THE IMPLEMENTATION OF PROGRAMMABLE ARCHITECTURE

Wireless interaction with dynamic structure

K. HOTTA\(^1\) and A. HOTTA\(^2\)
\(^1\) Keio University, Fujisawa, Japan
\(^2\) University of Tokyo, Tokyo, Japan
khotta@sfc.keio.ac.jp; hotta@archi1.t.u-tokyo.ac.jp
\(^2\) Kagawa university, Kagawa, Japan
sharisharishari@hotmail.co.jp

Abstract. True adaptability in architecture necessitates both dynamic hardware and software with the potential for continually renewable forms capable of all possible variations necessary for changing demands and conditions, without having to resort to one theoretically optimal solution. PA consists of both autonomous and subservient systems that maintain a constant homeostasis within its contained environment. The information flow between the Genetic Algorithms (GA) and user input prompts this hybrid system to generate the consequent, ever-changing physical form, while continuously optimizing it for environmental stimuli. This paper proposes a smart strategy for a human interactive-cybernetic architecture in the context of K. Hotta’s Programmable Architecture (PA), aimed at enhancing GA’s capabilities in continuous self-modelling and facilitating human-computer interface.

Keywords. Human-computer interaction; user interface.

1. Introduction

In recent years, the key word ‘adaptability’ has increasingly been used in the field of emergent architectural design such as in the late book titled ‘Adaptive Ecologies’ (Theodore et al, 2013). Also, one of the most established architectural conferences, ACADIA (The Association for Computer Aided Design in Architecture), titled its 2013 symposium ‘Adaptive Architecture’. There are many approaches to making architecture (both buildings and sys-
tems) adaptive and sustainable. Most contemporary architecture uses scientific approaches based on mathematics, physics, and computational tools and through experimentation is able to achieve higher levels of adaptability against ever changing circumstances. However, in considering the keyword ‘adaptability’ in architectural design, it is critical to discuss what adaptability is for. In the previous paper of PA (Hotta, 2013), environmental adaptability is the centre of discussion. The aim is to develop architecture with higher environmental adaptability, and to develop an architectural design methodology that will address issues of shape, systems, and devices that constitute the building.

2. State of the art

In the architecture field, there is a seminal book in terms of participatory planning methods; an architect-less architectural system in the vernacular tradition (Rudofsky, 1964). More recent, with reference to temporal design linked with the idea of emergence, it is worth looking at the Fun Palace Plan (Price, 1961) and Plug-in City (Archigram, 1964) in the United Kingdom, and the Metabolism Movement in Japan in the 1960s and 70s (Lin, 2010). The idea of emergence in the previously mentioned contexts is that the architect and designer designed ‘systems’ rather than depicting static images.

On the other hand, there are many precedents in the engineering field. For example, Cybernetics, advocated by Norbert Wiener (1961), was a synthetic academic discipline that dealt with the matter of control and correspondence in a system like an organism or a machine. As an extension of this line of thought, Control Theories describe the methods in engineering and mathematics, which aim to control dynamic behaviour. The usual objective of control theory is to control a system. It attempts to adjust the system behaviour through the use of feedback via a controller, further developed as P, PI, PID (Proportional, Integral, Derivative) (Minorsky, 1922).

In the robotics field, the ‘smart’ robotic system has its origins in Social Robot (Walter, 1951), where the first speculation of multiple interacting robots was conceived. Another early trial of ‘smart’ robot can be referred to in ‘Vehicles’ (Braitenberg, 1984), in which evolutionary thinking was implemented in the wiring itself. Subsumption Architecture (SA) is the reactive idea used in artificial intelligence (AI), a term and field invented by R. Brooks (1986), which was developed to determine robot behavior. Swarm robotics is a relatively recent methodology for a multiple robot system (Yim et al, 2000), based on swarm intelligence observed in animal behavioral patterns. These past and seminal works ultimately have led to Evolutional Computing (Sims, 1994) and Evolutional Robots (Lipson, 2000).
3. Background

As a case study of the PA, flexible kinetic-tensegritic structures are shown in previous thesis (Hotta, 2013). The hardware for this proposal is an accumulation of self-sufficient machines that is dedicated to the actions of sensing, calculating, and actuating. As a case study for this thesis, a kinetic canopy that is organized using tensegrity-based components of variable forms is proposed. This architectural robot is actuated by shape memory alloy (NiTi) instead of tensile wire, and its control is handled electrically by micro controllers Arduino (Banzi et al, 2005–). A physical model of this machine has been built at a one-to-one scale and user-tested via mobile devices such as a smart phone.

![Figure 1. 4th Generation, Physical Model (referring from PA, Hotta, 2013).](image)

The software for this proposal consists of a hybrid control system, which attempts to minimize the difference between the desired objective values and the measured values. This is a combination of automatic responses and user manipulations in order to achieve a faster and higher degree of adaptation. Utilizing the versatility of GA, multiple user inputs are proposed to partially substitute for its purely random mutations (usually GA use random digits for mutations).

![Figure 2. Drawing of PA (referring from PA, Hotta, 2013).](image)
This resolves GA’s shortcomings, namely protracted calculation time, lack of adaptability to a fluctuating objective function which represents the ideal condition at any given time, and the ability for ad hoc responses when the system experiences usage overload or random environmental fluctuations. Incorporating the user input, the system can respond rationally to actual conditions unanticipated by the GA. Therefore, the user can concurrently control the system locally, to reflect individual preferences, and contribute to the global optimization and increased efficiency of the system as a whole.

The exposure experiment-1 executes with 4 candidates, which consists of Fixed, Pre-optimised (with same algorithm), Realtime-Kinetic (Proposed PA system), Kinetic ultimate (ideal score). The aim of these experiments was to prove that the proposed kinetic roof adapts most effectively to sun exposure over time providing the best result. This set of GA experiments demonstrate the result that normal GA does not work effectively for dynamic situations. The score for the real-time kinetic roof was worse than the pre-optimized roof’s score. The following paragraph discusses the possible reason for this and suggests potential solutions. Usually GA is designed and used for static problems. The key issue to focus on in order to improve the score is how to compensate for the lack of initial information in a moving landscape. Two possibilities are: Insert sensor information into the GA’s ongoing calculation as an interruption input.
On those contexts, Experiment-2 has examined. While the previous model only allowed one input, namely the environmental input from the sun, this model has 2 inputs, one the top-down environmental input similar to the GA, and the other being a bottom-up user input explained below. Obviously, this is not a pure GA system, but rather a human-interactive evolutionary algorithm. Instead of ‘random’ mutation, user input is used to seed the next generation. This will help reduce the calculation time.

Five different executions are shown as graphs; bigger number is better. Each graph shows a unique ‘GA+Human’ method, the three candidates: (Fixed, Real-time-Kinetic, Kinetic-Ultimate) are the same. Each ‘GA+Human’ method’s score depends on the proficiency of the human and the timing (number of interventions.). The conclusion of this experiment is ‘GA+Human’ not always result better than normal GA (Realtime Kinetic) but, sometime could beyond it. This shows if the human manipulations were appropriate, this system can exceed normal GA system. Discrepancy points between the first model and this model are there because those are made by different settings but this is not a main point of this paper. The Grasshopper
experiment raised issues and then it made hypotheses that have been tested in this chapter. In this chapter, the mathematical model (Processing code) confirmed the same minimum set of points; showing that normal GA does not work effectively where dynamic fitness calculations are required. Here a model using Interactive Evolutionary Computation (IEC) was designed and tested. While reaching similar conclusions these chapters could be said to be logically independent.

4. Problem statement

Reviewing previous PA research (K. Hotta, 2013), several shortcomings raise up. Here is the problem statement. Human-machine hybridized systems yield relatively better results in the simulation process. However, the large quantity of sliders and lack of intuition in the interface hinders usability and efficiency.

5. Physical implementations

In order to get higher score, several systems are tested below.

![Figure 6. PA model version-1 and its system.](image)

The early model of PA, which was made by Arduino in ‘standalone mode’, used the PC strictly as a power source. Though receptive and relatively quick, it was limited to simple responsive or at most branching systems written by the syntax ‘if’, ‘for’, ‘case’ etc. The crucial shortcoming is there is no human interface.

![Figure 7. PA model and its system version 2.](image)
The wireless system was tested with TouchOSC (hexler.net, 2012), which was originally designed for MIDI controlled surfaces. Yet, its limitations were revealed in its inability to combine two sets of data.

Figure 8. PA model and its system version 3.

This implementation is the minimum set for PA, which can afford hybrid data flow, perform automatic calculations and respond to human input. However, the wireless interface with an end device, such as a smartphone, could not be created in reality.

Figure 9. PA model version 4.

Referring to Bongard’s idea of continuous self-modelling (Bongard, 2006), PA was utilized for simulating precise and physics based virtual models (Fig.10), which was used for previous PA experiments. Ultimately, this will result in the resistance against environmental catastrophic change such as earthquakes, typhoons, etc... Additionally, by selecting outputs instead of expressing every genome could be expected to reduce both energy and cost.

6. Experiment and result

Fig. 11 illustrates the comparison between the two different types of user interfaces. The ‘simple sliders’ (pictured left, Fig. 10) which control each genome independently, yield less performative results, as oppose to the ‘smart panel button’ (pictured right, Fig. 10), which connects 6 clouds of actuators into a single component, thus demonstrating that the panel control is more intuitive and corresponds accurately with the virtual modelling space.
Figure 10. Two types of Interfaces; Left is sliders, Right is panels.

Figure 11. The result of experiment, bigger is better.
7. Conclusion

Though advances in computational simulations have continuously advanced in recent years, their integration in human-computer interface and dynamic adaptability often lend architectural models to remain as static outputs. The smarter UI strategy is proposed as a step towards an adaptable architecture in two ways. First, by facilitating the computer-human interface in an intuitive modeling process, and second, by favoring a continuous dataflow between the virtual model and physical model, rather than a simple GA setup.

Acknowledgements

I would like to express my thank to Mr. M. Weinstock and Dr. G. Jeronimidis, as an academic supervisors, I. Wako and Y. Fujimaki as drawer. Also great thank for Union foundation as a sponsor.

References

Hotta, K.: 2013, Programmable Architecture, AA School, UK.
Rudofsky, B.: 1964, Architecture without Architects, Univ of New Mexico Press, Mexico.
Sims, K.: 1994, Evolving Virtual Creatures, Siggraph ’94, Orlando, USA.