontiers of Architectural Research* 1 (4): 392–99. <https://doi.org/10.1016/j.foar.2012.07.005>.

Rodrigues, Eugénio, Adélio Rodrigues Gaspar, and Álvaro Gomes. 2013. “An Evolutionary Strategy Enhanced with a Local Search Technique for the Space Allocation Problem in Architecture, Part 2: Validation and Performance Tests.” *Computer-Aided Design* 45 (5): 898–910. <https://doi.org/10.1016/j.cad.2013.01.003>.

Seattle, Office of Planning & Community Development. 2019. “Pre-Approved Plans for Accessory Dwelling Units Survey Results.” Seattle Office of Planning & Community Development. <http://www.seattle.gov/Documents/Departments/OPCD/OngoingInitiatives/EncouragingBackyardCottages/OPCDPreApprovedDADUSurveySummary.pdf>.

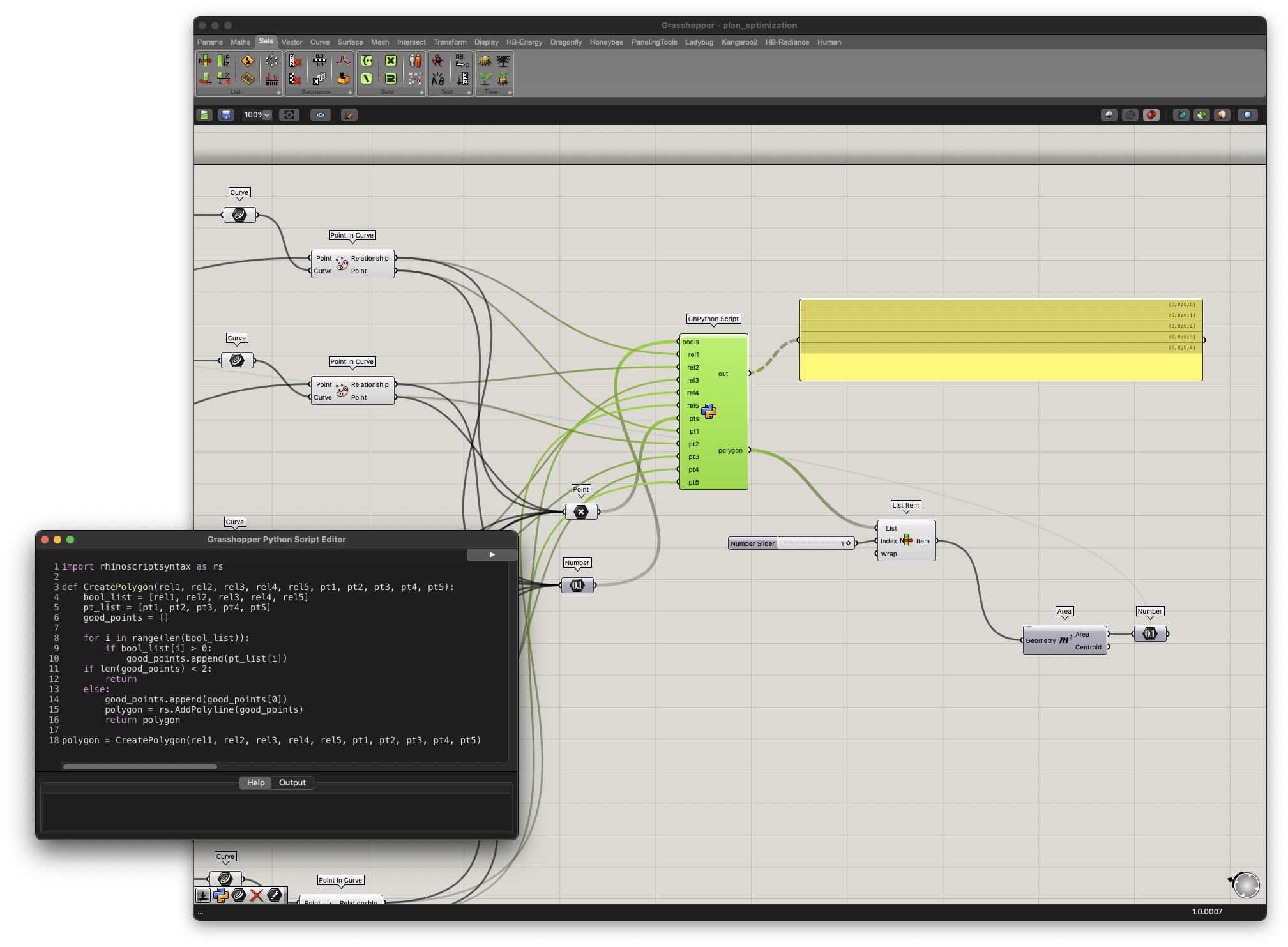
Wang, Likai, Kian Chen, Patrick Janssen, and Guohua ji. 2020. *Algorithmic Generation Of Architectural Massing Models For Building Design Optimisation - Parametric Modelling Using Subtractive and Additive Form Generation Principles*.

Wang, Likai, Patrick Janssen, Kian Chen, Ziyu Tong, and Guohua ji. 2019. “Subtractive Building Massing for Performance-Based Architectural Design Exploration: A Case Study of Daylighting Optimization.” *Sustainability* 11 (December): 6965. <https://doi.org/10.3390/su11246965>.

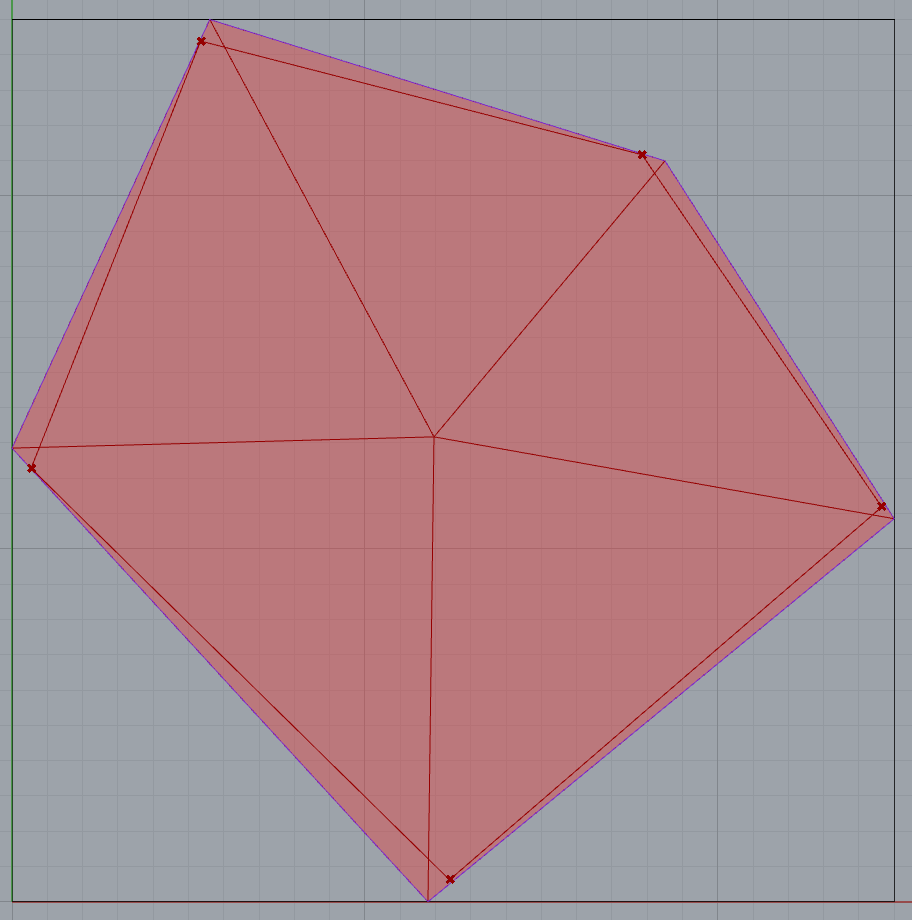
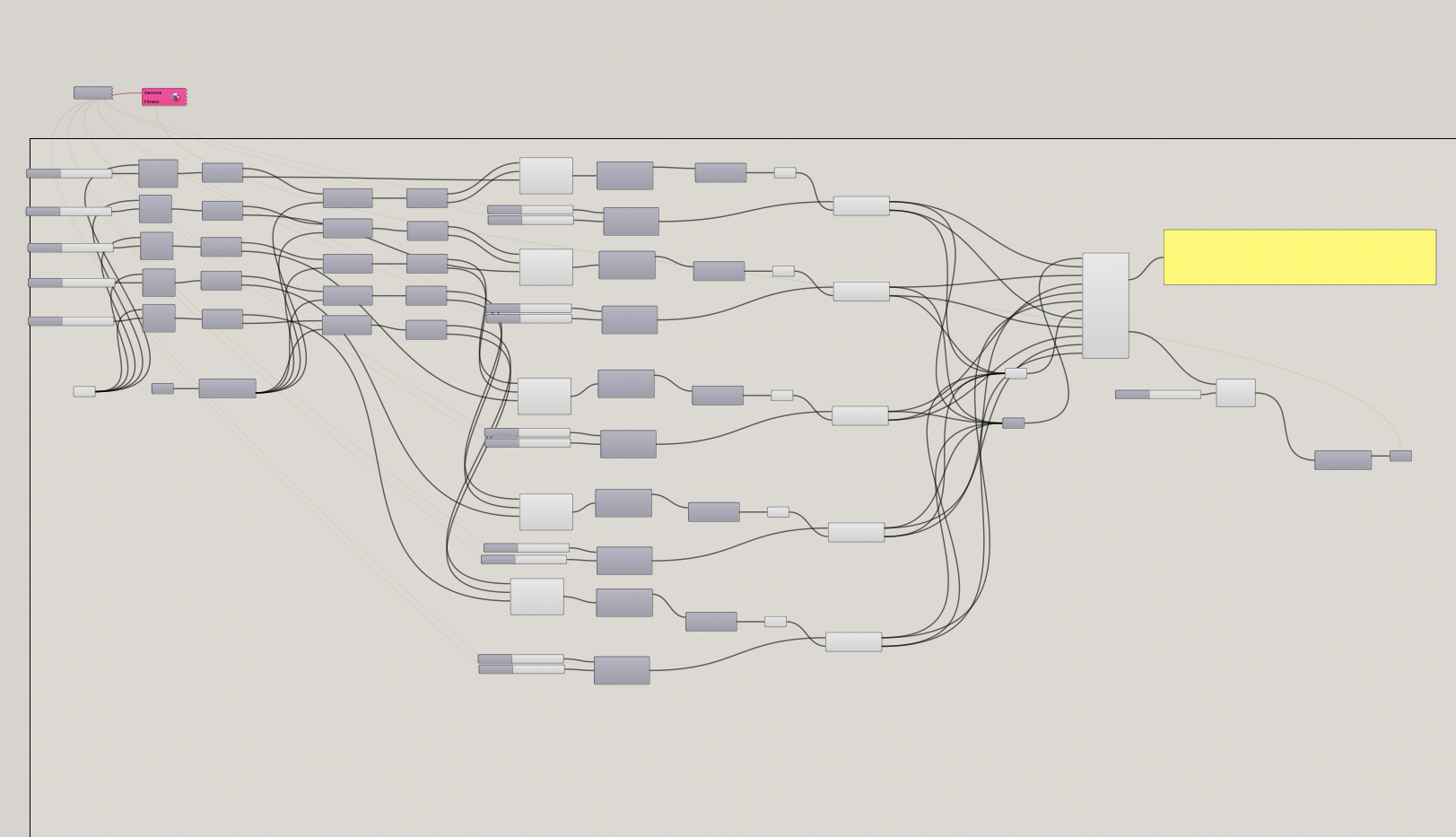
*From Draft 1:*

I am not yet to the place I wish to be for my draft, as I have been focusing much more exclusively on the more pragmatic, programming side of this research. However, I understand how it is beneficial to write while doing, I just have not fully adjusted to that workflow yet. I was stuck for a few days with my genetic algorithm but have since moved past the errors with the help of Tomas and am back on track with the development of the tool.

Since the previous draft, I have come to the conclusion that learning the architecture of genetic algorithms while learning to utilize genetic algorithms within architecture is not a task I can complete within 10 weeks. Therefore, I am relocating my focus unto the outcomes, rather than the process at this point and time.

Assorted \*\*draft\*\* images of work done so far: 

Above: implementation of portions of genetic algorithm code directly into Grasshopper

Above: simple 30 minute test of Galapagos evolutionary solver to maximize space on n-gon site by connecting vertices to centroid and bounding a collection of n points within each cell.

Above: Middle section of Grasshopper definition contains all number sliders which are used as variable inputs to the genetic algorithm within Galapagos

Final paper will include data visualizations, process diagrams, and actual design outcomes, in place of illegible screenshots of the Grasshopper script. :)