

**An Analysis of an Autonomous Smart House as an Organism:
An Alternative Pattern of Organization**

By Nicholas Jewkowicz

**A thesis submitted to the Faculty of Graduate and Postdoctoral Affairs in partial
fulfillment of the requirements for the degree of**

Master of Design

In

Industrial Design

Carleton University

Ottawa Ontario

©2016, Nicholas Jewkowicz

Abstract

This paper examines the current state of smart homes and proposes an alternative model based on biomimicry. It is argued that a house that is modeled on a basic living organism will be more efficient for the inhabitants, and more effective to insulate them from the unpredictable effects of climate change in the near future. By using an organism as a model, the house will be able to self-organize its systems, and adapt to both its inhabitants as well as environmental perturbations. This can be accomplished with the use of sensors and actuators in a decentralized configuration with artificial life programming. Since organisms are autonomous by definition, off-grid housing systems are infused to create a new housing model that is zero-emission, zero-waste, and can serve as a model for other forms of infrastructure at greater scales.

Keywords: Smart House, Self-Organizing Systems, Complex Adaptive Systems, Artificial Life, Wireless Sensor Networks.

ACKNOWLEDGEMENTS

Valerie Daley, my saviour!

Supervisor Dr. Thomas Garvey and co-supervisor Dr. Stephen Fai, thank you for your direction.

Also thank you to Dr. Lois Frankel, Dr. Won Joon Chung, Dr. Anthony Marini, Dr. Anil Somayaji, Dr. Nicola Santoro, and Dr. Dwight Deugo

And thank you to my fellow classmates, and for the support and encouragement of friends and family.

TABLE OF CONTENTS

Abstract	i
Acknowledgements	ii
Table of Contents	iii-iv
List of Figures	v
1.0. INTRODUCTION	1
1.1. Background Context.....	3
2.0. LITERATURE REVIEW	6
2.1. Autonomous Homes	7
2.2. Smart Homes	8
2.3. Responsive Architecture	9
2.4. Why a Living Cell?	11
2.5. Living Systems	12
2.6. Autopoiesis	13
2.7. Dissipative Structures	14
2.8. Artificial Life (AL)	16
2.9. Self-Organizing Systems	17
2.9.1. Feedback in Self-Organizing Systems	18
2.9.2. Fitness and Adaptation	19
2.10. Complex Adaptive Systems (CAS)	20
2.10.1. Designing and Programming a CAS	23
2.10.2. Adaptation in Agents	26
2.10.3. Genetic Algorithms (GA)	27
2.10.4. CAS as Models for Distributed Control	27
2.11. Engineering Complexity	28
3.0. METHODS	31

4.0. RESEARCH.....	32
4.1. Introduction to the Complex Adaptive House	32
4.2. Floor plans of the Complex Adaptive House with sensor and actuator legends	34
4.3. Case studies	43
Case study 1: Earthships	43
Case study 2: eHouse 2000	46
Case study 3: North House	48
Case study 4: MavHome	55
Case study 5: Ada	61
Case study 6: The Medina Smart Village	64
Case study 7: SANETs	69
Case study 8: SuOC	73
Case study 9: SodekoVS project	78
Case study 10: SOTA	81
4.4. Scenarios	85
Scenario 1: Drought	85
Scenario 2: Cold Dark Winter	86
Scenario 3: Storms	87
Scenario 4: Adaptation to Human Behaviour	88
5.0. CONCLUSIONS.....	88
REFERENCES	93
APPENDIX.....	99

List of Figures

Figure 1: Diagram of Literature Review	7
Figure 2: Trilogy of Life	14
Figure 3: Various Network Typologies	33
Figure 4: Basement Floor Plan	34
Figure 5: Main Floor Plan	36
Figure 6: Upper Floor Plan	37
Figure 7: The Complex Adaptive House Exploded	38
Figure 8: Diagram of What Systems are Coupled to Others	39
Figure 9: Sensor/Actuator Diagram With Microcontrollers and Adaptation	41
Figure 10: South Rendering	42
Figure 11: Overview of Case Studies	43
Figure 12: Alberta Earthship	44
Figure 13: eHouse 2000 Renderings, Eight Different Angles	47
Figure 14: A Depiction of Human-Home Interaction	50
Figure 15: Concept Maps of the Different Factors for a Smart Intervention Framework	51
Figure 16: Circadian-Rhythm-Aware Lighting at Night	52
Figure 17: Electricity and Heating Cost Savings With Occupancy Detection	53
Figure 18: MavHome Abstract Architecture	59
Figure 19: MavHome System Architecture	60
Figure 20: (a) MavHome Argus Sensor Network, and (b) MavHome Apartment Without Sensors	60
Figure 21: Overview of Ada Architecture Organized Into Conceptual Layers	62
Figure 22: People Interacting With Ada.....	64
Figure 23: Observer/Controller Architecture	74
Figure 24: Generic Observer/Controller Architecture	76
Figure 25: Observer/Controller Realization	77
Figure 26: SodekoVS Reference Architecture	80

1.0. INTRODUCTION

"People don't plan to fail, they just fail to plan." Emerson

The hypothesis being explored in this paper is that due to the advent of climate change, the most advantageous form of smart house will be an autonomous house whose functionality is modeled on that of a living organism. There is a growing interest in, and necessity for, off-grid housing in light of the increasing unpredictability of environmental fluctuations. According to the National Oceanic and Atmospheric Administration, the Earth's temperature has been rising steadily for the past fifty years (www.noaa.gov). This is because our atmosphere contains 42% more carbon dioxide than before the industrial revolution (www.davidsuzuki.org). The results of this warming include increasingly erratic fluctuations of atmospheric and oceanic patterns, bigger storms with greater frequency, increased floods, heat waves, droughts, and pests (www.davidsuzuki.org, IPCC, 2014 p.15). The predicted effects on human society include power outages, increased food prices, increased energy prices, flooding and water contamination, as well as rising sea levels with increased acidity (IPCC, 2014 p.13-15) Climate change will affect all areas of the globe disproportionately; some countries will be hit worse than others, but no part of the world will be exempt (IPCC, 2014 p.11).

It will be argued that an autonomous house can insulate its inhabitants from most catastrophes, as well as not contributing any greenhouse emissions. It will be shown that a living organism as the model for a smart house is a better alternative model than the current standard.

The best way to insulate ourselves is for more of our infrastructure to be not only autonomous, but responsive and decentralized. Biomimicry, which may prove to be essential in this regard, is the study of natural principles to provide solutions and innovation through technology (Benyus, 1997). Nature has provided a plethora of models that have been perfected through thousands or

millions of years of evolutionary trial and error. Many of these principles can now be reverse engineered and understood. This thesis is not the first to think about biomimicry in architecture; Ahmar (2011) specified six principles for architecture to successfully mimic an organism: “1) it must be dependent on sunlight, 2) it must optimize the system rather than its components, 3) it must be attuned to and dependent on local conditions, 4) it must be diverse in components, relationships and information, 5) it must create conditions favourable to sustained life, and 6) it must adapt and evolve at different levels and at different rates” (Ahmar, 2011). This paper will build upon that foundation.

In 1979 the New Alchemy Institute organized a conference entitled, *The Village as Solar Ecology: A Generic Design Conference*. The hypothesis or proposal surrounding this conference was:

“The blending of architecture, solar, wind, biological and electronic technologies with housing, food production, and waste utilization within an ecological and cultural context will be the basis of creating a new design science for the post-petroleum era” (Todd, 1984, p.12).

Many aspects of this research paper are drawn from these early pioneering ideas, but the juxtaposition with the model being proposed here is entirely new. This paper is partly a study in ‘ecological design,’ which is defined as “design for human settlements that incorporates principles inherent in the natural world in order to sustain human populations over a long span of time. This design adapts the wisdom and strategies of the natural world to human problems” (Todd, 1984, p. 1).

A large amount of research has been done in the fields of artificial life, self-organizing systems and complex adaptive systems. The design of ‘smart’ gadgets is also increasing in popularity, and appliances are becoming increasingly automated in ‘smart’ homes. Sensors and wireless

networks are becoming very commonplace and are gaining in complexity. Computer programs and algorithms have been invented that mimic Life's patterns and can replicate (on the computer screen) how living systems function and evolve. It is now possible to combine the principles underlying living systems with programming and building technologies.

1.1. Background Context

"Forests precede civilizations, deserts follow them." Chateaubriand

In the 1860s, physicist John Tyndall recognized the Earth's natural greenhouse effect and suggested that slight changes in the atmospheric composition could bring about climatic variations. In 1896, a seminal paper by Swedish scientist Svante Arrhenius first speculated that changes in the levels of carbon dioxide in the atmosphere could substantially alter the surface temperature by way of the greenhouse effect (Arrhenius, 1896). According to NASA, over 97% of the world's scientists are in agreement that climate change is both real, and the result of human-made greenhouse gases. Carbon dioxide levels are the highest they have been in 650,000 years. The global temperature has risen by 1.4 degrees F since 1880. Nine out of ten of the warmest years have occurred since the year 2000. 2016 is officially the warmest year on record. Arctic and Antarctic ice are shrinking at an alarming rate, which will result in sea levels rising and altering the ocean current patterns, which are directly linked to atmospheric currents. Sea levels rose 6.7 inches in the last century, however in the past decade, they rose almost twice that amount. The acidity levels in the ocean also increased by 30% since the industrial revolution (<http://climate.nasa.gov/>).

The planet is a complex dynamic and adaptive system. Systems can be analyzed in terms of either positive or negative feedback. A healthy living system is characterized as a stable negative feedback loop (<http://climate.nasa.gov>). Climate change is resulting in positive

feedback which is amplifying out of control. This materializes in the form of unpredictable weather shifts such as bigger, more destructive storms and temperature fluctuations, resulting in blackouts and property damage, increases in droughts and floods which destroy crops, which then lead to skyrocketing food prices. There is also the resulting introduction of pests (such as the pine beetle), which kill more trees as it spreads due to warming, resulting in increases in forest fires, which adds more CO₂ to the atmosphere, which contributes to further climate change. This warming melts the permafrost in the arctic, releasing the methane beneath which results in even more climate change (<http://climate.nasa.gov/>).

As climate change is classified as a ‘super wicked problem’, there is no central authority to effectively address it. An autonomous house that mimics an organism can provide insulation to its inhabitants from disasters on many fronts. In an autonomous house, all the primary elements for survival are accounted for: energy (electricity), HVAC, water, food, and waste removal. The inhabitants of an autonomous house can sit in relative comfort in the face of disaster. Brown, Conrad, and Beyeler (2012), concur that “climate change, and the impacts of climate change on the environment, population and engineered systems, is one of the problems that require a CAS approach for analysis and design of effective risk reduction actions.” CAS is short for ‘complex adaptive systems’ which is the proposed housing model being explored in this paper.

There has been a steady increase in smart house technology and technology in general, as well as a trend of increasing automation in the home environment (dishwashers, dryers, thermostats, etc). Ubiquitous computing, otherwise referred to as ‘the internet of things’ is entering the marketplace at an increasing rate, which will contribute to smart house functions and programming. Another advantage is that a smart house can be upgraded as newer technologies emerge.

The concept of using an organism as a model may be used at larger scales (buildings, neighborhoods, cities, countries), and in conjunction with the future 'smart grid.' It is less expensive (and stressful) to be prepared/insulated from disasters than to respond/rebuild after the fact. This sentiment is echoed by John and Nancy Todd (1994) who state that "in the evolving synthesis of biology and architecture, a neighbourhood could begin to function in a manner analogous to an organism. On the proposed block or neighbourhood scale, parts become symbiotic to the whole and the basic social and physical functions work together. The workings are felt and understood by residents, who live with and operate the components" (p. 116).

Humans are part of nature, but human dwellings and infrastructure are not integrated with their surrounding ecosystems. Ironically, technology can help accomplish this. When we analyze a living organism or system in terms of how it works, it may be able to provide a better model for an optimally efficient house for the residents to be completely self-reliant. Some of the scientific fields under examination in this paper are relatively new and groundbreaking, other studies have served as the foundation for these studies. Engineering complexity is the newest topic being explored, as it is still experimental. Although virtual simulations have been widely successful at this point, their application to real-world experiments are scarce. All smart house systems use centralized processing whereas the biological model requires decentralized processing and special programming. This paper will explore what is necessary to accomplish such a system.

"We've got the job of managing evolution now, whether by design or default, so we might as well get good at it." Julian Huxley

2.0. LITERATURE REVIEW

This literature review is organized into several areas of focus: it begins by discussing a few examples of both autonomous and smart dwellings. We then proceed to the more general concept of responsive architecture, which is gaining ground in various research labs. This will be followed by an examination of a cell (since it is the inspiration for the housing model) and how Life works biologically and physically, as well as a requisite detailing of self-organizing systems and complex adaptive systems. Finally, there will be an exploration of computation and simulation of these systems, and how to engineer such a system.

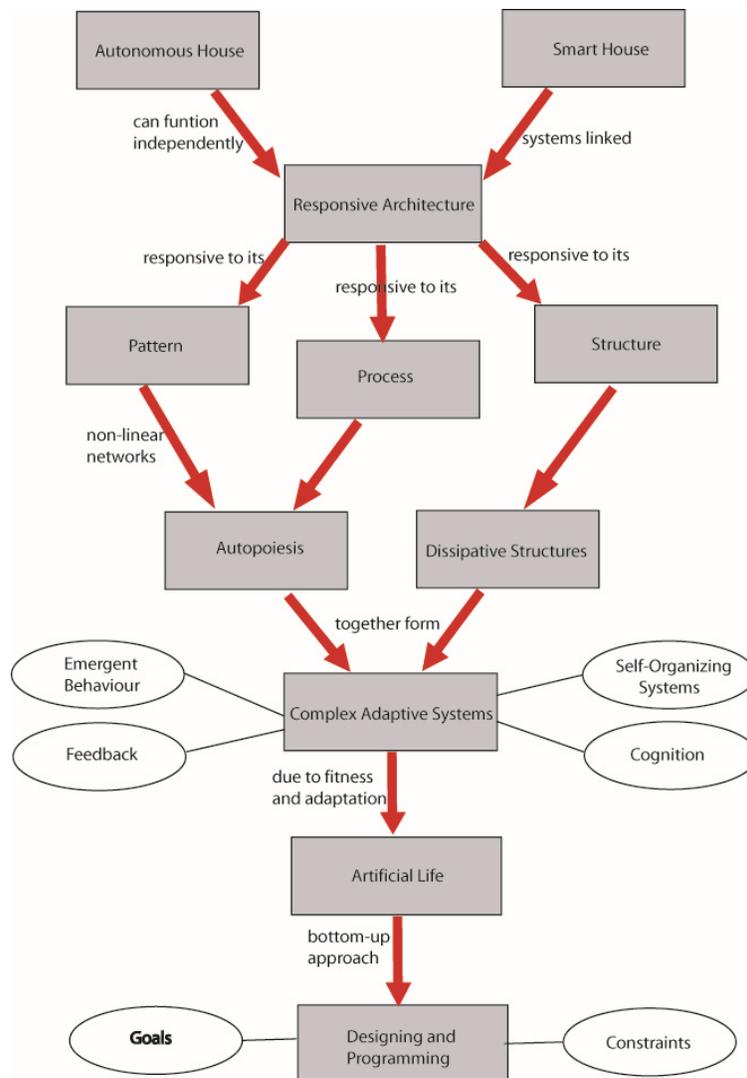


Figure 1: Diagram of Literature Review

2.1. Autonomous Homes

There is a difference to be noted between a smart house and an autonomous house. A smart house has its systems linked electronically but can still depend on the grid whereas an autonomous house is defined as “a house that can function independently of support and services from public facilities” (Vale and Vale, 1975, p. 7). An autonomous house is typically characterized by self-sufficiency by way of energy, food and water, as well as waste treatment.

Another distinction must be made concerning autonomous systems: “An autonomous *system* is

one that senses and responds to its environment. The vast research world of agent-based and multi-agent systems is concerned with such systems. Self-organizing systems are made up of autonomous systems, but not every autonomous system is self-organizing” (Parunak and Brueckner, 2011). Autonomous homes are not self-organizing, they are merely off-the-grid. The prospect of a house becoming truly autonomous in a self-organizing fashion is the purpose of this paper.

2.2. Smart Homes

Home automation has been the subject of early science fiction and has featured popularly in many past World’s Fairs. Remote control devices were introduced in the nineteenth century. For example, Nikola Tesla invented a system of remote control for vessels and vehicles in 1898. There has been a steady increase in various aspects of home automation throughout the twentieth century such as thermostats and laundry machines. Now with the advent of powerful computers, there is nothing that cannot be automated. Home automation is desirable for ease, security, comfort, and energy efficiency. This generally involves automation of systems within the home such as lighting, HVAC, appliances, security, and other systems which are controllable by the inhabitant through a central computer, smart phone or tablet.

More wireless home automation networks (WHANs) are entering the market every year. Some popular examples include ZigBee, Z-Wave, INSTEON, and Wavenis. Samsung is developing a Smart Home system and Apple is testing their HomeKit. Google has also developed two pieces of software for smart homes: Brillo, an operating system, and Weave which is a common language for devices to talk to each other. The list of functions that Weave performs will keep increasing as Google thinks of them, and outside developers are welcome to submit new applications to Google which they will vet. All of these systems are driven by centralized

computing, and as the number of controllable devices rises, interconnection and communication is bound to become more complex.

In terms of security, Parunak and Brueckner (2011) address the difference between open systems and closed systems (of which the closed self-organizing system is explored in this paper):

“Openness has implications for the security of a system, in two opposing directions. On the one hand, the more open a system is, the fewer restrictions are imposed on an element that seeks to participate in it, and the easier it is for malicious elements to insert themselves into the system’s operation. On the other hand, the more decentralized and localized a system’s decisions are, the harder it will be for a malicious element to understand and manipulate the overall state of the system. Roughly, open systems are easier to infiltrate than closed ones, and closed systems tend to limit the extent of damage that can be done. On this subject, engineering of self-organizing systems needs to draw extensively on work on cyber-security and trust.”

Most off-grid homes are small and contain the essentials for generating electricity such as solar panels and/or wind power, and usually employ some rain catchment for purification. Some have composting toilets or grey water systems, but for the most part they are minimal and do not appeal to the majority of the population. Smart homes are by far the more popular choice for home-owners in the near future. But smart homes are just as vulnerable to disruptions from grid failures as regular homes. So there is a trade-off between autonomous homes and smart homes: the autonomous home uses resources in a smart way, but lacks the convenience of smart homes. The obvious answer is to combine the two concepts.

2.3. Responsive Architecture

Responsive Architecture is an area of ongoing experimentation that owes its existence to early thought in the field of cybernetics. In the 1960s, architects like Nicholas Negroponte began to dabble with the possibility of combining technology into the built environment. The

cybernetician Gordon Pask wrote in 1969 that “architecture need not be inflexible, and its goals have less to do with permanence and endurance, but rather a reactive environment in which the inhabitants are in constant dialogue with the mechanisms of design or environmental computing machines” (Pask, 1969). The third wave of computing, or ubiquitous computing as it is often called, has granted the ability to endow any object, or any structure with embedded technology. This can enable information and feedback to be interchanged between a building, the host environment and the users through wireless networks. The integration of computers into everyday objects is leading to transformative advances in the built environment. Everyday objects can now ‘sense’, act, and communicate with other objects through embedded technology, which is becoming increasingly similar to the way living systems are networked. In 1991, Mark Weiser prophesied in *Scientific American* that “where virtual reality puts people inside a computer generated world, ubiquitous computing forces the computer to live out here in the world with people” (Weiser, 1991).

Most examples of responsive buildings are not fully responsive but rather contain some specific responsive system within, or as part of its skin. There are many examples of different types of responsive architectures to be found, but none yet built are truly synthetic organisms of the kind being proposed in this paper. We are at the dawn of the age of synthetic cognition as technology seeks to emulate nature as systems with increasingly complex webs of interactions give rise to emergent properties like those associated with natural living systems.

Rather than static boxes, the architecture of the coming age must be characterized by how the embedded technology within is effectively programmed and implemented, and by the satisfaction of the user experience. The architecture must be of a form that displays a synthetic form of cognition; that learns and adapts to its environment and to the needs and behaviours of

its human inhabitants. This can be accomplished through its patterns of organization. Nature's language is a language of relationships.

2.4. Why a Living Cell?

The cell is the simplest living system, and we want our house to mimic the efficiency of a living system. One could also think of it as an artificial ecosystem. "The *pattern of organization* of a living system is always a network pattern" (Capra, 1996, p. 162). A cell is comprised of a membrane surrounding it, a nucleus within which contains DNA and RNA. Outside the nucleus is a rich soup of chemical elements and factory stations that each contribute to the making of the whole through metabolic processes and catalytic reactions. No one element contained within the cell is itself alive; life emerges from their networked pattern of organization. The *structure* is also an essential factor since it is the physical embodiment of this pattern and is always a dissipative structure. *Process* is a third component which is "the activity involved in the continual embodiment of the system's pattern of organization" (Capra, 1996, p. 161). Process is the link between pattern and structure.

The cell as a system is organizationally closed, but is open to matter, energy, and information. Its own order and behaviour are derived within itself and not from the external environment, which means that living systems are self-organizing. They are autonomous but not isolated from their environment (Capra, 1996). A cell maintains its internal homeostasis even when there are extreme external fluctuations. As mentioned, the parts of a cell are not individually alive, but life emerges through the inter-relational sum of these parts. C. D. Broad coined the term 'emergent properties' in the 1920s for properties that spontaneously emerge at certain levels of complexity. The parts cannot be individually analyzed in reductionist fashion to understand emergence, but rather the underlying pattern of organization of those parts must be understood.

Furthermore, we find systems embedded within systems with different levels of complexity, each producing their own emergent properties at their own level. Organisms are networks of relationships within larger networks of relationships.

In a cell, individuality is maintained within its boundary despite the myriad transformations constantly occurring. It makes itself from within. In terms of a complex adaptive house, self-regulation and organization lead to resilience and robustness. It is self-maintaining by definition.

2.5. Living Systems

There is no definitive definition for Life, but many of Its characteristics have been described in different ways and through key criteria. It is most pragmatic to analyze Life from both a biological perspective and in terms of thermodynamics.

The understanding of life begins with the understanding of pattern; pattern is crucial to self-organization. Systemic properties arise from a configuration of ordered relationships. These components are always arranged in a network pattern. Network patterns are non-linear: messages may travel along cyclical paths and become feedback loops which give the organism the ability of self-regulation. An active network of communication will learn from mistakes as the consequences spread through feedback, allowing for correction, self-regulation and self-organization (Capra, 1996).

Capra (1996) describes the key criteria of a living system:

1. Pattern of organization- *the configuration of relationships* that determines the system's essential characteristics,
2. Structure- the *physical embodiment* of the system's pattern of organization,

3. Life process- the *activity involved* in the continual embodiment of the system's pattern of organization (Capra, 1996 p. 161).

The first and third criteria are described by autopoiesis and the second is described by dissipative structures as follows.

2.6. Autopoiesis

Heraclitus observed long ago that everything in the natural world is in flux. When we suppose we see static situations, we actually see processes of beginning and processes of ending occurring simultaneously.

The concept of '*autopoiesis*,' was coined by the Chilean biologists Francesco Varela and Humberto Maturana in 1972. It means literally 'self-making,' and is defined as: "the pattern underlying the phenomenon of self-organization, or autonomy that is characteristic of all living systems. It is a network of production processes, in which the function of each component is to participate in the production or transformation of other components in the network. In this way the entire network continually 'makes itself.' It is produced by its components and in turn produces those components" (Capra, 1996, p. 97). "In a living system the product of its operation is its own organization" (Maturana, Varela 1980, p. 82).

There is a distinction between first-order and second-order (multicellular) autopoietic systems. The heart is an autopoietic system as it sustains itself by regenerating the components within its boundary. However, it is composed of smaller autopoietic units such as different kinds of cells, and a human in his or her entirety is an autopoietic system. Life is a system of interlocking autopoietic systems (Luisi, Lazcano, and Varela, 1996).

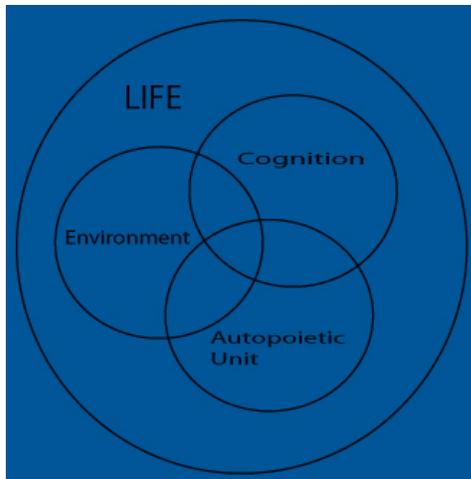


Figure 2: Trilogy of Life

2.7. Dissipative Structures

"Invention, it must be humbly admitted, does not consist in creating out of void, but out of chaos." Mary Shelley

However, autopoiesis does not address the physical constitution of the system's components. To understand the properties of the components and their physical interactions, a description must be given in the language of physics. The term *dissipative structures* was coined in the seventies by Prigogine and Nicolis who have written several books on the subject (Prigogine won a Nobel prize for his efforts in 1977). A dissipative system is "a thermodynamically open system which is operating out of, and often far from thermodynamic equilibrium in an environment with which it exchanges energy, matter and information" (Prigogine and Nicolis, 1977).

Organisms are one of the commonly used examples of dissipative structures. This term explains how organisms or open systems function from a thermodynamic standpoint. Energy must be continuously imported from the environment to counteract entropy. It must maintain an unchanging state even when inputs and outputs are still in operation, much like the human body maintains a temperature of 37 degrees C, despite outside factors. This is also referred to as homeostasis. Organisms appear unchanged over time because there is a constant influx and

exchange of matter, energy and information between the system and its environment (Sundarasaradula, Hasan, 2004).

According to Prigogine (1977), dissipative structures not only maintain themselves in a stable state far from equilibrium, but can also evolve. “When the flow of energy and matter through them increases, they may go through new instabilities and transform themselves into new emergent structures of increased complexity.” (Capra and Luisi, 2014, p.159).

One of Prigogine’s greatest achievements was resolving the paradox of the contradictory theories of evolution between physics and biology. Thermodynamics describes an engine slowly running down while biology describes life unfolding toward increasing order and complexity. Prigogine stated that the second law is not violated because at bifurcation points, order emerges spontaneously while “the total entropy of the system keeps increasing, but this increase in entropy is not a uniform increase in disorder. In the living world, order and disorder are always created simultaneously” (Capra, 2014, p.160).

Prigogine noted that dissipative structures are “islands of order in a sea of disorder, maintaining and even increasing their order at the expense of greater disorder in their environment” (Capra, 2014, p.160). This statement is reminiscent of the impact of human activity on our environment. Organisms consume ordered structures (food) from the environment, metabolize these resources, and dissipate structures of lower order (high entropy waste). So order floats in disorder while the overall entropy in the environment increases in accordance with the second law.

Autopoiesis and dissipative structures are used to describe and characterize complex adaptive systems that exist everywhere in nature. A system is complex if it consists of many interacting

elements, so that the behaviour of the system will be difficult to deduce from the behaviour of the parts. This occurs when there are many parts, and/or when there are many interactions between the parts. Emergent properties “are the novel properties that arise when a higher level of complexity is reached by putting together components of lower complexity. The properties are novel in the sense that they are not present in the parts: they emerge from the specific relationships and interactions among the parts in the organized ensemble” Capra, 2014, p. 154-155). This is the end goal of the Complex Adaptive House.

2.8. Artificial Life (AL)

Whereas biology is the scientific study of Life based on carbon-chain chemistry, AL tries to transcend this notion. It states that Life is a property of the *organization* of matter, not a property of matter. Life on Earth is separated into four levels: the molecular level, the cellular level, the organism level, and the population-ecosystem level. In AL studies, at each level, behaviour of the agents (components) and their interactions are specified, and the larger behaviour emerges spontaneously through the pattern of their interactions.

Artificial Life attempts to understand the essential general properties of living systems by synthesizing life-like behaviour in software, hardware, and bio-chemicals. Conventional methods in artificial intelligence have to struggle to reveal and explain emergence because they are reductionist. That is, they reduce systems to constituent subsystems and then study them in isolation. This is the top-down approach. In contrast, artificial life adopts the bottom-up approach which starts with a collection of entities exhibiting simple and well-understood behaviour and then synthesizes more complex systems (Kim and Cho, 2006). For this reason the bottom-up approach is preferred for the Complex Adaptive House.

2.9. Self-Organizing Systems

Understanding how Life works let alone artificial life, is impossible without understanding the mechanics of self-organization since it is an integral component. The house in question must therefore have this ability. There are a variety of definitions of self-organization depending on the field of study, whether it is provided by cybernetics, information theory, thermodynamics, mathematics, biology, synergetics or others. One definition is provided by Banzhaf (2009) who defines self-organizing systems as referring “to a class of systems that are able to change their internal structure and function in response to external circumstances. By self-organization it is understood that elements of a system are able to manipulate or organize other elements of the same system in a way that stabilizes either structure or function of the whole against external fluctuations.” It is not difficult to see how this description of a system would be advantageous if applied to the functioning’s of a smart house in the face of climate change.

As opposed to the Newtonian paradigm of mechanics in which everything is reducible and deterministic, the science of complexity highlights the fact that nature is creative and full of novelty. Self-organization is “the appearance of structure or pattern without an external agent imposing it” (Heylighen, 2001).

The study of self-organization began in the 1950s in the fields of cybernetics and thermodynamics. The principle was first proposed by the cybernetician W. Ross Ashby (1952), who noted that “a dynamical system, independent of its type or composition, always tends to evolve towards a state of equilibrium, or what would now be called an *attractor*. Self-organizing systems are characteristically robust and resilient. Unlike most human designed systems, they are generally undeterred by environmental perturbations or errors, and have an uncanny ability for self-restoration. For example, an ecosystem will suffer damage from a fire

but can recover fairly quickly due to its redundant, distributed organization: the undamaged parts make up for the damaged ones.

2.9.1. Feedback in Self-Organizing Systems

A system that is in the process of self-organizing begins with a positive feedback phase where some initial fluctuation is amplified and continues to amplify until it has exhausted its resources (like a hurricane). At this point the components will have aligned themselves into a configuration: an equilibrium or stationary state where no further growth is possible. The only changes are those that reduce it. When components begin to deviate from this new configuration, they are brought back in line, or suppressed. The system has now entered the negative feedback phase. Usually there will be many rotating positive and negative feedback loops so that some changes are amplified while others are suppressed.

“Once the system has settled into a negative feedback loop, it will be relatively impervious to external disturbances. The system has now become responsible for its own maintenance, and has thus become largely independent from the environment. It is thus ‘closed’ against influences from the outside. Although in general, there will still be exchange of matter and energy between system and environment, the organization is determined purely internally. Thus the system is thermodynamically open, but organizationally closed” (Heylighen, 2001).

The system may settle into any number of subsystems which will continue to interact, if somewhat indirectly. They become self-sufficient in these closed configurations, and begin

“determining subsystems at a higher hierarchical level, which contain the original subsystems as components. These higher-level systems themselves may interact until they hit on a closed pattern of interactions, thus defining a system of a yet higher order. This explains why complex systems tend to have a hierarchical ‘boxes within boxes’ architecture, where at each level you can distinguish a number of relatively autonomous, closed organizations” (Heylighen, 2001).

Heylighen (2001) then gives an example of a cell that is organizationally closed within a membrane, which contains a vast factory of processes and can remain undisturbed by outside

factors. But cells are organized in circuits that combine to form an organism. The organism is likewise connected by an ecosystem. “Organizational closure turns a collection of interacting elements into an individual, coherent whole. This whole has properties that arise out of its organization, and that cannot be reduced to the properties of its elements. Such properties are called *emergent*” (Heylighen, 2001).

2.9.2. Fitness and Adaptation

The specific configuration of a system can be referred to as ‘fit’ if it can maintain its configuration and grow, allowing it to succeed in its environment. Different configurations have degrees of fitness (likelihood of surviving). Adaptation means achieving fitness between system and environment, and every self-organizing system adapts to its environment (Heylighen, 2001).

In terms of self-organization, three things are necessary for the system to be adaptive:

- “1. An increase in the ability to utilize resources and energy more effectively and efficiently;
2. An increase in the ability to seek, process, and make sense of information;
3. An increase in the organization’s knowledge as a result of learning and relearning from history” (Sundarasaradula, Hasan, 2004, p. 141).

According to cybernetics, adaptation can be modeled as regulation and control. It must minimize any deviations from a certain configuration goal by shutting down any perturbations before they become large and difficult to control (Heylighen, 2001). Therefore, the system must be able to do two things:

- “1. Produce a sufficient variety of actions to cope with each of the possible perturbations (Ashby’s ‘law of variety’);
2. Select the most adequate counteraction for a given perturbation. Mechanical control systems, such as a thermostat or an automatic pilot have both variety and selectivity

built in by the system designer. Self-organizing systems need to autonomously evolve these capabilities” (Heylighen, 2001).

2.10. Complex Adaptive Systems (CAS)

“Change is the only constant.”

Until recently, scientific understanding of nature has consisted mainly of reducing complex phenomena down to their basic components (top-down approach). This has been quite successful in most scientific areas, however this reductionist method has not been successful at addressing the characteristics and behaviour of biological organisms, or Life itself, from chemistry and physics alone. Life cannot be understood by dissecting it down to its component parts, nor can behaviour be predicted or controlled. Life is a gestalt.

The study of Life requires multidisciplinary research, combining fields as diverse as biology, physics, computer science, and economics, and over the last few decades a new field of study called ‘complex adaptive systems’ (CAS) has emerged. These differing fields have observed one common characteristic of complex systems: adaptability.

A greater understanding of complexity in natural systems is being achieved through research into CAS. Complexity is “a phenomenon that emerges from the interaction of multiple, simple, but adaptive agents” (Wildberger, 1997). This understanding is gained through computer simulations and experimentation, which have led to “the design of distributed control systems based on the model of a complex system composed of multiple, autonomous, intelligent agents, competing and cooperating in the context of the whole system’s environment” (Wildberger, 1997).

“The basic approach to analyzing a CAS is to model its components as independent adaptive ‘agents;’ partly cooperating and partly competing with each other in their local operations while

pursuing global goals, set by a minimal supervisory function” (Wildberger, 1997). From CAS research experiments, came the method of letting the agents evolve and adapt to their environment as the environment is modified by external forces, as well as co-evolution with other agents. This research has found that “the agents will self-organize in a way that benefits the CAS even as it benefits them individually” (Wildberger, 1997).

It is important to note that a few agents will not make a sustainable complex dynamic system. There must be a certain amount of agents: a critical threshold for complex global behaviour to occur. The exact number of agents depends on the system (Gatti, et. al, 2008).

In order to design such a complex system with distributed rather than centralized control, the designer must first create a ‘fitness function’ for the agents consisting of goals and constraints. The agents then require “a sufficient variety of resources and strategies from which they can select combinations to use that improve their joint ‘fitness’ for controlling the system even as the uncontrollable parameters affecting the system change” (Wildberger, 1997).

Distributed control at the local level is preferable for complex systems and has many advantages over centralized control, especially when the components are dispersed over a wide area such as in telecommunications, transportation, or computer networks. But even in a small system of sensors and actuators, centralized control would “require all the sensor or instrument data be transmitted to a central location, all decisions made there, and detailed commands sent back to each actuator. In any situation subject to rapid changes, completely centralized control requires high data-rate, two –way communication, which is liable to disruption at the very time when it is most needed” (Wildberger, 1997). Distributed or decentralized control is far more efficient.

As mentioned, “the behaviour of a CAS is always generated by the adaptive interactions of its components. And the hierarchical structure characteristic of CAS is also generated-particular combinations of agents at one level become agents at the next level up” (Holland, 2014, p. 32).

All CAS agents are characterized by three behavioural characteristics:

1. *Performance* (moment-by-moment capabilities)
2. *Credit-assignment* (rating the usefulness of available capabilities)
3. *Rule-discovery* (generating new capabilities)

Performance is typically specified by IF/THEN rules: if this happens, then do that. A rule will send signals that activate other rules such as when a gene in a chromosome causes a protein to form that signals another gene to turn on. Agents contain a list of signals that are in use. “IF (signals x and y are present on the list) THEN (add signal w to the list)” (Holland, 2014, p.26). Signals are transmitted back and forth by rules, and rules link together to form sequences and subroutines (Holland, 2014, p.26), much like in computer programming. This makes it feasible to apply the signal-processing approach to a CAS composed of agents that can be modeled through computer simulation.

In order for an agent to be sensitive to its environment, it has a set of detectors to help it translate what is happening around it into signals for internal processing (sight, smell, touch etc.) such as the activities of other agents. The agent can then modify the environment through a set of effectors (muscles) that translate signals into actions (Holland, 2014, p. 27). At this point it may be useful to point out that agents can correspond to various components of the Complex Adaptive House in terms of sensors and actuators (detectors and effectors).

The exchange of signals between agents is best “represented by a directed network, where nodes represent rules, and there is a connection from node x to node y if rule x sends a signal

satisfying a condition of rule y” (Holland, 2014, p. 28). The performance of the system is measured by the flow of signals over this network. CAS networks are typically tangled webs with feedback loops and recirculation. Agents adapt by altering their signal-processing rules according to network changes, usually from environmental perturbations (Holland, 2014, p. 28).

Credit assignment. As an agent evolves through experience, it may be discovered that some rules are not as useful as others, and some may be altogether obsolete. Adaptation in this case requires the assignment of quantity or strength, to rate the usefulness of rules.

Rule discovery. Agents can regard new rules as hypotheses that require testing. In this case credit assignment is more about conformation. Strong rules are hypotheses that have been confirmed (or partially confirmed) and weak rules are hypotheses that are rejected. The goal is to replace weak rules with strong rules, even if a potential rule is plausibly stronger, it must be tested. This will allow the system to function at a higher level (Holland, 2014).

Generating new plausible hypotheses must be based on previous experience since randomly replacing instructions could be disastrous. This previous experience is referred to as *building blocks*. New combinations are generated by ‘cross-over’ rules, much like a horse breeder might breed a horse with long legs to one with a strong neck so that the offspring has both desirable traits. Strong rules are similarly cross-bred and are then tested by credit-assignment. This Darwinian approach allows weak rules to be replaced by stronger rules due to new combinations of ‘fitter’ building blocks (Holland, 2014).

2.10.1. Designing and Programming a CAS

In order to design and build a complex adaptive system we must first accept that there will be surprises, errors and problems at first which may require some human adaptation and tinkering.

Feedback control requires a repertoire of counteractions that can be implemented (Ashby's law of requisite variety) and that we know which countermeasure to perform in any given circumstance (requisite knowledge). The greater the variety of perturbations from the environment means a greater variety of actions that are required for the system to remain in control. In order for it to react quickly, it should have some expectation of consequences: it needs to anticipate. Expectations are subjective probabilities derived from experience: the more often B appears after A, the stronger the association $A \rightarrow B$. Ordering the possible options in terms of probability decreases the complexity of decision-making (Heylighen, 1994).

To be able to adapt and anticipate, a system should also be robust. If a system is fragile, it will 'break' before it is able to counteract a perturbation. Thus we can say that a system is robust if it continues to function in the face of perturbations. This can be achieved with modularity, degeneracy, distributed robustness, or redundancy. Complex systems will combine adaptation, anticipation, and robustness to cope with their unpredictable environment (Gershenson, 2007).

Degeneracy is the ability of elements that are structurally different to perform the same function. Distributed robustness occurs when the function of the system is performed by different components so that they can compensate effects of perturbations. Redundancy is given when there are several copies of a type of element, so that if one fails, others can take its role (Gershenson, 2007). In the context of the Complex Adaptive House, this means backup sensors (the rule of three).

Different combinations of these approaches will be tried by a self-organizing system in order to maintain its integrity during an onslaught of environmental perturbations. A system can modify itself through an adaptation mechanism in order to 'fit' better in its environment. This

adaptation mechanism is made possible through robustness, which allows the system to withstand environmental perturbations by maintaining function and purpose. Adaptation and robustness are achieved by anticipation, which prepares the system for change before the advent of perturbations. “Using cybernetic terminology, adaptation is analogous to feedback control, anticipation to feedforward control, and robustness to buffering. They should all be taken into account while engineering self-organizing systems” (Gershenson, 2007).

“Using computer experiments on CAS to simulate biological phenomena has been called ‘artificial life.’ The attractiveness of these methods for general purpose modeling, design, and analysis lies in their ability to produce complex emergent phenomena out of a small set of relatively simple rules, constraints, and relationships couched in either quantitative or qualitative terms. Inventing the right set of the local rules to achieve the desired global behavior is not always easy-although it often seems obvious afterward. For instance, flocking behavior requires only two basic rules: 1) stay close to the nearest bird, 2) avoid colliding with another bird or obstacles” (Wildberger, 1997).

In CAS, agents generally operate independently and interact by communicating specific information through messages. All agents have unique ID tags used to address messages to them, internal data storage, ‘ad rules’ or functions that handle the messages and provide each agent’s particular reactive and proactive behaviours.

Agent-based modeling begins with a given number of idealized active objects, each having assigned parameter values. Then there is a description of their starting behaviour in the context of a computerized environment (landscape), which includes the initial organizational pattern

representing the interrelationships among all the agents. These parameters reflect the particular system being modeled. “They may include:

- a) Attributes, capabilities or strategies, usually expressed as references or ‘pointers’ to algorithms,
- b) a range of allowable behaviors such as cooperation, competition, or conflict, and
- c) potential sources of strength and weakness, which may be physical, technological, intellectual, social, or political” (Wildberger, 1997).

2.10.2. Adaptation in Agents

“Adaptation may be defined as the capacity for modification of goal-oriented individual or collective behavior in response to changes in the environment” (Wildberger, 1997). An understanding of the many aspects of adaptation is crucial for the study of CAS. It employs two different types of learning: as an experience-based response, and innovation. Experience-based learning means the ability to sort through patterns in order to separate the optimal solution. Innovation means developing new patterns that were previously unknown.

Another distinction to be made is between individual and population learning. Individual agents need to demonstrate a certain degree of plasticity in order to respond to environmental perturbations as well as to other agents in a manner that strengthens their ability for survival, or meets other goals. To find a strategy that increases its fitness, the agent must gather and store enough information to sufficiently predict and respond to perturbations. “The population then adapts through the diversity of its individuals. This implies that some individuals always survive and their individual actions benefit the population goals, so that the population evolves over many generations, surviving as a recognizable organization” (Wildberger, 1997).

2.10.3. Genetic Algorithms (GA)

There are a few different methods to model adaptive agents in software, but they are mostly different variations on the genetic algorithms originally invented by John Holland as search and classification techniques, whose method is to mimic biological evolution and animal breeding. GA is a technique to model the evolution of complex systems by modeling sensors and actuators as a population of interacting agents that evolve and adapt through competition and cooperation for the purpose of establishing an amalgam of goals defined by the designer called their 'fitness functions' (Wildberger, 1997).

GA belong to a very broad class of systems that are 'adaptive nonlinear networks,' along with artificial neural network paradigms. These systems are comprised of many "units that interact in a nonlinear, competitive manner, and are modified by their own interaction or by external operators so that the overall system adapts to its environment, usually defined in terms of the desired output for any given input" (Wildberger, 1997). Competition and cooperation between the agents benefits the system, which adapts through change and growth.

2.10.4. CAS as Models for Distributed Control

In an engineered CAS that is characterized by distributed control, "each individual agent represents a local controller, comprising one or more sensors and/or actuators, a microprocessor with some memory and two-way communications" (Wildberger, 1997). With the speed and processing power of computers today, an engineered CAS can be designed with "a simple regulator or a PID controller with remotely adjustable bias, set point and/or gain" (Wildberger, 1997).

Agent based modeling (ABM), is derived from the physics of non-equilibrium thermodynamics.

Three claims can be made:

- “1. Constraints can be constructed from entropy-producing processes in the *bootstrapping phase* of self-organizing systems.
- 2. Positive feedback loops are critical in the *structure formation phase*.
- 3. Constraints tend to decay. The continued presence of far-from-equilibrium boundary conditions are required to reinforce constraints in the *maintenance phase*” (Guerin and Kunkle, 2004).

These three points indicate how systems learn the structure of their environment.

“Learning in ABM can occur through:

- (a) changes in agent interaction patterns, ex: edge weighting in neural networks,
- (b) changes in agents’ internal rules, ex: distributed classifier systems, or
- (c) changes to potential information stored in the environment, ex: pheromone trail following in ant foraging systems” (Guerin and Kunkle, 2004).

2.11. Engineering Complexity

In a large sensor network, localization is an important issue that must be addressed. In short, each sensor needs to know where it is located, as well as its location in relation to all the other sensors. This can be accomplished using a location system based on Kohonen’s Self-Organizing Maps (SOMs), which are able to provide intelligence features to sensor nodes. A SOM is “a particular neural network that learns to classify data without any supervision” Paladina, Paone, Iellamo, and Puliafito, (2007). A SOM is attached to each node (which uses very little computing resources and storage space), and the SOM system evaluates all the sensor positions in a distributed fashion. Previous to SOMs, localization was accomplished either through GPS or manual programming (expensive and time-consuming). SOM nodes evaluate each other using Kohonen’s learning algorithm.

Kramer and Magee (2007) give a general proposition for a three-layer system to design adaptive and self-managing systems whereby the components configure their own interactions. The lowest layer is called the *component control*, which is comprised of sensors, actuators and

control loops. The middle layer is called *change management* which is a sequencing function to which the component control reports changes to the state of the system. The middle layer is where new control behaviours are planned, and existing control parameters are adapted. The higher layer is called *goal management* where planning is finally executed according to the data coming from the middle layer, and the high-level goals specified by the designer.

Another way to think of engineered self-organizing systems is given by Parunak and Brueckner (2011) who state that “attempts to gain a formal purchase on self-organizing systems usually involve one or more of three critical dimensions: a vertical dimension (emergence) that relates lower-level and higher-level behaviors, a horizontal dimension (organization) that relates entities at a single level to one another, and a temporal dimension (dynamics) that explores how the system develops through time.”

Gershenson (2007) provides a strategy for the design of self-organizing systems that is composed of five phases: representation, modeling, simulation, application, and evaluation. It is a general methodology and not goal-specific to any particular purpose, but comprehensive nonetheless and useful as a general guideline to would-be designers in any field.

In the *representation phase*, first constraints and requirements are identified. Then the designer chooses a vocabulary for the agents, followed by an abstraction level, granularity, variables, and any necessary interactions. The system is then divided into semi-independent modules with internal goals, and with interactions with the environment while considering different levels of abstractions (Gershenson, 2007).

In the *modeling phase*, the designer must define an internal and distributed control mechanism which must *steer* the system rather than control it. To accomplish this, the designer must find

constraints that will prevent any negative interference between the agents (reduce friction) and promote synergy. This control mechanism must be adaptive and active in search solutions (insert algorithm here). This will act on the system by promoting randomness, noise, and variability. A mediator should also be included to synchronize the agents which will reduce waiting times (Gershenson, 2007).

In the *simulation phase*, different scenarios are tried and mediator strategies are tested.

Simulations proceed progressively from the abstract to the particular, and are continually refined. The *application phase* is used to develop and test models in a real physical system, and the *evaluation phase* is to measure performance and compare it to previous versions (Gershenson, 2007).

For the design of complex systems, Brueckner (2000) also provides this comprehensive list of suggestions:

- “1. Things, not functions: avoid functional decomposition, take real world units instead.
2. Small agents prefer many simple agents to a few complicated ones.
3. Diversity, heterogeneity: create agents with differing capabilities and characteristics.
4. Redundancy: the same capabilities should exist more than once, and there should be more than one way to solve a specific problem.
5. Decentralization: create proactive agents and avoid centralized services.
6. Modularity: it should be possible to compose the system’s functionality stepwise, in layers.
7. Parallelism: solve problems in parallel and allow agents to participate in several coalitions at once.
8. Bottom-up control: local interactions lead to a global result, with no entity executing control from the top.
9. Locality: sensor-motor interaction is local, as well as the interactions between agents.

10. Indirect communication: as far as possible, abstain from direct agent-to-agent communication. Passing messages through a shared environment allows communication to be decoupled in time.
11. Recursion, self-similarity: re-use successful structures and strategies at various levels.
12. Feedback, reinforcement: take into account the result of earlier actions.
13. Randomization: introduce a random factor in agent decisions to avoid negative synchronism (e.g., all agents heading for the shortest queue at the same instant).
14. Evolutionary change: prefer gradual and evolutionary change to abrupt and revolutionary change.
15. Information sharing: inform other agents. Learn as individuals or as a society.
16. Forgetting: outdated information must disappear automatically.
17. Multiple goals: include maintenance goals and achievement goals. Design the system to be able to pursue various goals at once.”

3.0. METHODS

“Machines are defective organisms” Lewis Mumford-Pentagon of Power

The goal is to design a general model and to outline some guidelines for a smart autonomous house that functions as an artificially-living organism in which emergent behaviour can arise by applying the principles of self-organization derived from the observation of natural principles, and computer simulated biomimicry. The application of technological hardware will become the components of the artificial organism. Since conducting computer simulations is currently outside the scope of this thesis, an analysis of existing literature will determine the feasibility or difficulty of its realization.

A virtual house was designed and modeled with 3D CAD consisting of diagrammatic physical systems and an outline of a decentralized wireless sensory network to control them. The network consists of ‘agents’ or components that interact with each other through messaging and

actions in reaction to various perturbations in a similar manner as a self-organizing system or complex adaptive system. Diagrams will provide illustrations of this.

Next, there is an exploration of a series of case studies of similar or related projects which was compared to my design which I have entitled ‘the Complex Adaptive House’. There will be an analysis of the comparative features between each case study and this proposed housing model.

Following that, there will be four general scenarios described: the first three outline some climate change-induced weather events and how the Complex Adaptive House would react, and the fourth scenario gives an example of how the house adapts to its human occupants.

4.0. RESEARCH

4.1 Introduction to the Complex Adaptive House

“You never change things by fighting the existing reality. To change something, build a new model that makes the existing model obsolete.” Buckminster Fuller

Smart homes and their ongoing development rely on a centralized hub. This centralized model lacks the robustness and resilience of a decentralized or distributed model. In the latter, if one component fails, others pick up the slack and compensate, whereas in the centralized model, there is one very large point of failure which, in the age of planned obsolescence, is more than disconcerting. The decentralized model also allows for adaptation in the agents as well as self-organization. In this ‘mesh topology’ at least two or more agents (or nodes) are linked with two or more redundant paths in case the link between one path fails, and so another compensates. This also allows for the shortest path to be selected in all cases leading to faster response times with less energy required. Centralized models do not adapt. They are a ‘master/slave’ configuration that is totalitarian. In a centralized rule-based model, decisions are made in a top-down manner, whereas in the decentralized model, decisions are made bottom-up which mimics

life processes. To be more accurate, decisions will be made from every direction. Centralized systems can be very accurate, however as the number of nodes increases along with the level of complexity, its performance becomes exponentially poorer, whereas the decentralized system thrives in complexity. The question is what is the most optimal model for the Complex Adaptive House?

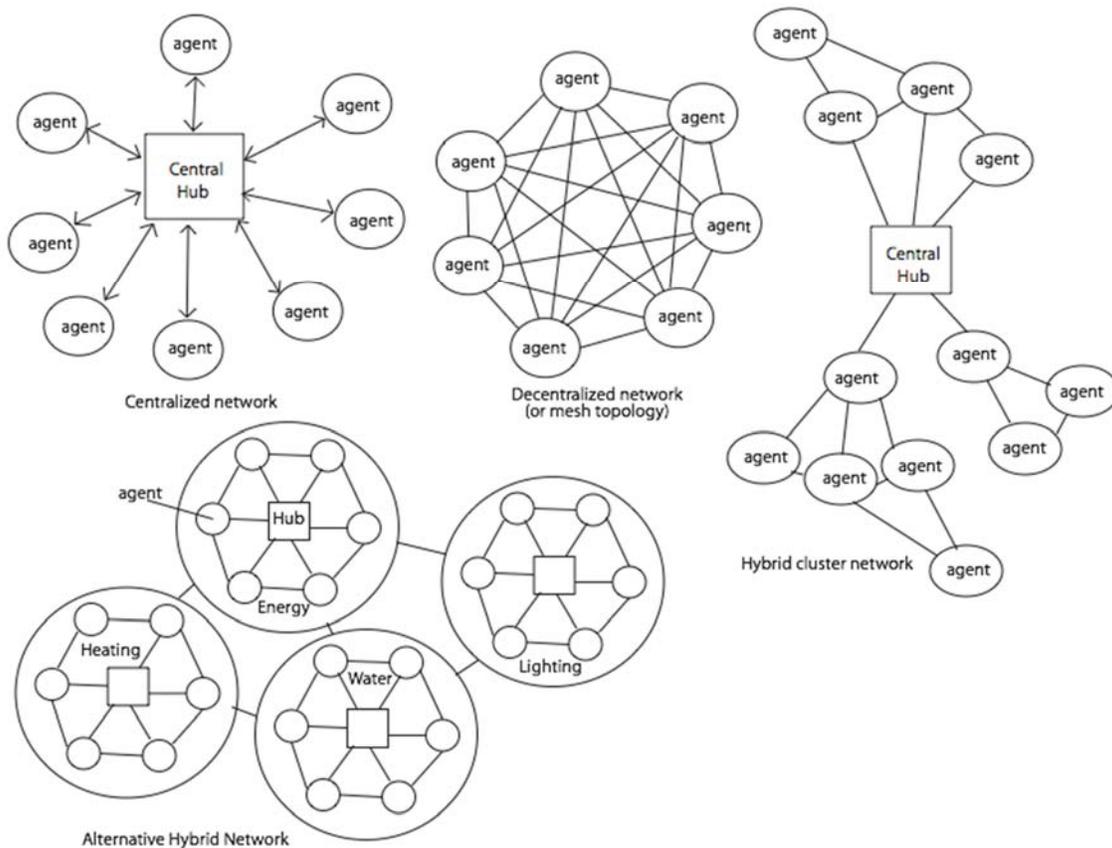


Figure 3: Various Network Typologies

The above diagram illustrates four possible network typologies. The decentralized version shows a mesh topology where everything is coupled to everything else. This has the advantage of allowing the system to figure out its own path workings, so even if the oven and the bathroom faucet never need to have anything to do with each other, the system will find unforeseen solutions by calculating every possible variable. Not every agent needs to be

coupled to every other agent, but collectively they will learn and adapt better through the increased complexity, even though some agents might never communicate with certain agents. Two additional hybrid typologies are provided to show that there are numerous ways of setting up a system. It is mostly dependent on the method of programming.

4.2. Floor plans of the Complex Adaptive House with sensor and actuator legends

Here is a layout for a complex adaptive house. It is to be noted that this is a rather large house meant for a family of three children. A complex adaptive house could just as easily be a two bedroom bungalow. Since large houses generally have very large carbon footprints, a large house was designed for this exercise in order to contrast the fact that aside from the building materials, the Complex Adaptive House has a zero carbon footprint.

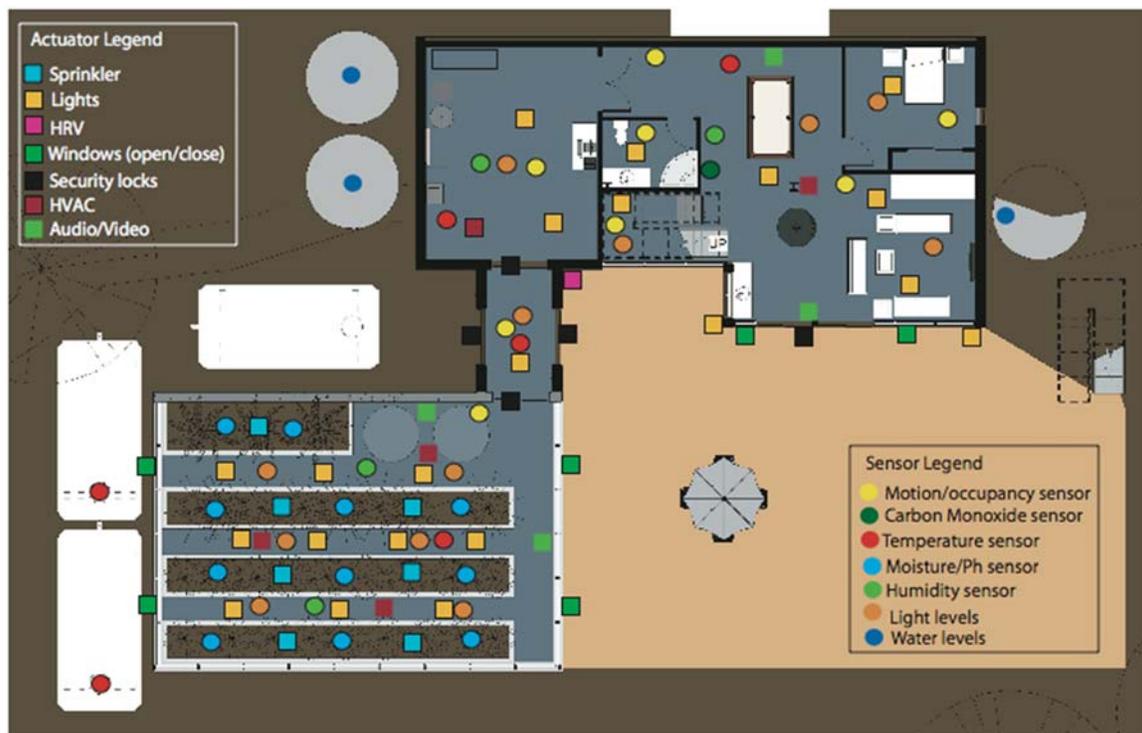


Figure 4: Basement Floor Plan

This house was designed on sloping terrain so that the basement opens to the backyard. The greenhouse is located South/West of the house. Smaller plants are at the front and taller plants

are at the back along with two aquaculture units. To the left of the greenhouse are two biogas digesters and a methane collection tank, as well as two large cisterns to store potable water, with an additional cistern to the right of the house for irrigation. A corridor connects the greenhouse to the house's mechanical room, which contains the computer server, battery bank, hot water tank, heat exchanger, etc. An additional bedroom is provided as well as some open space for a pool table or other activities, an analog fireplace and a TV viewing area. A wind turbine is located to the South/East of the house but is not visible in the plan. The above floor plan contains a rudimentary list of the main sensors and actuators. The mechanical room contains many additional specialized sensors and actuators that are part of the machinery (see appendix).

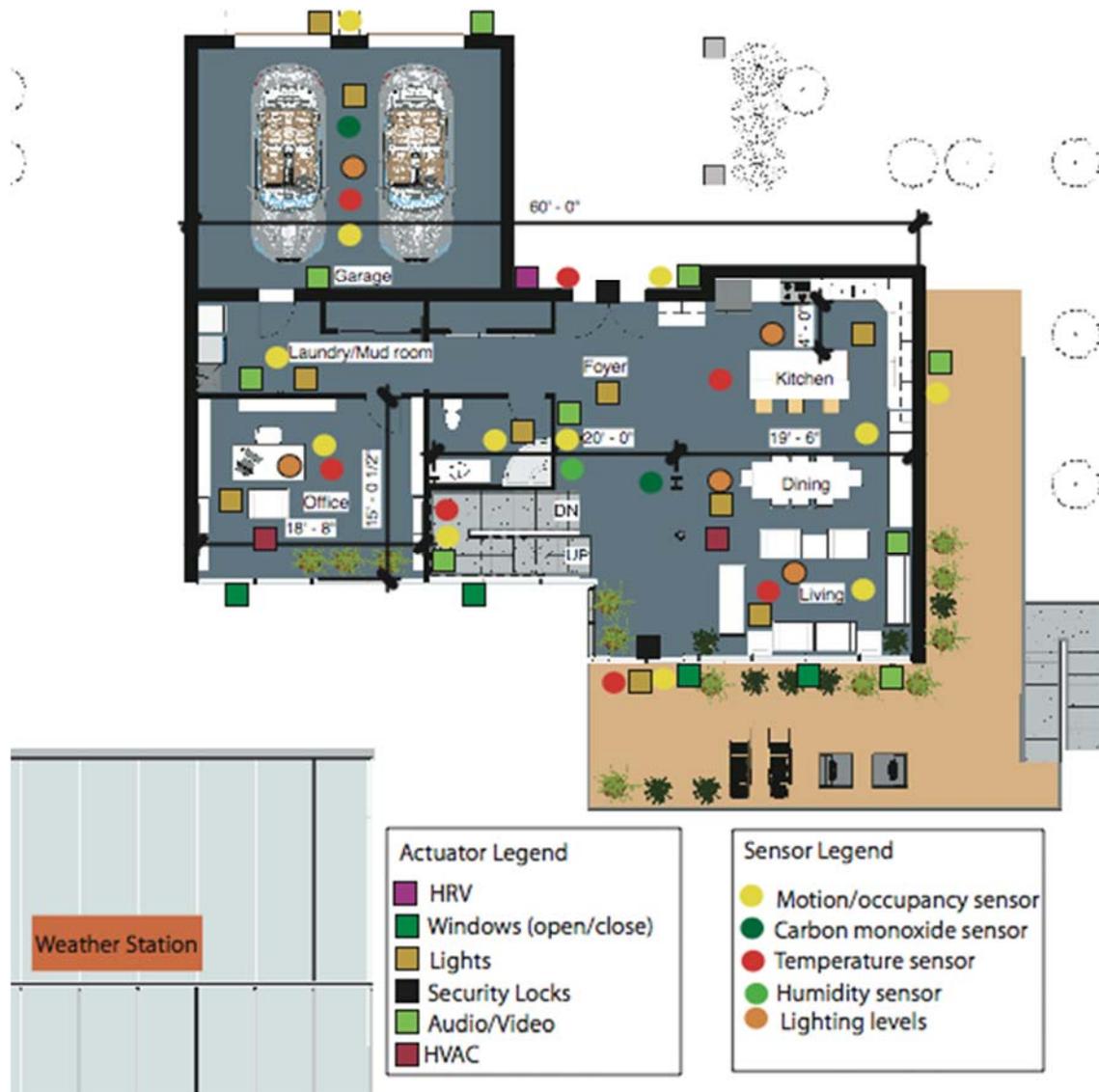


Figure 5: Main Floor Plan

The top of the greenhouse is visible. The right side is an open floor plan overlooking the backyard to the South. The back deck prevents the summer sun from heating the thermal mass in the basement but allows the winter sun to penetrate the glazing beneath.

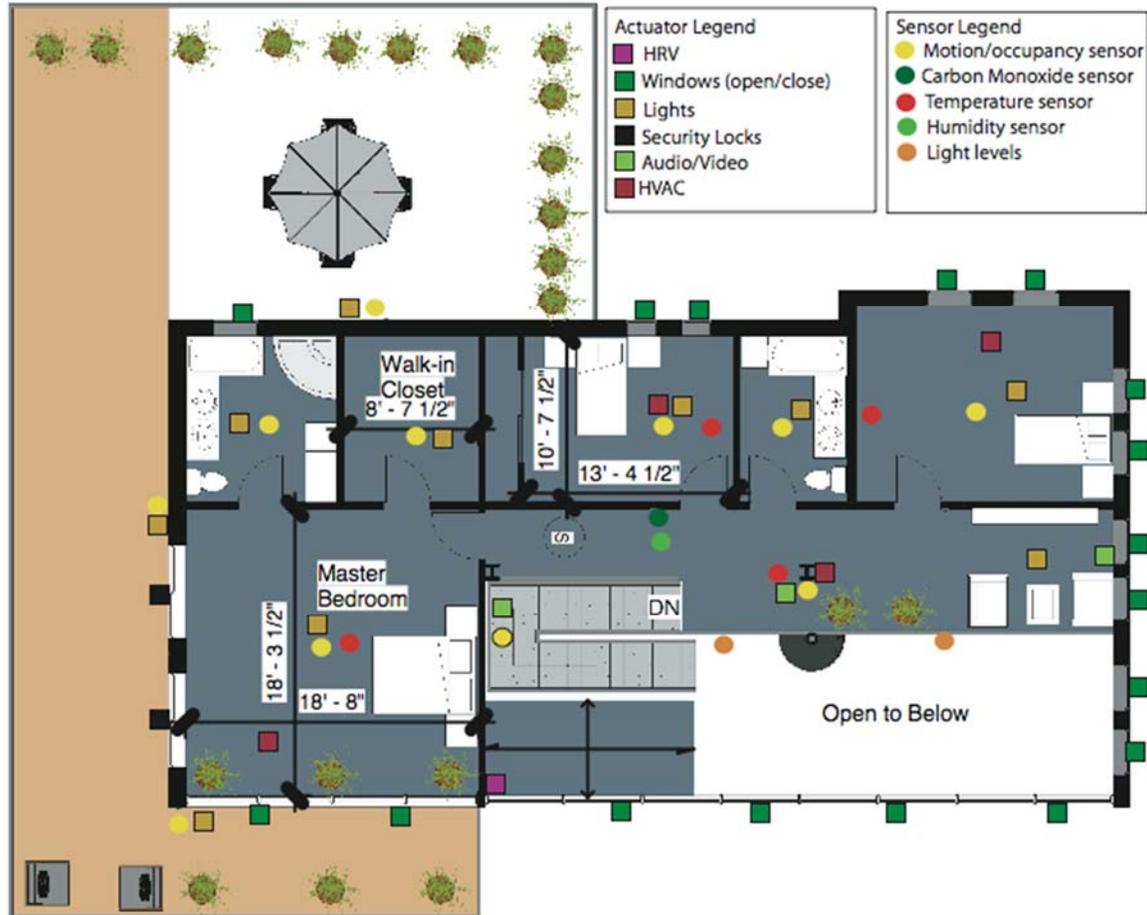


Figure 6: Upper Floor Plan

Generous bedroom sizes, closet space, and stacked plumbing to below are featured on this level, as well as the southern glazing continuing from the main level to the upper level. A deck wraps around the western side and covers the garage below where additional plants can be grown or summer relaxation in the shade. A large sloping roof stretches past the southern glazing to prevent the summer sun from penetrating into the house. The roof also serves as rain catchment, and is covered in photovoltaics with mounted tracking and a solar water tube array.

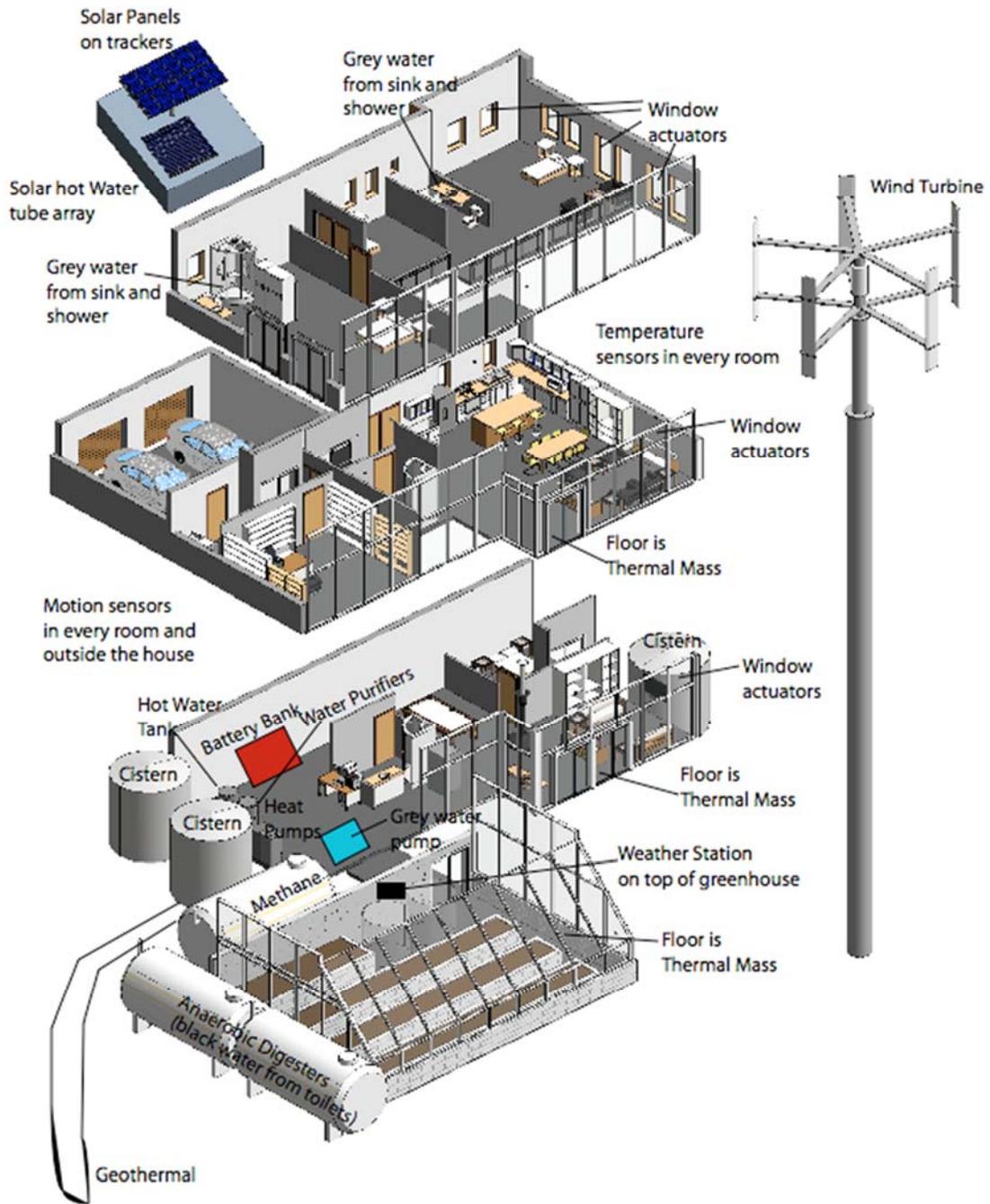
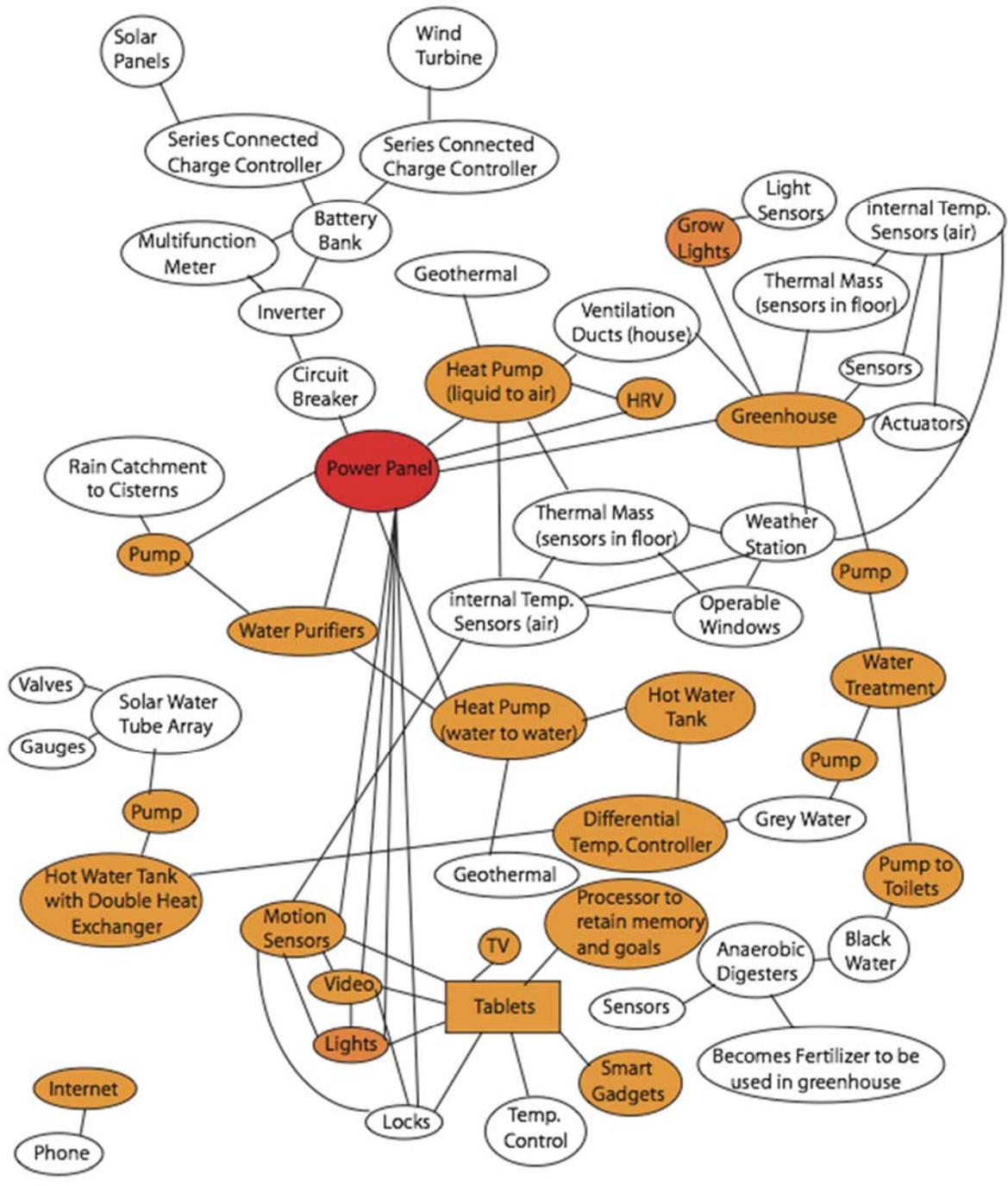


Figure 7: The Complex Adaptive House Exploded



*Orange bubbles indicate drawing electricity from Power Panel

Figure 8: Diagram of What Systems are Coupled to Others

The above diagram is meant to show the likely feedback pathways of the Complex Adaptive House. As it is presented here, it is not a true mesh topology but is rather meant to give a general sense of what agents will communicate with the most. Not every agent or component is featured in this diagram. A true mesh topology diagram of this would look very messy and would be difficult to discern. The bubbles in orange are coloured simply so that lines need not be drawn through other bubbles; making it cleaner. If a diagram were construed combining this concept with all actual floor plans it would look like a chaotic mess, unless it were drawn at a large scale such as a 24 x 36" poster. Not every agent or component is featured in this diagram.

The state of the house can be monitored on any tablet, which can show the state of each sensor, or provide some representation of the fitness landscape as a whole, as well as the ability to enter commands such as adjusting the temperature.

The important point is that there is no central hub to give these commands; the sensors themselves make decisions and communicate to each other by way of microcontrollers embedded within them. A microcontroller is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. The main server is mainly a history database and the keeper of goals and executive decisions given to it by the sensors once an adaptation has occurred. It is not a central controller, merely a repository.

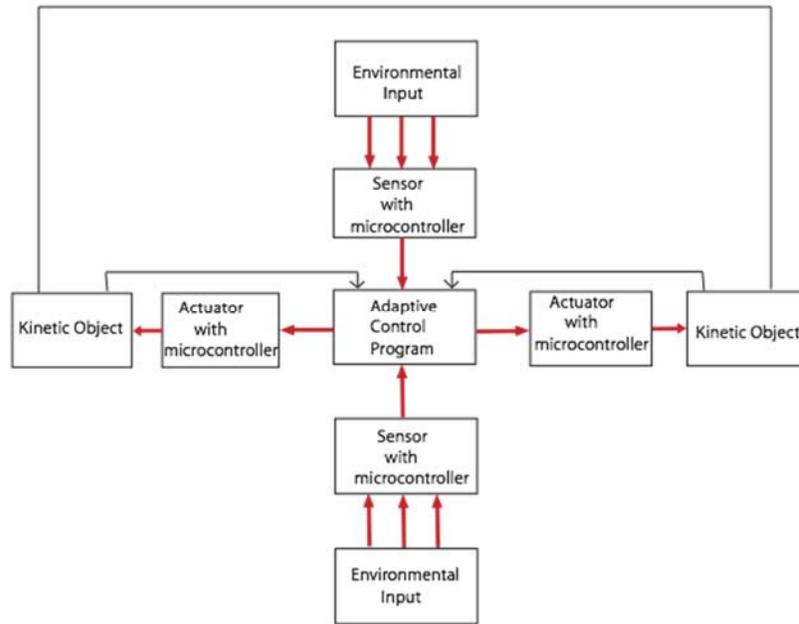


Figure 9: Sensor/Actuator Diagram With Microcontrollers and Adaptation

Adaptive programming can be inserted into each microcontroller in all sensors and actuators along with goals and if/then rules. Once the sensor understands its goals and limitations, it senses environmental cues and sends a signal to an actuator to which it is coupled. A sensor can be coupled to every single actuator or to a single one. In the case of the Complex Adaptive House, each sensor is coupled to all actuators so that the decision making process belongs collectively to the house and not to any particular sensor. All sensors will compete and cooperate for adaptive decision-making.



Figure 10: South Rendering

4.3. Case Studies

"Once something emerges as a successful differentiation, it becomes a new generality from which further differentiations can emerge." Jane Jacobs- The Nature of Economies

Although self-organizing systems and complex adaptive systems are thriving as research fields, the many discoveries have not yet been applied to a smart house. Existing artificial self-organizing systems include the Internet, economies and traffic systems, but since the human brain is the ultimate self-organizing system, robotics and artificial intelligence are benefitting from its study and mimicry. Any system that becomes increasingly complex requires a decentralized or distributed approach since centralization cannot handle complexity.

The following case studies represent various aspects of the Complex Adaptive House, but none of them fully encompass its scope. They range from a successful autonomous housing model to various smart houses, to an artificial organism, a smart village, and decentralized, self-organizing and adaptive sensor networks and software programs that would be pertinent to the Complex Adaptive House.

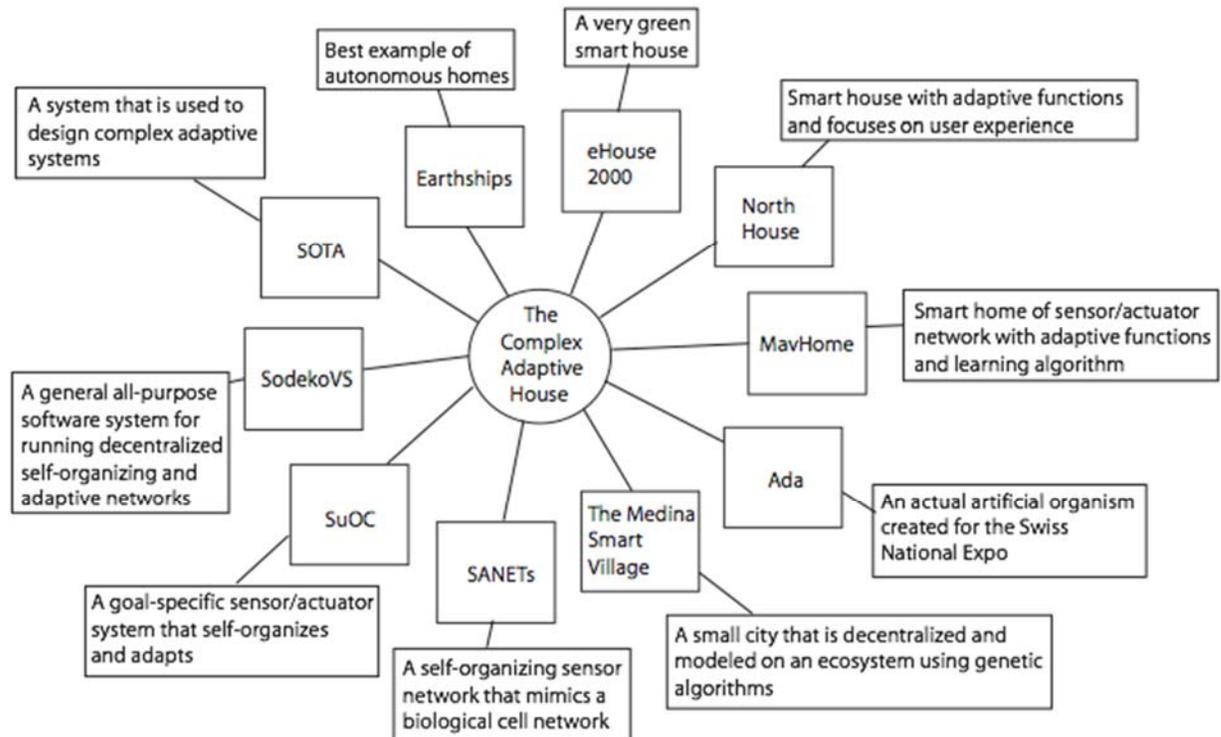


Figure 11: Overview of case studies

Case Study 1: Earthships

Some of the most famous off-grid homes are those designed by the architect Michael Reynolds who refers to his work as ‘Earthship Bioteecture.’ This involves using recycled materials such as tin cans, glass bottles and discarded tires, which are rammed with earth and provide ample insulation. These serve as the bricks, which are mortared together and then plastered over. His first house was built in 1972 and has since perfected his designs through trial and error. In a 2007 documentary, Reynolds describes his latest Earthship: "There's nothing coming into this house, no power lines, no gas lines, no sewage lines coming out, no water lines coming in, no energy being used ... We're sitting on 6,000 gallons of water, growing food, sewage internalized, 70 degrees [or 21 degrees C] year-round ... What these kind of houses are doing is taking every aspect of your life and putting it into your own hands ... A family of four could totally survive here without having to go to the store" (The Garbage Warrior, 2007).



Figure 12: Alberta Earthship

Reynolds experimented with this technique in New Mexico, however these Earthships are gaining popularity in Montana and Alberta, which feature over a dozen of these homes. They are winter-tested and are built with large south-facing windows angled to collect sunlight and heat. They have solar panels and wind turbines to generate electricity and cisterns to collect rain and snow melt that are filtered for drinking water. Earthships tend to include a greenhouse, and increasingly in newer versions, aquaponics. Earthship biotecture kits can be purchased from <http://earthship.org>.

Earthships produce their own electricity with a prepackaged photovoltaic/wind power system. This energy is stored in batteries and supplied to your electrical outlets. Earthships can have multiple sources of power, all automated, including grid-intertie. Photovoltaic panels convert the sun's energy into DC current electricity which is stored in a battery bank. An Earthship Power Organizing Module draws the electricity from the batteries, inverts some of it for AC electricity and supplies it to the home. The POM can accept electricity from a gasoline generator, intertie with the city energy grid, etc.

Water is caught from a roof with a potable surface. From the roof, the water is channeled through silt catches into cisterns. Cisterns are sized to the local climate and are buried and

completely protected from the sun. The water from the cistern is gravity-fed into a Water Organizing Module with a pump and filter. The pump pushes the water into a pressure tank to supply code required water pressure. The filters clean the water for consumption and cleaning.

The hot water recirculation system is a water saving set-up designed so that when a timer switch is turned at a fixture, the water that has cooled in the hot water lines is returned to the cisterns while replacing it with hot water.

These buildings feature a self-contained sewage system. After water is used in sinks or showers, it drains into linear biologically developed interior grey water treatment and containment systems (grey water planters). Clean-looking but not drinkable water is piped to flush the toilets. Flushed water goes outside to a septic tank that is solar heated to treat it anaerobically. Then sometime later, it is directed to an exterior botanical cell. This facilitates total containment of remaining effluent and directs its use toward exterior landscaping. Afterwards, the water again tests below measurable nitrate levels.

The gWOM pumps the treated water, which is now grey water to the toilet. The Earthship grey water system has been researched and developed by Earthship Biotechnology for over 20 years. This system significantly decreases the amount of water needed.

Comparative features:

Reynolds sums up some of his design philosophy: “Single-family cellular homes are built every minute somewhere in the world. Sustainable practices must be implemented on a decentralized individual basis for evolution to happen. If a home is built and something in the technology evolves, then the next home, the next hour will address it. This hourly evolution puts us in a position to survive. Fifty-year evolution increments resulting from centralized mega systems are

sure to be fatal to the human race. The Earthship concepts are singular, cellular, individual and decentralized. They are the future”(<http://earthship.org/the-argument-for-decentralization>).

It is clear that Reynolds has the same basic design philosophy as what is being proposed in this paper. Centralization will be the downfall of our infrastructure, and decentralization will insulate us from extreme weather events as well as human error.

Reynolds’ experience of perfecting his designs is possibly the best example of autonomous homes. These Earthships are perfect for getting off the grid and living comfortably but they are not intended to be smart homes, however they could easily be converted into a Complex Adaptive Earthship with sensors, actuators and artificial life programming.

Case Study 2: eHouse 2000

Architect Michael McDonough designed this experimental house in Stone Ridge, New York as a laboratory for living. It is an internet-based structure that involved a team of engineers, building scientists, computer scientists and environmentalists. Most of the house was built off-site in a controlled environment and assembled on site. McDonough was clear about emphasizing efficiency rather than convenience through technology. eHouse 2000 demonstrates how wireless network technology can be used to save energy in combination with radiant cooling and passive solar heating. It utilizes sensors to monitor temperature, humidity, heat gain and ground freezing near systems that are prone to damage. Sensors also monitor areas that are prone to failure or poor performance such as drains and flashings. The data obtained from these sensors is accessed from a home computer using SCADA software (Supervisory Control And Data Acquisition). The inhabitants are alerted to any problems, and it is designed to automatically make necessary adjustments or allow the inhabitants to make them from the computer. The system is also monitored and controlled from a website.

Other features of eHouse include on-site geothermal energy, off-site wind power (zero-energy home model), photovoltaic uninterruptible power source (UPS) technologies, thermal solar technologies, multi-fuel radiant heating, cooling, and snow melt systems, high efficiency computer monitored domestic hot water, centralized displacement ventilation, a full fresh air dual core ERV-HRV system, passive solar high thermal mass floors and roofs, computer-controlled rainwater retention, a thermal reservoir with water-to-water heat exchanger technologies, selected high performance smart house building controls, demand-response distributed intelligence systems controls, supervisory control and data acquisition (SCADA) controls, web-based systems monitoring and control, CAT 5e backbone ELAN with wireless networking, cross-platform Mac/PC compatible networking and remote access.

The high-efficiency wood-fired Rumford fireplace in the living room, which is very close to the kitchen, doubles as a bake oven. There is plenty of day-lighting and with operable windows and natural ventilation, but most impressively, there is a vegetable garden on the roof.



Figure 13: eHouse Renderings, Eight Different Angles

Comparative features:

eHouse is similar to the Complex Adaptive House on many fronts such as multiple energy harvesting techniques (solar, wind, and geothermal). It is made from either healthy or locally sourced materials. It makes use of sensors to automate many things inside and out. However, it is controlled from a website through satellite internet access, which is fine if one wants to turn on the lights while in Timbuktu, but unsettling if the internet becomes 'glitchy'. There is still a single point of failure, luckily many features of the house are analog, including the windows.

It is unclear if eHouse is completely off the grid; the issue of waste is not identified and although there is mention of a cistern and a snow-melt system, it is unclear if this is to supplement water from the local county or if it the only source.

By all accounts McDonough's project is a success, but the Complex Adaptive House (CAH) seeks to take the concept a step further. The CAH is a closed system in the sense that it is not accessible by the internet for security reasons. eHouse is a centrally controlled system where the designer picked and chose which systems to automate in true smart house fashion, whereas the CAH's systems are coupled feedback loops where each system feeds into another in one big loop. In nature, waste equals food which the CAH mimics. Also food is grown year round whereas eHouse is limited to growing only in summer. Still, eHouse might be the closest precursor to the Complex Adaptive House.

Case Study 3: North House

A design team at Simon Frazer University have been working on smart house systems and are underwhelmed with the current standard. They argue that current building automation focuses too much on saving energy and water, and far too often neglect the user experience and human

comfort. They propose a system that is comprised of a combination of information and interaction design, automation and control, adaptive intelligent agents, and distributed ambient sensing. Makonin, Bartram and Popowich (2013) point out that inhabitants of smart homes can often find the control systems cumbersome and difficult to manage, especially when the automation moves from user control to some form of rule-based behaviour because the rules might not accurately reflect what the inhabitants actually want. Most users of smart homes want their systems to help with conservation actions such as reducing heat, turning off lights, appliances and devices, but that subsystems are not properly integrated into the larger whole.

While automation in its current form is well suited to reducing energy use, computational intelligence promises simple, adaptive controls that can minimize user effort and optimize energy use without causing discomfort and incurring override reactions which are problematic in overly intrusive systems. “Automated home systems provide two types of control: user control (an aggregation of lower-level controls into more zone-specific groups) and rule-based control, where the system makes decisions, usually based on a schedule. These types of controls are difficult to manage because they force the static definition of a complex, a priori configuration that lacks a dynamic and holistic view of the home” (Makonin et. al., 2013).

People overwhelmingly see rule-based systems as problematic and prone to errors because few people configure the thermostat interface properly because it is invariably complex. More importantly, peoples’ lives are far more complex and variable than the simple rules accept. People end up setting simple schedules and house systems end up running needlessly while they are away.

The authors “advocate for technology that helps humans behave appropriately rather than relieving them of any operational involvement with their homes. Issues of trust and

customization are important. The advantages of smart homes to date have tended to be outweighed by complexity, but it is clear that context-aware systems, distributed smart sensor or agent networks, and adaptive behaviour hold substantial promise” (Makonin et. al., 2013).

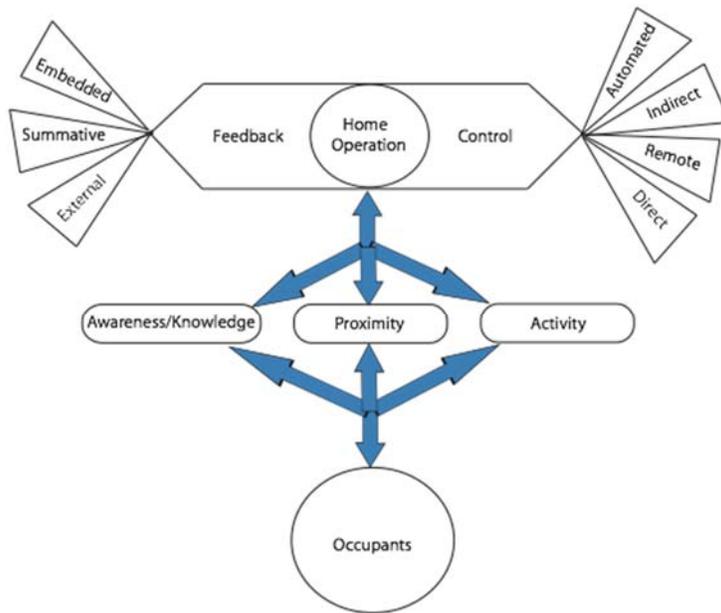


Figure 14: A Depiction of Human-Home Interaction. *How humans interact with the home is central to any automation strategy and depends on awareness or knowledge, proximity, and activity.* (Courtesy of Makonin et. al., 2013)

The smart home is a complex ecosystem involving the occupants, physical and operational components, external and internal context, and dependencies between them. The operational components are the embedded technologies which can include monitoring displays, thermostats, windows, doors, appliances, fans and any other device that allows some kind of feedback control.

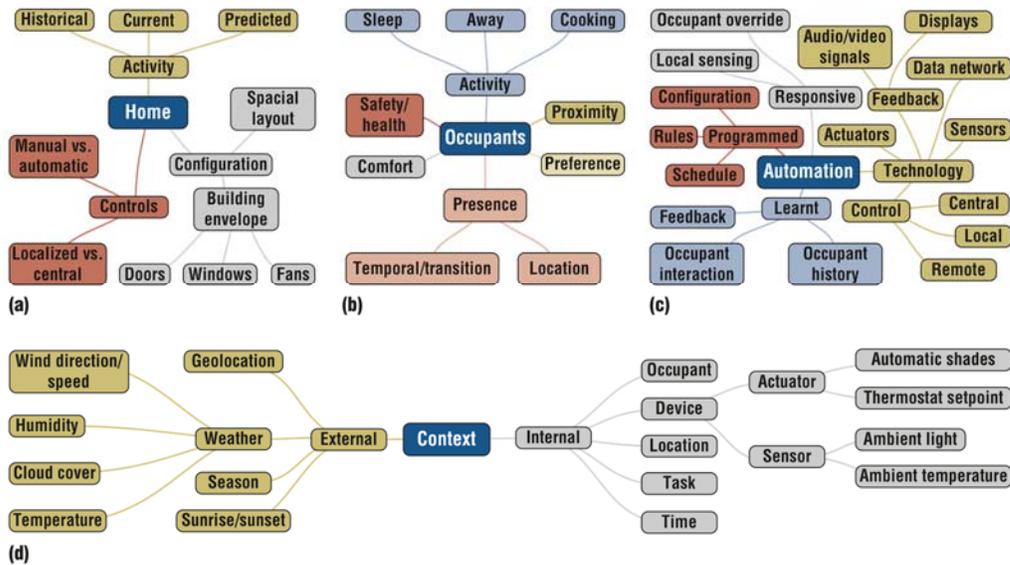


Figure 15: Concept Maps of the Different Factors Relevant for a Smart Intervention Framework: (a) the home, (b) occupants, (c) automation, and (d) context. (Courtesy of Makonin et. al., 2013)

A smart home is a complex ecosystem with many attributes whose state is dependent on its occupants, activities and the configuration of its components (pattern of organization). Figure 22 shows some of these dependencies. Occupant use of the home is difficult to capture computationally; activity, presence, proximity, health and comfort are critical to designing intervention strategies. The authors state that different classes of automation should exist ranging from learned to programmed, and from fully automated to human initiated actions. Makonin et. al., (2013) present two case studies based on their experience designing smart home systems, specifically as the interactive design team. They were invited to tackle the system for ‘North house’, a small solar-powered home designed to achieve net-zero performance in Canada. “It incorporates sophisticated custom energy systems, adaptive intelligent building envelope technologies, specialized lighting and climate systems, and automated optimization behavior” (Makonin et. al., 2013). The control system consisted of several subsystems with intelligent behaviour but the team identified several problems which

they sought to address. “The North House engineers created a technological complex system that was smart in terms of optimal energy efficiency, but the house didn’t balance occupancy comfort with optimal energy efficiency. In fact, it didn’t even know that the occupants existed” (Makonin et. al., 2013). It seemed that overriding the system was a considerable inconvenience. Both of the following case studies show that rigid rule-based programming should be replaced with a dynamic system that can adapt.

The first case study explores adaptive approaches to night lighting. The aim was to avoid disturbing the occupants’ circadian rhythms if someone gets out of bed at night by providing an adequate amount of light to navigate. Using the same light level during the day would be inappropriate. The posed the question of when to prompt an occupant for an action and when to initiate it automatically. The system must be able to identify when people are sleeping. The sensors in the room must coordinate action and feedback to raise the ambient light level to a degree that would not disrupt the persons’ circadian sleep rhythms. The following figure demonstrates a possible implementation.

```
HOME_STATE (variables):
home_sleeping = yes;
bedroom_light_level = 0;
bedroom_light_switch = off;

RULES:
WHEN bedroom_light_switch IS pressed_once AND
WHEN bedroom_light_switch WAS off AND
WHEN home_sleeping IS yes
SET bedroom_light_level = CIRCADIAN_LIGHT_LEVEL;
WHEN bedroom_light_switch IS pressed_twice
SET bedroom_light_level = MAX_LIGHT_LEVEL;
WHEN bedroom_light_switch IS pressed_once AND
WHEN home_sleeping IS no
SET bedroom_light_level = MAX_LIGHT_LEVEL;
WHEN bedroom_light_switch IS pressed_once AND
WHEN bedroom_light_switch WAS on
SET bedroom_light_level = MIN_LIGHT_LEVEL;
```

Figure 16: Circadian-Rhythm-Aware Lighting at Night. (Courtesy of Makonin et. al., 2013)

The rules in Figure 16 are triggered when an event happens which is determined by a learning

algorithm. “The first rule will set the ambient light in a room to a lighting level that won’t disturb the occupant’s circadian sleep rhythms if the light switch is pressed once when it’s off and the house is in *sleeping mode*. The second rule overrides the first if the occupant presses the light switch twice. In case the occupant wants to wake up and fully turn on the lights, the third rule says that if the light switch is pressed once when it was off and the house isn’t in *sleeping mode*, then we should have full lighting. The fourth rule handles turning off the lights in a room” (Makonin et. al., 2013). This provides a balance between occupant comfort and technology.

The second case study involves adaptive HVAC acclimatization. The authors state that monitoring home occupancy using power monitoring and ambient light sensors is necessary to achieve an adaptive HVAC system. They have also modified it to use the arming and disarming of the alarm to determine occupancy. When nobody is home, the system can be put into power saving mode by changing the thermostat set-point and turning off various equipment by sending specific commands around the house. When people return, the power saving mode is reversed.

The following figure shows how that might look.

```
HOME_STATE (variables):
home_occupied = yes;
alarm_system = disarmed;
temp_setpoint = 21.0;
TEMP_INCREMENT = VACANT_TEMP_SETPOINT / 2;
RULES:
WHEN alarm_system IS armed_away
SET home_occupied = no AND
SET temp_setpoint = VACANT_TEMP_SETPOINT AND
SET ambient_displays.send_cmd = 'turn off' AND
SET automation.send_cmd = 'set away profile';
WHEN alarm_system IS armed_away AND
WHEN home_occupied 95% yes WITHIN_TIME - 30 MIN
SET temp_setpoint += TEMP_INCREMENT
WHEN alarm_system IS disarmed
SET home_occupied = yes AND
SET temp_setpoint += TEMP_INCREMENT AND
SET ambient_displays.send_cmd = 'turn on' AND
SET automation.send_cmd = 'set athome profile';
```

Figure 17: Electricity and Heating Cost-Savings With Occupancy Detection

This example does not address every situation in which someone might want to override the temperature set-point, but rather it is meant to highlight some key points that show how rule-based systems with adaptive learning can be used. The first rule says when the system is in away mode, do the following actions: change the thermostat set-point to save energy and turn off all ambient displays. The authors state that many other rules could have been added but that would complicate this example. The next rule states that when the house is unoccupied and there is a 95% chance the occupants will be back within 30 minutes, then increase the heating or cooling. This inference is derived from a learning algorithm that makes rules more adaptable. The third rule states that the system is disarmed when the house is occupied. When the occupant is home the thermostat set-point is changed to its original value. If the adaptive intervention is wrong, it would mean that the occupant returned early or late. If early, the occupant's comfort would not be optimal, but the system would still be optimal in terms of energy efficiency, and the occupant still does not have the inconvenience of setting the thermostat upon their return. If they return late, the occupant's comfort level would be optimal but not so in terms of energy efficiency. Either way there is a slight inconvenience, and either way they do not have to set the thermostat. So having dynamic rules that adapt to occupant behaviour strikes a balance between occupant comfort and energy efficiency. Adaptability can also mean that the system might notice that the occupant returns late every Friday, and adjusts to this behaviour.

Understanding how people use energy in the home is a new area of design research that requires a user-centered approach. A major problem with smart homes in general is that the users are overwhelmed by intrusive and inappropriate automated operations. It is infeasible to design a system that fully encapsulates the complexities of daily life, but adaptation can certainly steer

the system in the right direction. Also, the occupant must be able to easily override any inappropriate intervention by the system. Part of this research is identifying the sub-problems that occur as a result of human behaviour through scenarios and contexts. The authors state that “distributing rules with learning that solve sub-problems should reside in sensors and other equipment around the home. This distribution of intelligence is needed to reduce system latency and create modularity” (Makonin et. al., 2013). They further state that “reducing system latency increases system responsiveness to human-home interactions—which is important when responsiveness must be immediate, as in the case of adaptive lighting for circadian rhythms (responsiveness to light-switch presses must be in the microseconds). Addressing sub-problems in a modularized fashion creates a type of plug-and-play system. This benefits the smart home by allowing functionality to be extensible, malleable, and upgradeable” (Makonin et. al., 2013).

Comparative features:

The authors continue to test smart homes and insist that they must be deployed and studied in situ, not theoretically in a simulation. This study provides a window into the advantages of employing adaptive programming into a distributed wireless network rather than the standard smart house model, which can frustrate the occupants from a lack of user-centered experience. When programming the Complex Adaptive House, it is important to take a user-centric approach as well as the consideration of energy efficiency. This study shows that the two are intertwined. Though this is a relatively simple project without the amount of complexity being proposed by the Complex Adaptive House, North House addresses some key issues that standard smart houses do not.

Case Study 4: MavHome

“Environments that sense their own well-being and can request repair or notify

inhabitants of emergencies can save property and lives. Homes that can increase their own self-sufficiency over time can augment busy or aging inhabitants allowing people to live in their homes longer (potentially alleviating some health care system burdens) and free time to allow people to focus on other aspects of their lives” (Cook, Youngblood and Das, 2006).

The MavHome smart home (Managing and Adaptive Versatile Home) is a project being conducted at the University of Texas. The researchers there are intent on designing a “versatile and intelligent home environment that is able to acquire and apply knowledge about its inhabitants and their surroundings in order to adapt to the inhabitants and meet the goals of comfort and efficiency” (Cook, et. al., 2006). It is composed of agents (sensors and actuators) which allow the home to adaptively control climate, water, lighting, maintenance, and multimedia entertainment. The authors stress that, as opposed to other smart environment projects being conducted by various companies, this one uses “multiple heterogeneous learning algorithms in order to identify repeatable behaviours, predict inhabitant activity, and learn a control strategy for a large complex environment” (Cook et. al, 2006).

The MavHome uses predictive algorithms to acquire and apply information about the occupants to meet the goals of comfort and efficiency. It has the ability to decide whether or not to automate some specific activity as well as to find ways of improving the activity in order to meet system goals. It tries to predict an occupants’ next action concerning selected repetitive tasks based on previous interactions with various devices, while keeping prediction errors to a minimum.

MavHome uses the *Active-LeZi* algorithm to meet its prediction requirements. The researchers have claimed 100% predictive accuracy on a 30-day trial for a variety of scenarios. When predictive accuracy drops below a predetermined level, “the predictor is retrained over data from a sliding window to adapt for concept drift in the changing patterns of inhabitants” (Cook

et. al, 2006).

The MavHome architecture is a hierarchy of rational agents which cooperate to meet an overall goal. Each agent is defined by four layers of cooperation: “The **Decision** layer selects actions for the agent to execute based on information supplied from other layers. The **Information** layer gathers, stores, and generates knowledge useful for decision making. The **Communication** layer includes software to format and route information between agents, between users and the house, and between the house and external resources. The **Physical** layer contains the basic hardware within the house including individual devices, transducers, and network hardware. Because the architecture is hierarchical, the physical layer may actually represent another agent in the hierarchy” (Cook et. al, 2006).

Sensors send signals to other agents through the Communication layer. The database stores the information while it is being processed into useful knowledge (patterns and predictions). Information is then given to the Decision layer which then selects an action, and relates it to the Information layer. Upon updating the database, the Communication layer transmits the action to an actuator in order to effect some change. A specific interface agent provides interaction with users and external resources like the Internet. Agents can communicate with parent/child agents or other agents at the same level in the hierarchy.

Figure 18 shows the set of concrete functional layers which serve to realize the set of abstract layers previously mentioned. The **Physical Components layer** is composed of all devices utilized such as powerline control interface hardware, touch screens, gesture input devices, cameras etc, with the exception of the main computer itself, which, along with the sensor/actuator network, is considered the host of all layers. The **Computer Interface layer**

consists of the hardware interfaces to the devices such as USB or PCI card interfaces, device drivers, the operating system, and all software interfaces. “In the **Logical Interface layer**, the hardware device services and APIs are utilized to create simple, light-weight programs that create a series of atomic services around each sensor and actuator in the system. These logical proxies provide information and control via socket and shared memory-based interfaces in a modular design. All of the lower layers are based on simple single application components, but in higher layers the components become more complex” (Cook et. al, 2006). The **Middleware layer** services the upper layers to facilitate communication, process mobility, and service discovery using point-to-point and publish-subscribe forms, naming/service discovery, and a mechanism to move system components between computing hardware devices. “The **Services layer** utilizes the Middleware layer to gather information from lower layers and provide information to system applications above. Services either store information, generate knowledge, aggregate lower level components, or provide some value-added non-decision making computational function or feature (ex. user interfaces” (Cook et. al, 2006). Finally in the **Applications layer**, all learning and decision-making components conduct their operations.

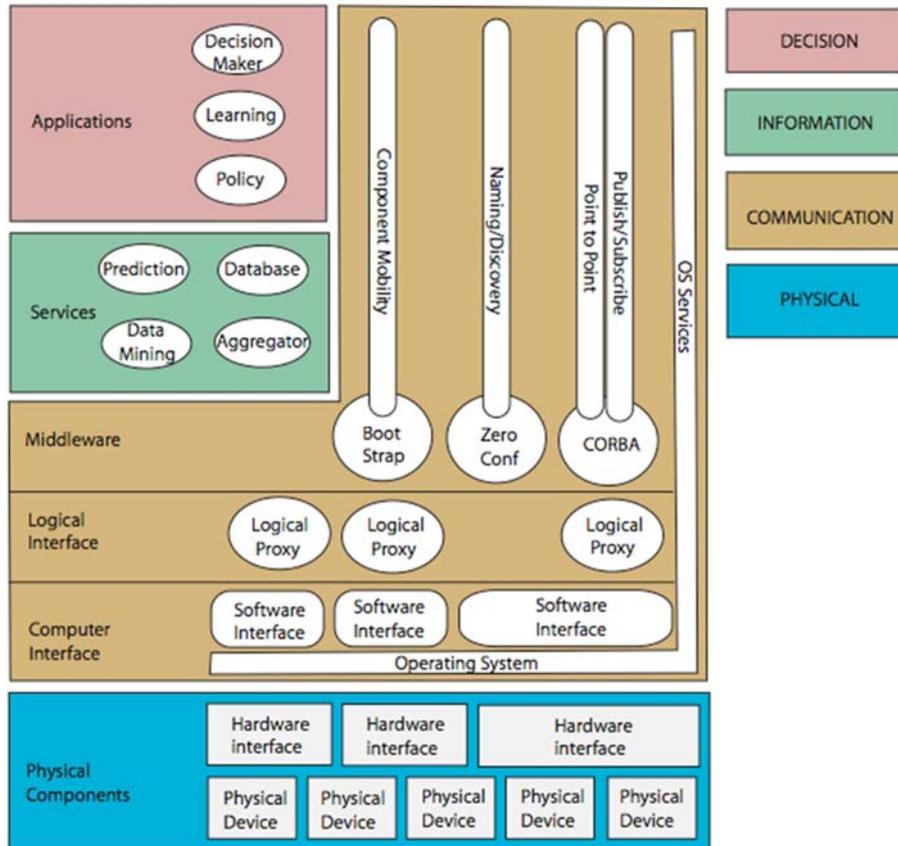


Figure 18: MavHome Abstract Architecture

As mentioned, this system used a variety of technologies and algorithms for specific purposes. MavHome uses the Common Object Request Broker Architecture (CORBA) to communicate between agents because the authors state that it has a clear and simple interface design, ease of integration, and object-oriented design. The ZeroConf technology allows new agents to be introduced to the system without having to redesign everything. The algorithms used for discovery, prediction, and decision-making are designed to minimize resource usage, maximize comfort, and adapt to the inhabitants.

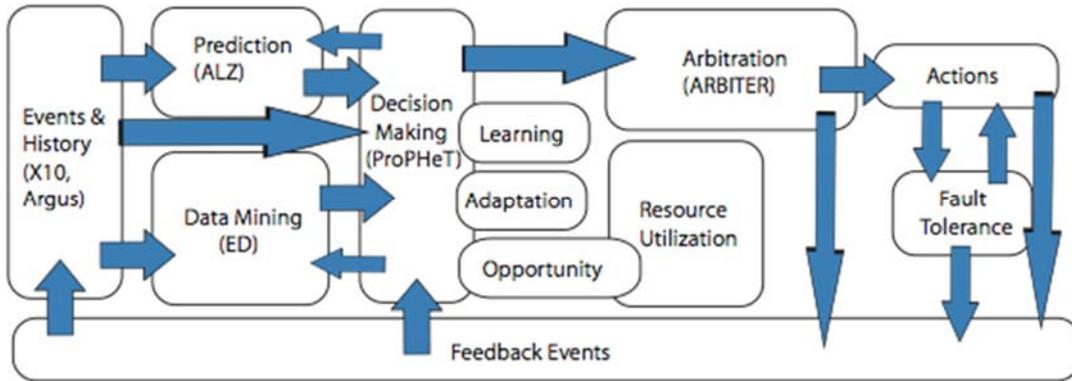


Figure 19: MavHome System Architecture



Figure 20: (a) MavHome Argus Sensor Network, and (b) MavHome Apartment Without Sensors.

Comparative features:

This project bares some similarities to the Complex Adaptive House, except that it utilizes AI (artificial intelligence) programming rather than AL (artificial life) programming, and is centralized rather than decentralized. Still it is useful to describe the manner in which it functions in terms of the wide variety of sensors and actuators deployed, and to understand what algorithms are used. The MavHome is an interesting case study in terms of adapting to human inhabitants and the automation of many basic needs. However it is not equipped to handle the level of complexity generated by the Complex Adaptive House. But it is projects such as this that make the realization of the Complex Adaptive House completely feasible.

Case Study 5: Ada

An interesting example of an interactive and adaptive system is Ada, which is a room that was constructed for the Swiss national exhibition Expo in 2002. Ada was “conceived as an artificial organism that can interact and communicate with her visitors” (Eng, Babler, 2001). The inventors stated that “buildings can be conceived and constructed as perceiving, acting and adapting entities not unlike biological systems” (Eng, Babler, 2001). Ada was equipped with vision, hearing and touch in the form of various sensors including a matrix of cameras and microphones as well as pressure-sensitive floor tiles. The inventors state clearly that on top of those basic physical attributes, Ada displays four main behavioural functions. She tracks visitors, individually or in groups, identifies those who are more ‘interesting’ because of their responsiveness to simple cues that Ada provides and uses to probe their reactions. Ada then ushers them into groups in order to play a number of group-based games while evaluating the results, expressing emotional states, and regulating the distribution and flow of her visitors. The inventors stress that this “scenario is not a static progression, but a set of interconnected,

interdependent, simultaneously evolving internal processes” (Eng and Babler, 2001).

“Like an organism, Ada’s output is designed to have a certain level of coherence and convey an impression of a basic unitary sentience to her visitors” (Eng, Babler, 2003). The inventors say that Ada was “used to gain practical experience in handling large-scale behavioural integration issues in autonomous systems” (Eng, Babler, 2003). Over the course of five months, Ada received roughly 550,000 visitors. Ada is equipped with 15 video inputs, 367 x 3 tactile inputs, 9 audio input channels, 46 mechanical degrees of freedom, 17 output audio channels, 367 x 3 floor tile lights, 30 ambient lights and 20 full-screen video outputs. “All of these inputs and outputs can be addressed independently, giving a rich array of sensory modalities and output possibilities” (Eng, Babler, 2003). Ada has a tracking system comprised of floor tile pressure sensors which determine the location, speed, direction and weight of her visitors. In addition, a vision system collects images of people, and an audio system which localizes people and recognizes sounds such as the word ‘Ada’ with pitch, note and key to distinguish between individuals.

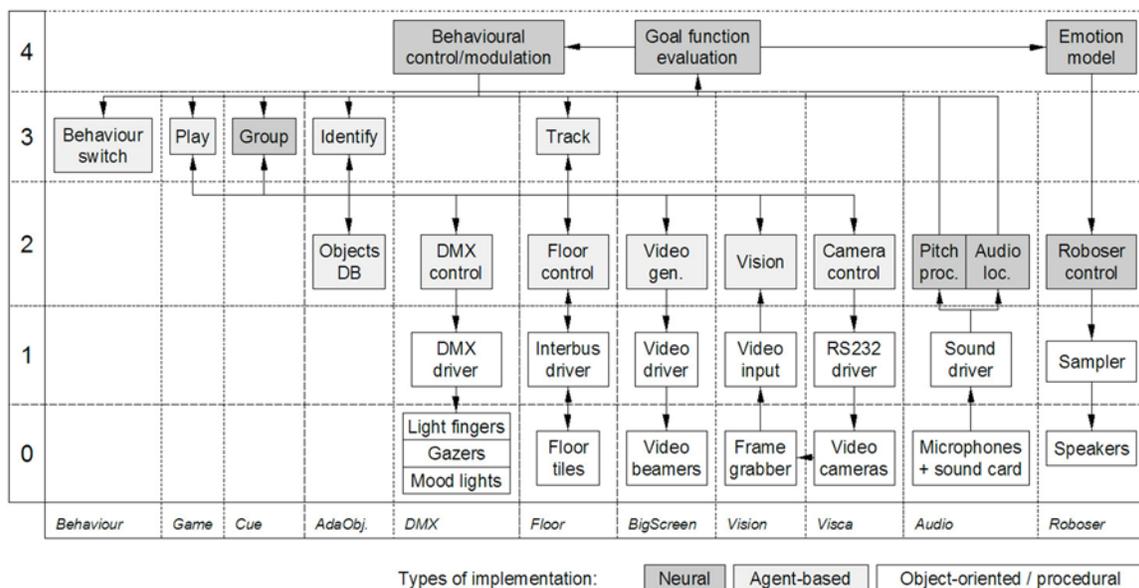


Figure 21: Overview of Ada System Architecture Organized Into Conceptual Layers

The inventors stress that Ada needed to convince her visitors that she is an artificial organism. She accomplished this through the nature of their interactions. “It must be understandable to visitors and sufficiently rich in the depth of interactions so that visitors feel the presence of a basic unitary intelligence” (Eng, Babler, 2003). Ada tracks people while letting them know they are being tracked. Then she identifies those she deems more ‘interesting’ by their responsiveness to simple cues that she uses to probe their reactions. These people are encouraged to form a group through light and sound cues. Ada may then play one of a number of games with them while continuously “evaluating the results of her actions and expressing emotional states accordingly, and trying to regulate the distribution and flow of visitors. These four behavioural functions are decomposed into smaller behaviours that call on the core services as needed” (Eng, Babler, 2003).

Ada’s architecture is described in Figure 21 as a “series of levels with a gradient of decreasing biological plausibility as the proportion of traditional procedural code increases” (Eng, Babler, 2003). Each level is comprised of modules that communicate with other modules in the same layer and adjacent layers. “The metaphor used is that of distributed brain-like computation, characterized by tight coupling within individual modules and loose coupling between modules. The underlying software is a mixture of simulated neural networks, agent-based systems and conventional procedural or object-oriented software” (Eng, Babler, 2003). The different computing paradigms have been engineered to co-exist and cooperate usefully.

Comparative features:

There are obvious comparisons to be made between Ada and the Complex Adaptive House. As artificial organisms, they both rely on many sensors in competition and cooperation with each other in a distributed manner as highlighted in the literature review under the ‘complex adaptive

systems' section. They are both agent-based systems that learn and adapt to their environments. They both have a large number of inputs, however the Complex Adaptive House has many more different types, whereas Ada has many of the same types of inputs. The difference is that Ada tracks her visitors and lets them know, and is very interactive, whereas the Complex Adaptive House has very different goals and works in the background without calling attention to itself. It does not need to convince anyone that it is an artificial organism. Although their purposes differ, conceptually they are very similar. The fact that the Complex Adaptive House has many more differing types of inputs makes it more complex, and thus the possible amount of outputs increases dramatically, leading to more complex behaviour in the system.



Figure 22: People Interacting With Ada

Case Study 6: The Medina Smart Village

In 2002, a research fellow and two graduate students from MIT won 2nd prize in an international design competition for the design of a smart village in Egypt. The project explored “innovative environmentally responsible and socially proactive ways to build in developing countries” (Magnoli, Bonanni, and Khalaf, 2002). The project is unique in that it is based on a

genetic algorithm. The algorithm generated the architecture using the behaviour of biological creatures in artificial systems. It was fueled by renewable energies and built with recyclable materials with a flexible organization modeled on an ecosystem. The designers used concepts from artificial life using agents that follow simple low-level rules in order to achieve high-level group behaviour. They found that when they combined the principles of artificial life and architecture, “both high-energy efficiency and optimal placement of built units on a specific site” (Magnoli et. al., 2002) were the result. This flexibility is achieved at four levels: decentralized urban planning, reconfigurable spaces, modular building blocks, and kinetic structures that respond to light, climate and people.

The reason the designers approached it this way is because linear, static structures fail if just a single element fails because the cause-effect chain is interrupted leading to many detrimental consequences and unpredictable effects. A systemic structure such as they have proposed is “a highly-efficient, low-consumption circular structure where an element’s waste equals another element’s food. In ecosystems, as much as in human brains, the systemic structure performs even if more than one element collapses” (Magnoli et. al., 2002). Genetic structures are always more efficient than linear static structures. The designers used wrote a behavioural code to model a high degree of complexity of urban structures.

Some of the results were: “each building component has the same rights of any individual component of society, it can be plugged into a neural spine at any scale; each architectural element can be combined with any other, has access to physical and natural resource, and responds to individual inputs; each space and function can be adapted, reconfigured, and customized by people according to the democratic combination of individual pressures which leads to reducing both social and environmental costs; each structure reacts systemically to the

enormous amount of socio-environmental inputs” (Magnoli et. al., 2002). In this way each design element is able to be involved on the four levels previously mentioned.

1: Decentralized urban planning

“The primary motivation of the design lies in creating a design solution that is flexible and adaptive at any scale, and at instances, responsive and intelligently active with respect to the changing individual and climatic contexts” (Magnoli et. al., 2002). The goal was to provide a framework of flexibility that can be configured and reconfigured on two scales. The first was to satisfy the community with the ability to adapt and expand to meet the changing programmatic spatial requirements, and the second was at a local scale in order to optimize an individual building for changing environmental conditions. “Such adaptability then aims to meet the changing needs of the users and their activities/environment for comfort and optimum spatial efficiency and thermal conditions. Through the framework provided, which is rooted in Arabic vernacular layouts of towns, numerous configurations can be satisfied to accommodate the ‘modular design’ and ‘information technology’ philosophies” (Magnoli et. al., 2002).

The village is organized according to the sun which allows every building optimal light for the solar panels, as well as shadow for the users. There is equal access to outdoor and indoor public areas and services and the village plan is a single entity that adapts to any conditions. This takes the form of hexagonal, non-hierarchical units which are defined by a series of parallel walls that function as a ‘networked spine’ or sub-structure. Each cell of this honeycomb configuration is defined by both users and environmental conditions, which become increasingly varied in terms of usage. Smaller structures can be plugged into the ‘spine’ to create a secondary layered fabric. Secondary in-fill structures can be plugged into the network creating infinite arrangement

possibilities.

2. Reconfigurable spaces

The 18-inch double walled components along the modular ‘plug-in’ corridor contain all necessary electrical and HVAC equipment with a seamless flexible connection. Window and door layouts are reconfigurable along with plumbing fixtures. The rooms are attached to the corridor configuration with shared walls that can provide any two rooms with the capability of expansion to a shared common space.

3. Modular Building Blocks

There are small office rooms as well as large covered spaces containing smaller rooms so that larger spaces can provide thermally protected common areas with smaller support areas plugged directly into the spine. “The building as a whole will demonstrate how transformable objects can dynamically occupy predefine physical space as well as how moving physical objects can share a common physical space to create adaptable spatial configurations. Adaptability may range from multi-use interior re-organization to structure transformability. General implications include an adaptable response to spatial efficiency and environmental responsiveness” (Magnoli et. al., 2002).

4. Kinetic structures that respond to light, climate and people

These buildings and accompanying architectural objects use fewer resources and adapt to complex site and program requirements. They use natural daylight without interfering with visibility, and they take advantage of natural ventilation through a dynamic roof system. The operable windows allow for a wide range of response to changing environmental conditions. “Since the sun changes position, the roof keeps moving like a living structure” (Magnoli et. al.,

2002). On a winter day the kinetic structure opens to the sun and its warm light, and on a winter night, the roof closes to release the warmth accumulated by the thermal mass. On a summer day the roof denies the sun's rays in such a way as to allow cross ventilation. On a summer night the kinetic roof opens to allow the cool breeze.

The designers have created a decentralized with a distributed load on resources by giving each unit an ideal maximum and minimum viability level. The approach can be extended as it grows. They use an artificial life approach to optimally place units on a given plot of land. Using the genetic algorithm, size, density and other attributes of the units are decided according to the most positive impact they can have on the community. Attributes can be toggled so that different generations of units replace the parent unit until a fit child unit is found. The algorithm then measures the cost of these attributes in that location. Models can then be manually tweaked by architects or city planners. Attributes of units or plots can be chosen to reflect what matters most for each individual.

Another feature of the village is that the streets and squares are filled with responsive umbrellas that work as a neural system for buildings. These kinetic umbrellas behave together and respond to people staying under them or passing under them. They shield people from the summer sun and open when it is cold to preserve whatever heat is in the ground. Each umbrella is outfitted with an infrared sensor, a thermometer and a chip to sense people's movements beneath. The umbrellas have the capacity to learn the behaviour of people, whether alone or in groups.

Comparative features:

This project has some interesting implications for the future of the built environment as well as for the Complex Adaptive House, mainly the use of artificial life programming and their use of

genetic algorithms, as well as the decentralization and distributed resources. It is self-sufficient and extremely efficient, and built for adaptation and flexibility. The Complex Adaptive House is meant to be a single cell that is scalable. The Medina Smart Village is at that larger scale and is an artificial organism of a different sort. Decentralization is not incorporated into the infrastructure of the cities of the world as much as it should be. It is the notion of putting all one's eggs in one basket. This concept could and should apply to electricity, water and waste management. This project has taken the concept of the Complex Adaptive House and scaled it up to the size of a small city. The city planners of the world would be wise to study this project.

Case Study 7: SANETs

Dressler, Dietrich, German, and Kruger, (2009) have been researching self-organizing sensor networks and have developed SANETs (sensor and actor networks), which self-organize using some simple rules. These rules are promising for the Complex Adaptive House since this SANET system is mimicked on how a biological cellular network functions. The authors assure that this system is extremely energy efficient and is network-centric (distributed). According to their simulations, SANET outperformed classical ad hoc techniques.

First of all, there are a number of approaches that have been tried in relation to *autonomic networking*. One way is to cluster the sensors and actuators into groups that enable simple coordination and control strategies. Another is to group nodes according to main objectives (Dressler et. al, 2009). The SANET approach is a different communication and control paradigm within an emerging area of research known as *bio-inspired networking*, which is based on adaptation mechanisms of cell biology. This system processes received data to perform routing decisions and can generate data according to a set of rules, as well as initiating actuators. It “operates on sets of messages instead of applying rules to single data packets”

(Dressler et al., 2009). “This allows performing data aggregation and fusion algorithms without the need of maintaining dedicated meta information describing the message history” (Dressler et al., 2009). Messages remain until it decides that it is time for them to be discarded. This allows it to store information about its state without referring to data structures outside its set of rules. If need be, new rules can be applied to the histories of previously received messages. This system was designed to address the common obstacles in event-based coordination systems of scale, stability, and extensibility, and supports large scale networks and complex rule sets.

Like biological cellular transmission, in SANET, the information itself addresses the destination. Mimicking a cell network leads to an optimal information exchange. Furthermore, the authors maintain that “the interaction and collaboration between the nodes finally leads to an optimized system behaviour in an emergent way” (Dressler et al., 2009).

A flexible rule system is applied to evaluate received messages between agents and to provide programming in a similar way as a cell network. The local behaviour is controlled by a rule interpreter in the form of simple state machines and applies new rules to previously received messages. Three categories of actions are available: rule execution (operations on the received messages), node control (control of local node behaviour such as the addition of sensors), and simulation control (actions needed for experiment control without influence on the node behaviour).

1. Rule behaviour

The following actions are given as examples to be used for network-centric processing of messages.

-!stop – Early termination of the rule execution. Depending on the current state (number and kind of received messages) it may be necessary to stop the current processing of the rule set. The next iteration will start with the first available rule.

-!drop – Erases messages in the current set and needs to be called if messages have been successfully processed.

- !**dropduplicates** – All duplicates are discarded according to a unique identifier in each message. This command is needed to emulate.
- !**return** – A new message is created and appended to the source message set.
- !**returnAll** – Copies of all messages in the current set are created and stored in the source message set.
- !**send** – A new message is created and submitted to the lower layer protocol for transmission to neighbouring nodes.
- !**sendAll** – Copies of all messages in the current set are created and submitted to the lower layer protocol for the transmission to neighbouring nodes.
- !**actuate** – A message is sent to locally connected actuators.

2. Node control

Actions have been integrated to control local agent behaviour in order to allow the ability to enable/disable sensors and actuators as well as to modify the current rule set.

-!**controlSensor** – A control message is sent to all sensors. According to the submitted attributes in **\$control**, the behaviour of the sensors can be controlled: **rsnSensorEnable** and **rsnSensorDisable**, **rsnSensorSetType** updates the type field of the sensor, and **rsnSensorSetMeasuringInterval** changes the sampling frequency.

-!**controlActuator** - This command controls actuators. The attribute **\$control** defines the action: the actuator is enabled or disabled by **rsnActuatorEnable** or **rsnActuatorDisable**, and **rsnActuatorSetType** updates the type field of the actuator.

-!**controlManagement** – The management plane defines the rule set itself. The **\$control** attribute is used to specify the intended action: the rule interpretation can be started or stopped by **rsnManagementFromRsnString** or **rsnManagementFromRsnFile**, and the evaluation interval can be configured by **rsnManagementSetEvaluationInterval**.

3. Simulation control

The following actions have been integrated for simplified control of simulation experiments and are initiated based on the current state of an agent after the reception of a specific message.

-!**recordAll** – Statistics are recorded for all messages in the current working set. The following information is stored: ID of the current node, ID of the node that generated the message, node specific ID of the message, globally unique ID of a message, hop count, current time, and delay (elapsed time since message creation).

-!**endSimulation** – This action terminates an experiment.

In addition, each message is specifically encoded to allow receiving nodes to determine the meaning of the message and the required behaviour. This encoding can be changed according to the application scenario. Each agent can store and update state information locally. In the

context of self-organization, this refers to the local state of an autonomous system. Information can be updated according to received messages or by other local observations.

Message attributes:

Attribute	Description
\$name	Descriptive name of the message
\$type	Type of message; describes the content
\$position	Position of the source node
\$hopcount	Number of traversed nodes
\$priority	Importance factor of this message
\$length	Length of the message
\$creationTime	Timestamp describing the creation of the message
\$value	Message type specific value
\$text	Further informative text to qualify the value

Node attributes:

Attribute	Description
:count	Number of messages in the current working set
:totalMessageCount	Number of all messages received by the node
:hostname	ID of the current host
:position	Position of the node
:random	Random value for probabilistic decisions

Endowing a network with organic computing provides abilities such as self-organization, self-regulation, self-repair, and adaptation. This means that the designer's task is simplified considerably because it is not necessary to specify low-level system behaviour for every occurring situation or perturbation. Instead, high-level goals can be specified and the system is free to discover its own solution pathways (Dressler et al., 2009).

Comparative features:

Like the Complex Adaptive House, these designers have based their work on mimicking their system on the communication functionality of cell biology. This study justifies the fact that this distributed method rather than centralized control is more advantageous when a system is large and complex. Mimicking cell biology is more efficient because it allows for very specific

responses to problems, it shortens information pathways, and it directs each problem to the appropriate helper component. This case study provides insight as to how the Complex Adaptive House would be programmed in terms of its sensor/actuator network, where the microcontrollers that are attached to the sensors and actuators can be outfitted with this type of algorithm.

Case Study 8: SuOC

Every year more and more appliances and devices are equipped with embedded smart technology and wireless communication. This gives them to become flexible and multifunctional. However, this increase in complexity results in unreliability. A German team has addressed this issue by creating a software architecture based on organic computing. They created a system that has features such as self-organization, self-configuration, self-repair, and adaptation. The reason this is advantageous is that “the designer’s task is simplified because it is no longer necessary to exactly specify the low-level system behaviour in all possible situations that might occur, but instead leaving the system with a certain degree of freedom which allows it to react in an intelligent way to new situations” (Branke, Mnif, Muller-Schloer, and Prothmann, 2007). This means that using the system is simplified by setting a few high-level goals rather than having to manipulate a lot of low-level parameters whose influences are uncertain. The designers of this architecture have chosen to exemplify it into a traffic control system, however this project is about creating a generic template for implementing life-like computing systems for a wide variety of technical applications.

Some technical drawbacks to programming a bio-inspired network system of self-organization and adaptability are that in order for the system to learn, it has to be allowed to make some early errors, it may react slowly at first, be unpredictable and difficult to control, much like any

infant. Therefore, a ‘happy medium’ must be found between creative self-organized bottom-up processes, and top-down control. According to Branke, et.al., (2007). to this end, an observer/controller architecture has been designed in which an observer (built into the system) can detect errors early on and execute appropriate counter measures. Thus, self-organization is allowed while emergent behaviour is monitored and controlled.

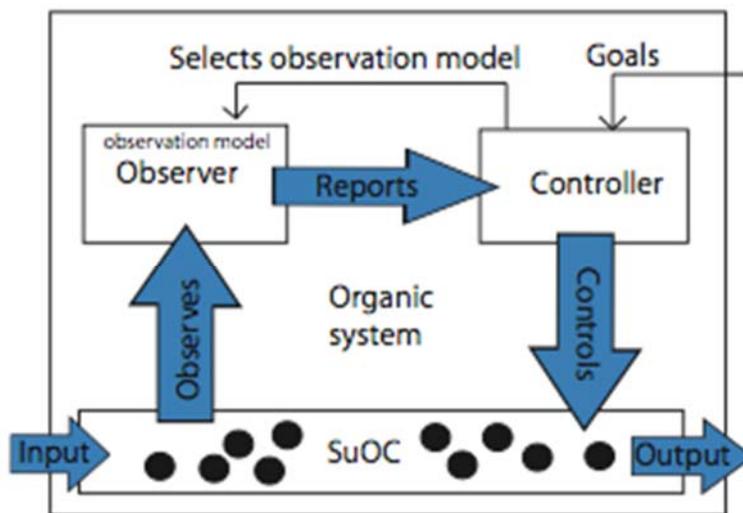


Figure 23: Observer/Controller Architecture

The authors state that the SuOC (system under observation and control) consists of a large number of communicating agents (sensors and actuators), and the observer collects data from both micro and macro level states (situation parameters) which give indications about the overall state and the dynamics of the system (Branke et. al., 2007). The controller compares these situation parameters with the intended goals and can then decide if an intervention is required and what action to take. Actions taken may be: “the local decision rules of the SuOC elements, the system structure including e.g. the communication between the SuOC elements or the number of elements, or the environment, which will indirectly influence the system by changing the data observed by the SuOC elements through their local sensors” (Branke et. al.,

2007). The observer/controller interferes only if necessary and does not control single elements, but rather system parameters.

In respect to the Complex Adaptive House, it is worth going into more detail on how this network system functions so that a better picture may be gained on its feasibility. The observation process of the SuOC is comprised of several steps beginning with steps monitoring, followed by pre-processing, then data analysis, prediction, and finally aggregation.

The controller guides and controls the self-organization process derived from the actions or data given by the observer. The main component of the controller is a decision module called an ‘action selector’ which ultimately selects the appropriate action quickly in real-time. There is also an adaptation module imbued with learning and planning capabilities by evaluating previous actions in order to improve the mapping of situations and actions by way of a history data base. The adaptation module can be connected to a simulation module which allows it to approximate the consequences of possible actions before executing them in the real system. The adaptation module’s purpose is to match the system behaviour with the goals formulated by the designer or user (Branke et. al., 2007).

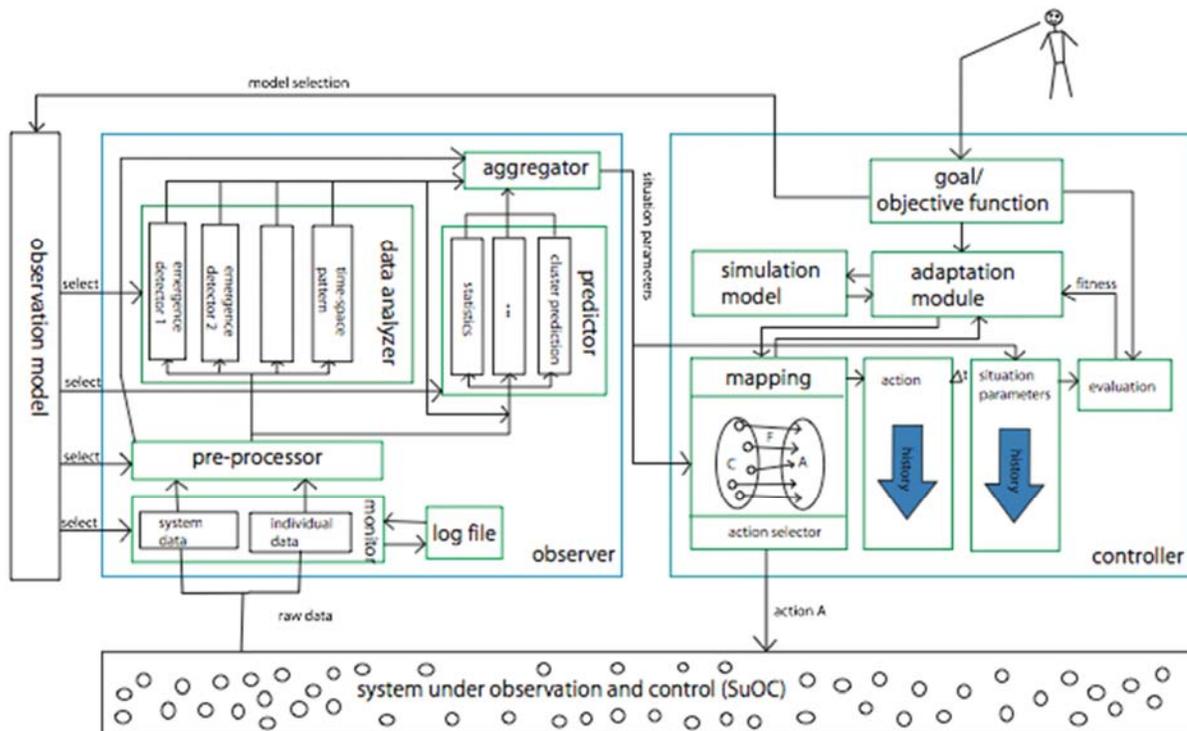


Figure 24: Generic Observer/Controller Architecture

The system must be customized to various scenarios by adapting the components of the observer and controller. This system can be designed in three ways: “(a) *Central*- an observer/controller for the whole technical system. (b) *Decentral*- an observer/controller on each system element. (c) *Multi-level*- an observer/controller on each system element as well as one for the whole technical system” (Branke et. al, 2007). This third option is the obvious choice for the Complex Adaptive House since it mimics the natural hierarchic structure of an organism.

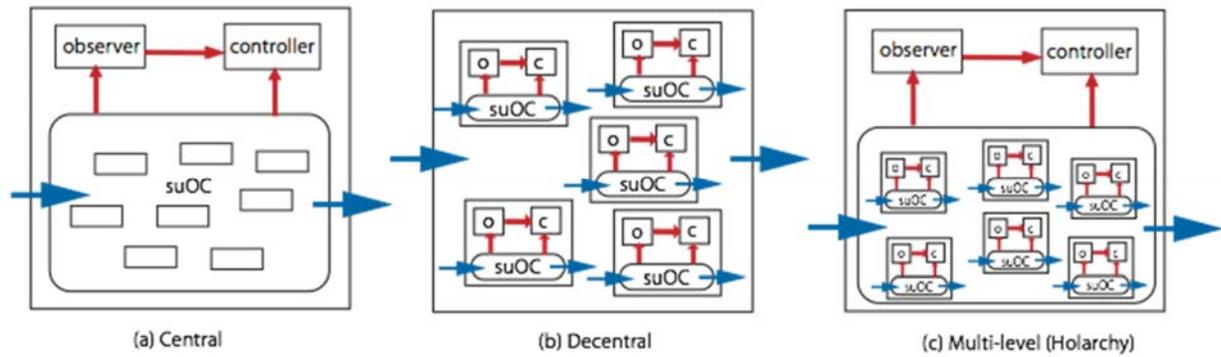


Figure 25: Observer/Controller Realization

Comparative features:

The designers of this project understand that as devices and systems increase in complexity, traditional systems will increasingly fail. It is not possible to design every reaction to all possible variables between the environment and human variability. Therefore this project aims to provide a generic architecture that can be applied to a myriad of projects including the Complex Adaptive House. The system architecture outline here can self-organize its agents autonomously independent of explicit external interferences. The program encourages adaptation to develop robustness, and it behaves in a flexible way to modify its behaviour in response to changing requirements. In addition, the observer/controller architecture allows for external control over self-organizing behaviour so that emergence does not turn the system into a 'wild thing' using higher level objective functions within multi-level organic computing architectures. The designers understand that systems that exhibit life-like qualities such as those mentioned are far more equipped to manage complexity with a high degree of reliability than traditional systems. There are many possible ways one could go about programming the Complex Adaptive House, and this particular software package is an excellent contender.

Case Study 9: The SodekoVS Project

The SodekoVS project was initiated to address the challenge of engineering adaptation in various systems. The approach considers system architecture and software development methodology as integrally intertwined aspects for system construction. The reason for this endeavour is that self-organizing systems promise new software quality attributes that cannot be obtained using standard approaches. Self-organizing systems promote self-adaptability, self-configuration and self-protection, which means that it can manage itself at runtime. “Self-organizing dynamics, inspired by biological, physical and social systems, can be integrated into applications by composing modules that distribute feedback control structures among system entities. These compositions support hierarchical as well as completely decentralized solutions without a single point of failure” (Sudeikat, Braubach, Pokahr, Renz and Lamersdorf, 2009).

Application scenarios given in this project include small satellites that are able to perform a mission in concert, underground urban transport systems consisting of many autonomous vehicles, each responsible for reaching a target area, and the monitoring and automatic reconfiguration of server farms in case of changing environments. Although none of these resemble the Complex Adaptive House, the concepts are completely applicable.

Because single entities of the systems may fail at any time, such as a satellite runs out of energy or a server has a hardware failure, it is crucial that those failures not harm the overall system.

“All mentioned examples have in common that they assume a decentralized infrastructure consisting of a multitude of autonomous entities, which have to interact in order to perform the intended functionalities. This high number of entities as well as the demand for autonomous behaviour in combination with coordinated actions requires novel software concepts. In addition, the ability to systematically construct these kinds of systems is also an important aspect, which is not well supported until now” (Sudeikat et. al., 2009)

These problems will be addressed by using nature-inspired design paradigms through coordination strategies to equip software with adaptability and robustness, based on decentralized, self-organizing principles. Thus far, software of this type has only been used for specific applications and is not reusable. This project aims to engineer software that is generic enough to be used in a large variety of projects.

The tools required for the development of self-organizing applications are field-tested with coordination strategies, and are bottom-up and simulation-based. This provides practical guidance on how to conceive and revise non-linear system properties. Self-organization in software for this project is tackled first with a reference architecture with a conceptual framework for its configuration and integration. Middleware services are required as coordination mechanisms.

A library of coordination mechanisms is needed in order to apply self-organizing software to specific problems as well as a generic user interface which allows their reuse. “The provided library provides a catalogue of mechanism patterns as reusable components that provide systematic problem-oriented descriptions of mechanisms in an abstract, reusable format. This facilitates the selection and combination of mechanisms” (Sudeikat et. al., 2009).

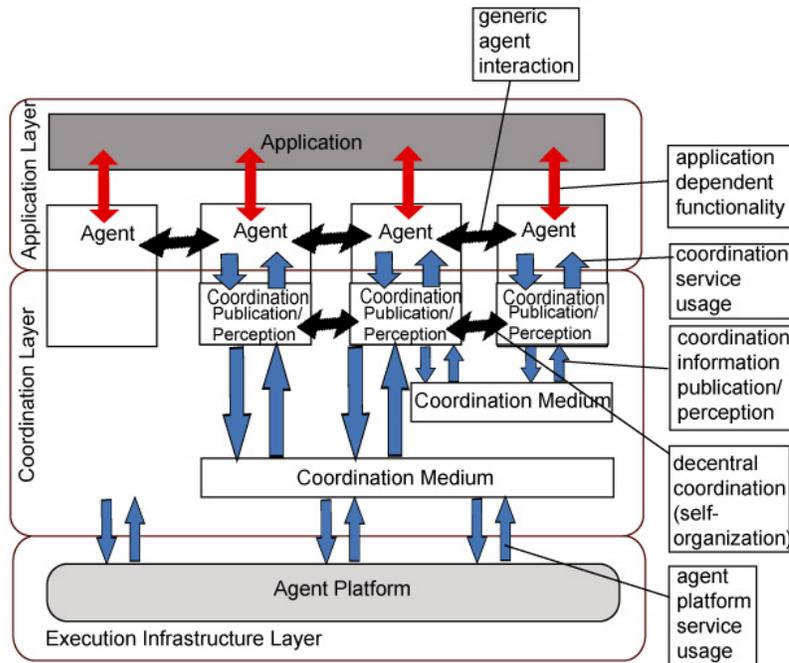


Figure 26: SodekoVS Reference Architecture

Figure 26 shows a blueprint for the SodekoVS reference architecture for self-organizing applications. There are three abstraction layers: the first layer at the top has standard functionalities and is linked to agents which are responsible for achieving self-organizing characteristics. The coordination layer below it contains agents and a substrate containing coordination media. Distinct coordination media contain mechanism instances which are interfaced by coordination components that can modify agent states. Coordination strategies are enacted by: “ (1) configuring the component internal information to be exchanged, (2) defining the dynamics of the information flows between components and (3) declaring how individual components adjust their local activities. Finally, the execution infrastructure layer is responsible for providing basic services to the coordination layer. These services e.g. include the agent management and execution” (Sudeikat et. al., 2009).

This reference architecture is made to resemble nature-inspired self-organization strategies and

to be integrated into conventional software systems. Because the research team is after a generic adaptive, self-organizing system and not something specific, conducting tailored simulations is essential since doing real-world testing would be impractical and extremely difficult.

The team has identified distributed control loops as “the driving force of self-organizing dynamics, and their explicit modeling promises an approach to relate coordination models to entity designs” (Sudeikat et. al., 2009). Their modeling techniques are inspired by System Dynamics (SD) concepts, which provides descriptions in terms of system state variables such as accumulative values of system qualities, and causal relations between them that identify rates of change among the values. This demonstrates how system elements mutually influence and facilitate the description of feedback loops. Transferring these ideas to the realm of agent-based designs has allowed researchers to describe how agents influence each other and their environment.

Comparative features:

The implications of this project are important for the Complex Adaptive House in terms of a generic system that can be tailored to fit any system. Every system has its own programmed goals. If the SodekoVS system is combined with the goals of the Complex Adaptive House via all the microcontrollers in the sensors and actuators, this might very well constitute a software package complete with simulation software.

Case study 10: SOTA

The increasing complexity and dynamics in which software systems are implemented require solutions to make such systems autonomic. In other words, capable of dynamically self-adapting their behaviour in response to changing situation parameters. To this end some Italian researchers from the University of Modena have developed a new general model for modeling

adaptation requirements which they have named SOTA (State Of The Affairs). This system has the ability to tackle key issues in the design and development of complex self-adaptive software systems by bringing together the lessons of goal-oriented modeling and context-aware system modeling. SOTA enables early verification requirements, identification of knowledge requirements for self-adaptation, and identification of the most suitable self-adaptive patterns.

The key idea in SOTA is to perceive a self-adaptive software system as a sort of complex dynamic system immersed in a virtual n-dimension phase space, each dimension being associated to an internal or environmental parameter of interest for the execution of the system. “Functional requirements (goals) are associated to areas of the phase space the system has to reach, non-functional requirements are associated to the trajectory of the system should try to walk through, self-adaptation is associated to the capability of the system to re-join proper trajectories when moved away from it” (Abeywickrama, Biccocchi, and Zambonelli, 2012).

SOTA exhibits several advantages and proposes itself as a general purpose and effective framework in the following ways: “it can be used to early assess self-adaptation requirements via model-checking techniques, it can be used as a tool to support the process of identifying which knowledge must be made available to the system and its components, to support adaptation, and it can effectively support designers in the most appropriate internal software structures and processes to adopt in self-adaptive system design” (Abeywickrama et. al., 2012).

SOTA has a unique take on sensors in that it virtualizes them. Virtual sensors are useful for “grouping a number of physical sensors for the sake of fault tolerance, converting sensor readings into relevant information, and grouping different physical sensors allowing multi-modal recognition capabilities” (Abeywickrama et. al., 2012). The issue of identifying which

kinds of virtual sensors are required to enable and facilitate adaptation is necessary to drive activities related to knowledge modeling and processing.

Furthermore, the authors have defined a taxonomy of architectural patterns for adaptive components and ensemble components. At the center of this taxonomy is the idea that self-adaptivity requires the presence of a feedback loop or control loop. “A feedback loop is the part of the system that allows for feedback and self-correction towards goal achievement, i.e., self-adjusting behaviour in response to the system’s changes. It provides a generic mechanism for adaptation where it provides means for inspecting and analyzing the system at the service component or ensemble level and for reacting accordingly” (Abeywickrama et. al., 2012). In SOTA, they use feedback loops to describe how goals and utilities are expressed in the adaptive patterns.

The primitive service component is a non-adaptive pattern provides functions. The reactive service component modifies its behaviour and reacts to environmental perturbations. It has no feedback loops present, so the adaptation is a result of its interactions with the environment. The reactive service component has the capability of perceiving the SOTA space and applies actions to modify its position in the space in the medium term (it has utilities), but it has no means to aim in the long term, so it does not have goals.

“The goal-oriented service component has internal control loops and the adaptation is explicit. It not only reacts to external events but also actively tries to adapt its goal or utility-oriented behaviour. In addition to having utilities, this component can apply actions in the SOTA space in the long term (it has goals)” (Abeywickrama et. al., 2012).

The autonomic service component features an explicit external feedback loop. This pattern is comprised of a controller service component and an autonomic control manager at the end of the loop to monitor and direct behaviour. The services, goals and utilities of the controller service component are tightly coupled with the autonomic control manager. Like the goal-oriented service component, this pattern has goals and utilities.

“Adaptation patterns defined for the ensemble level are categorized based on features such as the implicit or explicit nature of the feedback loops and the pattern’s goals” (Abeywickrama et. al., 2012). Three key pattern categories are identified: (a) environment mediated: “ensembles in which the adaptation activities are not explicitly designed but emerge from the interactions of the components and the shared environment; (b) negotiation/competition: ensembles in which the adaptive behaviour is explicitly designed using interaction patterns between components; (c) norm based: ensembles in which there is a set of components with a global control loop over the ensembles to control and direct the overall behaviour” (Abeywickrama et. al., 2012).

Comparative features:

This project features another method of going about the realization of how the Complex Adaptive House can be programmed in the way that I have described a decentralized sensor/actuator network that is self-organizing and adaptive. SOTA provides a general framework that can be implemented into the microcontrollers within the sensor network as well as some insight into the nature of feedback loops which figure heavily in the overall system of the Complex Adaptive House. SOTA is also very goal-oriented which is advantageous since the house requires goals so that homeostasis is preserved, and emergent behaviour is controlled while still allowing adaptation to occur.

4.4. Scenarios

Climate change is predicted to unfold in a variety of ways in a non-linear and unpredictable ways. However, it is certainly possible to predict what kind of disasters can occur. The next three scenarios depict some of the most likely environmental events. Since this particular house is a steel and concrete-framed construction, earthquakes are not of particular concern. Tornados are another matter that is not addressed simply because the only way for any structure to survive a direct hit is if it is reinforced beyond the scope of this project. The fourth scenario is meant to illustrate adaptation to human behaviour in a day-to-day setting. Most of our infrastructure is centralized which is problematic when the power lines go down, or there are long periods without precipitation or contamination of a central water supply. Waste treatment facilities should also be more decentralized.

Scenario 1: Drought

In the event that there is a prolonged period without rain and conditions are very hot and dry, although there will be no shortage of electricity from the PV array, it will become necessary to conserve the water that has been harvested beforehand and stored in the cisterns. It begins with the weather station on top of the greenhouse combined with the local forecast which tells the system that it might be a while before new rain can replenish the water supply. The system goes into water conservation mode, as the sensors in the cisterns measure the water levels. Outdoor irrigation ceases in case the water within the irrigation cistern needs to be diverted for indoor use. In the greenhouse, moisture sensors will become more stringent about how much the crops get watered, giving them just enough so that they do not die. Digital water meters posted at every faucet will tell the inhabitants at every opportunity how much water they are using and giving warnings when necessary. Auto shut-off mechanisms can kick in to ensure that the least

amount of water is wasted. A countdown display in the shower can inform the user how long they have to clean themselves before the shower shuts off.

In this scenario the house also needs to keep itself cool. Deep overhangs and balconies ensure that the summer sun does not touch the southern glazing let alone penetrate into the house.

Fresh air intake enters low to the ground at the shaded North side of the house and is exhausted at the top of the South side, creating a solar chimney effect. This is accomplished with sensors telling actuators to open vents and run fans. In addition, the geothermal system can cool down the internal environment by pulling the heat from the house back into the ground. Since it has a variety of ways to choose from, the house will make its own decisions based on efficiency, and not because any specific pre-programmed action.

Scenario 2: Cold dark winter

In this scenario, the house cannot rely on the PV array or solar hot water. During dark winter days, thermal mass is rendered useless so other means of acquiring heat and electricity are needed. Luckily, this time of year tends to be windy so the wind turbine will supplant the PV array. The geothermal system will pull most of the heat required (up to 80%), and the rest can be supplemented by an electric furnace as needed. A high-efficiency air-to-air exchanger or heat recovery ventilator (HRV) will minimize this need by transferring the warmth in outgoing air to the incoming cold air before it is transferred to the furnace. Finally, if one of these should fail for some reason, a high-efficiency analog fireplace can be utilized. The thermal mass in the floors will also retain some of the non-solar heat produced within the house and release it should the indoor temperature drop below a certain threshold. The house will be sure to have the battery bank fully charged. Snow can be harvested instead of rain, although with more difficulty. In the greenhouse, grow lights respond to natural light levels, or rather lack thereof.

The house will behave differently in the winter than summer; in winter the windows are closed while in the summer the windows will be opening and closing periodically as the weather dictates. In every season the goal of the house is to maintain homeostasis as dictated by the occupants. How the house accomplishes this differs from season to season.

Scenario 3: Storms

One of the terrifying features of climate change is bigger, more ferocious storms. High winds and lightning can take down power lines which can leave people without power for days or weeks. The Complex Adaptive House uses these destructive events to its advantage by harvesting those high winds and heavy rains. A storm is an opportunity for the wind turbine to charge the battery bank to full capacity and to fill the cisterns, although if winds become too high the turbine shuts down to avoid damage. Windows will close automatically. If there is a sudden temperature drop, the house automatically makes any required adjustments. It is also an opportunity to replenish water levels in the cisterns. The large sloping roof has several downspouts for rain to be funneled into the cisterns, and at ground level various catchment systems can be implemented as well.

But what if the storm is really, really bad to the point where damage occurs? If a sensor network is set up like a decentralized neural network, and has been trained to perform various tasks, it will still be able to perform those tasks if some part of the system is damaged. Damage will over time decrease performance, but it will do so gracefully, not as a complete system failure.

Traditional systems simply stop working when certain components are removed. Other agents can pick up the slack when damage occurs.

Scenario 4: Adaptation to human behaviour

Adapting to changing weather patterns should be relatively simple for the house compared to adapting to the habits of its inhabitants and visitors. Sensors present in household appliances will aid in this adaptation. The habits of the inhabitants will inform the personality that the house develops organically. Sensors in all the faucets, showers and toilets will inform it about water usage habits, and the house in turn will ensure that people are not being wasteful (especially children), thus altering the habits of the users. The house will heat the TV viewing area if that is where all the inhabitants are clustered, and it will learn that the inhabitants might take bathroom breaks, or trips to the kitchen for snacks, and when people go to sleep, heat will be diverted to the upstairs. What this concept contains that every other smart house lacks is adaptability. In a normal smart house, if occupant behaviours change, there can be frustration that the house is not performing to their desires or comfort level. The Complex Adaptive House has the unique ability to adjust its performance without the need for the occupants to reprogram it.

5.0. CONCLUSIONS

“In wildness is the preservation of the world.” -Thoreau

Smart houses are becoming increasingly available, and smart house components are available at retail stores so that homeowners can make parts of their houses smart one piece at a time. They follow a centralized model that is predictable but subject to the same failures as any other house that relies on the grid. What is proposed here is an alternative model that is decentralized and autonomous, making it a more logical and practical model in the face of impending climate change. It relies on artificial life programming rather than artificial intelligence so that it will display behaviour and have somewhat of a personality, but not a mind. An analysis of the existing literature has revealed that no one has yet attempted to implement this model into a

smart house, but there is much experimentation occurring with agent-based self-organizing systems and complex adaptive systems all over the world.

At the beginning of this research, an unknown factor was whether emergence would happen once the system was in play. In fact it is impossible answer this question without simulations or actually building the Complex Adaptive House and studying it. However, throughout the course of research I have discovered that emergence is more of a certainty. The real question is how much of a wild thing it will become? It will inevitably take on a distinct personality; in response to perturbations it will have a variety of responses. The exact mix of responses *is* its personality. It depends on the personality it is given, but if feedback loops are implemented into a system that interacts with people, it will take on a personality automatically.

The best way to think about such a house is as a series of coupled non-linear feedback loops that all deal with different parts of the system, and that they are coupled together in ways that they can trade off between systems. For example, temperature maintenance is a non-linear feedback loop, and so is energy maintenance, and they are coupled together. Each system has specific goals which are pushed towards optimization. It will have different choices from which it must decide which we cannot predict. Its personality will emerge from these choices as well as the commands that the user dictates. So far, computer scientists do their best to avoid highly coupled non-linear feedback loops. Most areas prefer deterministic and predictable programming, and adaptation is avoided because of unpredictability and the difficulty of debugging. So for the purposes of this exercise, adaptation must happen within certain confines or it could turn wild very fast.

To build a house that is a true complex adaptive system would not be preferable to its inhabitants because it would push back if the behaviour of the users runs counter to the goals of the house. A pure complex adaptive house is definitely feasible, but some degree of control is required. A hybrid system is the optimal choice so that it can self-organize and adapt within specific confines. Adaptation will occur bottom-up, top-down, and even sideways. A central hub is not required; decentralization is preferable because a window can tell the user that there is something wrong with the door, it does not need to go to a central hub. This leads to robustness because agents pick up the slack when another agent experiences a failure. However, a central hub can have its uses, much like humans benefit from a central nervous system. A central hub can make executive policy decisions based on the adaptations of the agents. It will become a *tangled hierarchy*. Rather than letting the house choose some emergent temperature, or presetting one temperature at all times, the user should reserve the right to change the temperature, and the house can accommodate for that however it chooses.

For the purposes of this exercise, the Complex Adaptive House is situated in the middle of nowhere, far from any existing infrastructure. But, this new model of decentralized organization, from the autonomous house, to a neighbourhood cell, to the city proper, composed of decentralized self-organizing cells that are all linked to a smart grid, should be the *infrastructural hermeneutical foundation of the future*. It can be thought of as a cell within an organism, within an ecosystem, within a bioregion; hierarchical and self-organizing. This equates to a house within a neighbourhood, within a ward, within a city.

The green building movement is thriving all over the world as architects and engineers seek new and innovative ways of creating efficient and healthy buildings. The green movement and

the smart home revolution also need to come together in new and innovative ways. The Complex Adaptive House is the answer.

The up-front costs will pay for themselves over time, and it would be in everyone's best interests to have this new model established before disaster strikes, which could substantially lower rebuilding costs and time, and a possible death toll. Rather than to be at the mercy of the ravages of climate change, this model seeks to take what nature throws at it and use it to its advantage whenever possible, since it learns from, and adapts to, environmental perturbations.

The notion of a Complex Adaptive House that is modeled on a living organism is, as of yet unheard of, but due to the mounting levels of complexity that are being introduced into daily human experience coupled with the super wicked problem of climate change, this dwelling model seems the most prudent, especially in conjunction with a smart grid at larger scales.

The properties of Life have been analyzed by scientists with a top-down approach only to identify that organic systems must be modeled using a bottom-up approach. The processes that underlie Life are described by autopoiesis in biological terminology, and by dissipative structures in thermodynamics. Life exists at the edge of chaos, far from thermodynamic equilibrium. Organisms can be described as complex adaptive systems, of which the process of self-organization is key. The process of self-organization has been successfully simulated and can be replicated in the laboratory. Simulations have yielded learning algorithms, genetic algorithms, evolutionary algorithms and so on, which are different ways of addressing the same thing. By mimicking these bio-inspired principles, software has been developed that can dictate the behaviour of a sensor/actuator network to perform in the same manner as an organism.

These programmable network systems are used currently for a variety of research topics such as

self-organization in traffic control systems, and the upcoming electrical smart grid systems. This is a new and developing field of interest, and although the research has not pointed to any of this being applied to a smart house, the research does clearly indicate that such an endeavour is completely plausible. In fact, the section about automation in greenhouses in the appendix demonstrates that research and implementation of adaptive automation systems have been used successfully in a variety of ways for many years.

An immediate challenge would be to find the exact algorithm and conduct simulations. After which, the next challenge would be to customize the software for the Complex Adaptive House, and if successful, to standardize it as a software package, much like the SodekoVS project which is generic and customizable. Thereafter, software packages at larger scales could be developed.

"As we build, so shall we live." Richard Register

REFERENCES

- Abeywickrama, D., Zambonelli, F. (2012). SOTA: Towards a General Model for Self-Adaptive Systems. *IEEE 21st International WETICE*.
- Ahmar, S. (2011) *Biomimicry as a Tool for Sustainable Architectural Design: Towards Morphogenetic Architecture* (Master's Thesis, Alexandria University).
- Albright, L., Arvanitis, K., & Drysdale, A. (2001). Environmental Control for Plants on Earth and in Space. *IEEE Control Systems*, 21 (5), 28–47.
- Arrhenius, S. (1896). On the Influence of Carbonic Acid in the Air upon the Temperature of the Ground. *Philosophical Magazine and Journal of Science*, Series 5, Volume 4, April 1896, pp237-276.
- Ashby, W. R. (1952). *Design for a Brain: The Origin of Adaptive Behavior*. Chapman and Hall, London.
- Ashby, W. R. (1956). *An Introduction to Cybernetics*. Chapman and Hall, London.
- Banzhaf, W. (2009). Self-Organizing Systems. *Encyclopedia of Complexity and Systems Science*. Springer, Heidelberg, 8040-8050.
- Branke, J., Mnif, M., Muller-Schloer, C., Prothmann, H. (2007). Organic Computing- Addressing Complexity by Controlled Self-Organization. *Second international Symposium on Leveraging of Formal Methods, Verification and Validation*. IEEE.
- Brooks, R. (1999). *Cambrian Intelligence: The Early History of the New AI*. Cambridge, Massachusetts: The MIT Press.
- Brown, T., Conrad S., Beyeler, W. (2012). *Complex Adaptive Systems Engineering: Improving our Understanding of Complex Systems and Reducing Their Risk*. Earth Systems Engineering, Sandia National Laboratories. Newcastle, United Kingdom.
- Brueckner, S. (2000). *Return from the Ant: Synthetic Ecosystems for Manufacturing Control*. PhD thesis, Institute of Computer Science, Humboldt University, Berlin, Germany.
- Caponetto, R., Fortuna, L., Nunnari, G., Occhipinti, L., & Xibilia, M. (2000). Soft Computing for Greenhouse Climate Control. *IEEE Transactions on Fuzzy Systems*, 8 (6), 753–760.
- Capra, F. (1996). *The Web of Life: A New Understanding of Living Systems*. New York: Random House.
- Capra, F., Luisi, P. L. (2014). *The Systems Way of Life: A Unifying Vision*. U.K. Cambridge University Press.
- Carr, J. (1993). *Sensors and Circuits: Sensors, Transducers, and Supporting Circuits for Electronic Instrumentation, Measurement, and Control*. New Jersey: PTR Prentice Hall Inc.

- Chen, S. (2009). The Autonomous House: A Bio-Hydrogen Based Energy Self-Sufficient Approach. *International Journal of Environmental Research and Public Health* (pp. 1515-1529). Vol. 6.
- Cook, D., Youngblood, M., Das, S. (2006). A Multi-Agent Approach to Controlling a Smart Environment. Published in J.C. Augusto and C.D. Nugent (Eds.): *Designing Smart Homes*, LNAI 4008, pp. 165-182, 2006. Springer-Verlag Berlin Heidelberg 2006.
- Dressler, F., Dietrich, I., German, R., Kruger, B. (2009). A Rule-Based System for Programming Self-Organized Sensor and Actor Networks. *Computer Networks*, Vol. 53 pp. 1737-1750.
- Eng, K., Babler, A. (2001). Ada: Buildings as Organisms. *Game, Set and Match* (Conference Proceedings), Faculty of Architecture, TU Delft, Holland. Nov. 13-14.
- Eng, K., Babler, A. (2003). Ada-Intelligent Space: An artificial creature for the Swiss Expo.02. *Proceedings of IEEE/RSJ International Conference on Robotics and Automation (ICRA 2003)* Taipei, Taiwan, September 2003.
- Fang, X., Jialiao, C., Libin, Z., Hongwu, Z. (2006). Self-Tuning Fuzzy Logic Control of Greenhouse Temperature Using Real-Coded Genetic Algorithm. *Proceedings of IEEE International Conference on Control, Automation, Robotics, and Vision*, pp. 1–6.
- Fourati, F., & Chtourou, M. (2007). A Greenhouse Control with Feed-Forward and Recurrent Neural Networks. *2007 Simulation Modelling Practice and Theory*, 15, 1016–1028.
- Frei, R., Di Marzo Serugendo, G. (2011). Concepts in Complexity Engineering. *Bio-Inspired Computation*, Vol. 3. No. 2.
- Frei, R., Di Marzo Serugendo, G. (2011). Advances in Complexity Engineering. *Bio-Inspired Computation*, Vol. 3. No. 4.
- Fromm, J. (2006). *On Engineering and Emergence*. Distributed Systems Group, EECS Department of Electrical Engineering and Computer Science, Kassel University, Germany.
- Gatti, M., de Lucena, C., Alencar, P., Cowan, D. (2008). Self-Organization and Emergent Behavior in multi-Agent Systems: a Bio-inspired Method and Representation Model. *Monografias em Ciência da Computação*, No. 19/08.
- Gershenson, C. (2007). *Design and Control of Self-Organizing Systems*. PhD Dissertation. Faculty of Science and Center Leo Apostel for Interdisciplinary Studies, Vrije Universiteit, Belgium, Brussels.
- Guerin, S., Kunkle, D. (2004). Emergence of Constraint in Self-Organizing Systems. *Nonlinear Dynamics, Psychology, and Life Sciences*, Vol. 8, No. 2, April, 2004.

Henderson, G. (1993). Housing, Energy and the Environment. *Proceedings of the Innovative Housing '93 Conference* (21-25 June 1993; Vancouver: Ministry of Supply and Services, Canada 1993), Volume 1, p.5.

Heylighen, F. (1994). Fitness as Default: The Evolutionary Basis for Cognitive Complexity Reduction. *In Cybernetics and Systems '94*, R. Trappl, (Ed.). World Science, Singapore, 1595-1602. 20

Heylighen, F. (2001). The Science of Self-Organization and Adaptivity. *The Encyclopedia of Life Support Systems* (pp. 253-280). Vol. 5, Issue 3.

Heylighen, F., Gershenson, C. (2003): "The Meaning of Self-organization in Computing", *IEEE Intelligent Systems*, section Trends & Controversies - Self-organization and Information Systems, May/June 2003.

Ho, M. W., (2008). *The Rainbow and the Worm: The Physics of Organisms*. World Scientific Publishing Co. UK.

IPCC (2014). Climate Change 2014: Synthesis Report. Contribution of Working Groups I, II, and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change [Core Writing Team, R.K. Pachauri, L.A. Meyer (eds.)].

Kauffman, S. (1993). *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press. New York.

Kemp, W. H. (2005). *The Renewable Energy Handbook: A Guide to Rural Energy Independence, Off-Grid and Sustainable Living*. Aztext Press. Tamworth, Ont. Canada.

Kim, K., Cho, S. (2006). A Comprehensive Overview of the Applications of Artificial Life. *Artificial Life* (Vol 12: 153-182).

Kolokotsa, D., Saridakis, G., Dalamagkidis, K., Dolianitis, S., & Kaliakatsos, I. (2010). Development of an Intelligent Indoor Environment and Energy Management System for the Greenhouses. *2010 Energy Conversion and Management*, 51 (1), 155–168.

Kramer, J., Magee, J. (2007). Self-Managed systems: an Architectural Challenge. *Future of Software Engineering (FOSE)*, pp. 259-268, IEEE Computer Society, Washington DC, USA.

Kwok, A., Grondzik, W. (2007). *The Green Studio Handbook: Environmental Strategies for Schematic Design*. Elsevier, Burlington, MA, USA.

Lafont, F., Balmat, J. (2002). Optimized Fuzzy Control of a Greenhouse. *Fuzzy Sets and Systems*, 128, 47–59.

Langton, C. (1989). *Artificial Life*. Proceedings of an Interdisciplinary Workshop on the Synthesis and Simulation of Living Systems. Addison-Wesley.

Langton, C. (1990). Computation at the Edge of Chaos: Phase Transitions and

Emergent Computation. *Physica D* 42. (pp. 12-37). Elsevier Science Publishers, Holland.

Levario, H. K. (2007). *Design for Water: Rainwater Harvesting, Stormwater Catchment, and Alternate Water Reuse*. New Society Publishers, Canada.

Luisi, P.L., A. Lazcano, and F. Varela (1996). *What is Life? Defining life and the transition to life*, in M. Rizzotti, ed., *Defining Life: the Central Problem in Theoretical Biology*. Padua: University of Padova, pp. 149-65.

Magnoli, G., Bonanni, L., and Khalaf, R. (2002). Designing a DNA for adaptive architecture: a new built environment for social sustainability. *Design and Nature*. WIT Press.

Makonin, S., Bartram, L., and Popwich, F. (2013). A Smarter Smart Home: Case Studies of Ambient Intelligence. *IEEE Pervasive Computing*. Vol. 12, Issue 1, Jan-Mar. 2013.

Maturana H., Varela, F. (1972). *Autopoiesis and Cognition: The Realization of the Living*. D. Reidel Publishing Company, Boston.

McDonough, W., Braungart, M. (2002). *Buildings like Trees, Cities Like Forests*. The Catalog of the Future, Pearson Press.

Nelson, P. (1998). *Greenhouse Operation and Management* (5th ed.). Upper Saddle River, United States: Prentice Hall.

Nguyen, T., Kim, J., Choi, D. (2009) *ISS: The Interactive Smart Home Simulator*. ICACT Conference, p. 1828-1833.

Nirupama, B. (2005). *Wireless Sensor Networks*. Boston: Artech House Inc.

Occhipinti, L., Nunnari, G. (1996). *Synthesis of a Greenhouse Climate Controller using AI-based Techniques*. In 1996 8th Mediterranean Electrotechnical Conference (Melecon) pp. 230–233.

Olariu, S., Xu, Q., Eltoweissy, M., Wadaa, A., Zomaya, A. Y. (2005). Protecting the Communication Structure in Sensor Networks. *International Journal of Distributed Sensor Networks*, 1: 187-203.

Paladina, L., Paone M., Iellamo, G., Puliafito, A. (2007). *Self-Organizing Maps for Distributed Localization in Wireless Sensor Networks*. IEEE Conference.

Parunak, H., Brueckner, S. (2011). *Software Engineering for Self-Organizing Systems*. Vector Research Center, Jacobs Engineering Group.

Pask, G. (1969). The Architectural Relevance of Cybernetics. *Architectural Design*, September, 1969.

Ponce, P., Molina, A., Cepeda, P., Lugo, E. (2015). *Greenhouse Design and Control*. Taylor & Francis Group, London, UK.

Prigogine, I., Nicolis, G. (1989). *Exploring Complexity: An Introduction*. W. H.

- Freeman and Company, New York.
- Prigogine, I., Stengers, I. (1984) *Order out of Chaos*. Bantam Books, New York.
- Pucheta, J., Patino, H., Fullana, R., Schugurensky, C., & Kuchen, B. (2006). A Neuro-Dynamic Programming-Based Optimal Controller for Tomato Seedling Growth in Greenhouse Systems. *Neural Processing Letters*, 25, 241–260.
- Roaf, S., Fuentes, M., Thomas S. (2007). *Ecohouse*. Elsevier Ltd, Oxford, UK.
- Sethi, V. (2009). *On the Selection of Shape and Orientation of a Greenhouse: Thermal Modeling and Experimental Validation*. *Solar Energy*, 83, 21–38.
- Shihua, L., Shiyan, L., & Limei, J. (2008). Application of Adaptive Fuzzy Controller in Intelligent Greenhouse Control System. In *2008 IEEE international conference on automation and logistics (ICAL)* (pp. 1708–1712).
- Simpson, R., Schreckenghost, D., LoPresti, E., Kirsch, N. (2006). Plans and Planning in Smart Homes. Published in J.C. Augusto and C.D. Nugent (Eds.): *Designing Smart Homes*, LNAI 4008, pp.71-84, 2006. Springer-Verlag Berlin Heidelberg 2006.
- Sudeikat, J., Braubach, L., Pokahr, A., Renz, W., Lamersdorf, W. (2009). *Systematically Engineering Self-Organizing Systems: The SodekoVS Approach*. Electronic Communications of the EASST, Vol. 17. University of Hamburg, Germany.
- Sundarasaradula, D. & Hasan, H. M. (2004). A Unified Open Systems Model for Explaining Organisational Change. In S. Gregor & D. Hart (Eds.), *Information Systems Foundations: Constructing and Criticising* (pp. 125-142). Canberra, ACT, Australia: Australian National University Press.
- Tiwari, G. (2003). *Greenhouse Technology for Controlled Environment*. Pangbourne, England: Alpha Science International Ltd.
- Todd, N. and J. (1984). *Bioshelters, Ocean Arks, City Farming: Ecology as the Basis of Design*. Sierra Club Books, San Fransisco.
- Todd, N. and J. (1994). *From Eco-Cities to Living Machines: Principles of Ecological Design*. North Atlantic Books, Berkeley, California.
- Vale, B.: Vale, R. (1975) *The Autonomous House: Design and Planning for Self-Sufficiency*. Thames and Hudson; London, UK.
- Vincent, J. F. V., Bogatyreva, O. A., Bogatyrev, N. R., Bowyer, A. & Pahal, A.-K. (2006). Biomimetics - its practice and theory. *Journal of the Royal Society Interface*, April.
- Vincent, J. & Mann, D. (2002). *Systematic Technology Transfer From Biology to Engineering*.

The Royal Society, 360, 159-173.

Weiser, M. (1991). The Computer of the 21st Century. *Scientific American Special Issue on Communications, Computers and Networks*, September, 1991.

Wildberger, A. M. (1997). Complex Adaptive Systems: Concepts and Power Industry Applications. *IEEE Control Systems Magazine*, December, Vol. 17, Issue 6.

Yousefi, M., Hasanzadeh, S., Mirinejad, H., & Ghasemian, M. (2010). A Hybrid Neuro-Fuzzy Approach for Greenhouse Climate Modeling. In 2010, *5th IEEE International Conference of Intelligent Systems*, (pp. 212–217).

APPENDIX

It is now necessary to provide as detailed a survey as possible of the technological components of the Complex Adaptive House. Many materials can be substituted for others such as instead of common wood-frame construction, one might choose to incorporate straw bale walls, or the rammed-earth construction technique of Mike Reynolds's Earthships. For the purposes of this exercise, concrete slabs and steel frame construction are used to help defend against severe environmental events such earthquakes and tornadoes.

1. Energy Generation

A key feature of complex adaptive systems is their resiliency and robustness due to their aptitude for redundancy. Therefore the Complex Adaptive House should have several methods for acquiring electricity so that if there is little sun such as in the winter months, power generation is not completely dependent on solar panels.

First we must ask how much electricity we need. The problem is that every house is different which makes determining an average difficult. Since the complex adaptive house is off-grid, it makes sense to have more than enough rather than less. Excess power can always be stored in a battery bank. According to Statistics Canada, in 2007 households spent on average \$1147 on electricity and \$610 on natural gas. They report 130 GJ for a four-person household (36,111.11 kilowatts) (0.90 GJ/m², or 250 kilowatts/m²) the U.S. averages between 550 to 1200 kilowatt hours per month or 18.3 to 40 kilowatt hours/day (www.eia.gov). 40 kWh/day is average and can be higher if heat is electric (Kemp, 2005, p.241). Aside from government statistics, online forums can be a useful resource to determine how much electricity various families consume. Here are some examples:

“There are two of us in a two-story three-bedroom all-electric house (electric heat, hot water, appliances, plus I work at home so my office equipment is on all day). Our average summer consumption is about 18 kWh per day; in winter it ranges from 60 to 120 kWh/day.”

“My wife and I, along with our 5 kids, 2 dogs and a pig, live in a 5 bedroom house in the country. Everything is electric. One of our children is home schooled so there is always at least one person in the house at all times. We replaced every bulb in the house with LED lighting, just replaced the refrigerator with an energy efficient one, replaced half of our windows with newer more efficient ones, nag the kids constantly to turn off the lights and our electric bill is still SKY HIGH. We average between 50 to 70 kWh per day in the spring through fall and then between 100 and 140 per day in the winter. Before you ask, we don't overuse the heat/ac. In the summer our house is usually warm (73) and in the winter it's slightly cold (68). Blankets and sweatshirts are the normal winter attire in the house.”

“We are 2 adults + 2 children living in a 3 stories semi-detached house in Montreal (Canada). Our house is about 10 years old. We use about 60 kWh per day (from a low 36 kWh/day in summer to a high 105 kWh/day in winter). This includes electric heating, electric appliances, hot water, and A/C. According to my public electric utility, this is considered a low consumption pattern for our family profile.”

“As an electrical engineer for a power company I see on average a house that is 1600-2000 square feet and having electric appliances except for the furnace use an average of 1500 kWatt-hours per month making 50 kWh per day and a total of 18,000 kWh per year.”

Despite the 40 kWh/day average, energy usage can be much higher or lower depending on a variety of variables such as temperature/climate, habits, number of gadgets, and the energy efficiency of the house in question. Furthermore, most houses are not all electric.

1.1. Solar Panels

The most prevalent question is how many solar panels are needed despite all the other energy-saving devices?

Figure your daily usage. If you use 1200 kWh/month, that's $1200 \text{ kWh} / 30.4 \text{ days per month} = 39.47 \text{ kWh/day}$.

Figure how many kWh you need to generate each hour. You'll get 5 to 6 hours of peak sun per day on average. We'll call that 5 hours/day to be conservative. Since you need to generate 39.47 kWh/day, you need a $39.47 \text{ kWh} / 5 \text{ hours} = 7.89 \text{ kWp}$ system.

Add a safety margin. Multiply by 1.25 to account for shading, dust, and bad feng shui. So $5.92 \text{ kW} \times 1.25 = 9.87 \text{ kWp}$ system. Each 1 kWp of system takes about 100 square feet (10' x 10') of roof space, or the solar panels can be mounted in the yard.

(<http://michaelbluejay.com/electricity/solar.html>). Kemp (2005) also notes that a typical PV array rated at 1000 Watts of output power, over a five-hour period of sunlight equates to 5 kilowatt hours of energy (p.251).

Obviously it is unrealistic to have one thousand square feet of solar panels. Luckily, this is only one of several strategies that must be employed to generate all electrical needs, so the amount of PV square footage will get reduced. Before detailing other energy sources, it is worth noting that mounting them on a solar tracker increases summer production by up to 50%, and winter production between 10-20% (Kemp, 2005, p. 251).

1.2. Wind Turbines

Another strategy for collecting renewable energy is by installing a wind turbine for additional energy, which is linked to a utility tie-in inverter, a battery bank, a utility switch box, a battery system switch box, a utility meter, a service panel, which is then linked to a computer system (Kemp, 2005, p. 149). One case study said their 1500 Watt turbine provides 15% of total energy (and that is not in an especially windy area) (Kemp, 2005, p. 149). A 2 kW wind turbine x 5 hours of operation = 10,000 Watt-hour production. There is also the larger 10 kW turbine with a blade diameter of 23 feet, which will by itself power a large house (Kemp, 2005, p.257). The higher up they are installed, the greater the power they produce. Typical tower heights range from 60 to 120 feet. “At 8 mph wind speed, the Bergey Excel-S power output increases from 330 kWh to 550 kWh when the tower height is increased from 60 feet to 120 feet” (Kemp, 2005, p. 266).

Employing a *diversion charge controller* diverts excess energy to an auxiliary air or water heater unit according to battery voltage or state of charge. This is because whereas a PV array can be disconnected at will, wind turbines must have an electrical load connected at all times. Removal of the load during operation can cause damage (turbine can accelerate too much). This means that the house must know when to switch the energy from the turbine to the battery banks, or to directly supply electricity to the house, or to dump it into heating water or air. It is important to note that wind maps can be obtained from the government, and a wind survey should be done for several months at the site. Wind atlases are available at:

<http://rredc.nrel.gov/wind/pubs/atlas/> for the United States, and www.windatlas.ca/en/index.php for Canada.

1.3. Micro Hydro Systems

Micro hydro turbines can be installed only if the building site in question has a stream, river or waterfall close by. Units under 1 kW capacity are less expensive than solar or wind turbines and run 24 hours a day. Whereas a PV array of 1000 Watts over a five-hour period generates 5 kW of power, a 200 Watt hydro turbine operating 24 hours produces the same amount of energy (Kemp, 2005, p. 289).

1.4. Battery Banks

Whatever the configuration of energy harvesting systems, a battery bank will be required to store the energy to be used when needed.

Batteries store energy at 12, 24, or 48 volts, which must be converted to a typical 120/240 Vac (alternating current) through an inverter. One type of battery for a house system is the deep-cycle lead-acid industrial batteries manufactured with liquid or gelled 'maintenance-free' electrolyte. They should last between 15-20 years and are recyclable, and many companies offer trade-in discounts. A *multifunction meter* operates much like a car gas gauge. These meters measure battery voltage, energy capacity in amp-hours and a measurement of the current that is flowing into (charge) or out of (discharge) the battery. The meter can also measure the number of days since the batteries were fully charged, the total number of amp-hours charged/discharged since installation, when it is time to recharge the battery, when the voltage is too low for operation, when the battery is charging, and when the battery is fully charged (Kemp, 2005).

The voltage from the PV array or turbine is fed through a *series-connected charge controller* before going to the battery bank. The charge controller monitors the charge levels and will regulate whether the batteries get charged or not. A battery that is repeatedly drained completely

will only last maybe 300 cycles whereas if the same battery is cycled to only 25% depth of discharge, it will last 1500 cycles. A good level of depth discharge to shoot for is a maximum of 50% with typical levels of between 20% and 30% (Kemp, 2005, p. 317). Battery banks should be stored at room temperature and ventilated. Once the inverter converts the energy into the standard AC 120/240 volts, it is fed through the protective circuit breaker and into the power panel, which distributes it throughout the house. Big systems like a wind turbine and PV arrays with a hefty electric load will require a second inverter. For off-grid scenarios, either a sine wave or modified square wave inverter can be used. The latter are less expensive.

Elon Musk, the owner and founder of the Tesla Motor Company and Space X, has just announced the launch of their new home battery called the Powerwall. These batteries are lithium-ion rechargeable and can be stacked together (up to nine), and store up to 10 kW apiece.

A CBC article was posted on September 1, 2014 entitled *Vanadium battery technology could transform power grids*. The highlights of this article are that vanadium, which is element 23 on the periodic table between titanium and chromium, is beginning to be used in batteries. With a lithium battery, which has different elements on both sides of the battery, the electrodes degrade a little with every charge. A vanadium battery uses the same element on both sides, which apparently means it could last forever. These batteries also have a response time of under half a millisecond for a 100% load change.

Second generation vanadium redox batteries (vanadium/bromine) are said to double the energy density. With the advent of third generation batteries, it may be quadrupled.

“The extremely large capacities possible from vanadium redox batteries make them well suited to use in large power storage applications such as helping to average out the production of

highly variable generation sources such as wind or solar power, helping generators cope with large surges in demand or leveling out supply/demand at a transmission constrained region. The limited self-discharge characteristics of vanadium redox batteries make them useful in applications where the batteries must be stored for long periods of time with little maintenance while maintaining a ready state.”

2.0. Space Heating

Another way to reduce the amount of solar panels needed is employ many other energy-saving devices throughout the house. According to Natural Resources Canada, the bulk of energy usage was for space heating due to Canada’s cold winter climate.

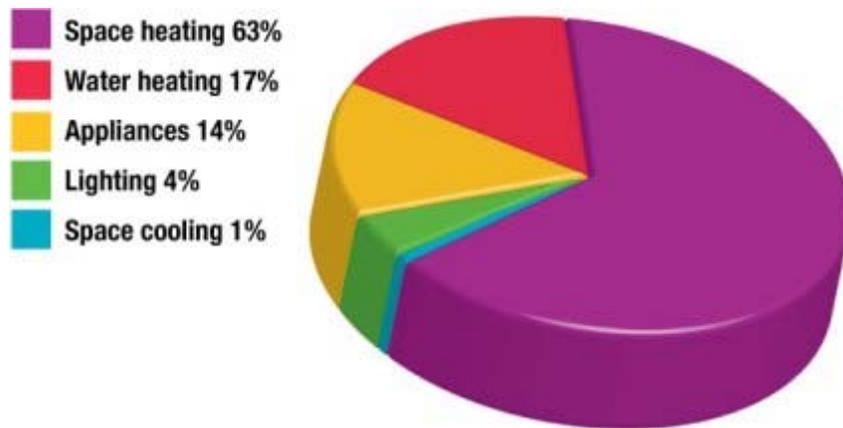


Figure 27: Pie chart of home energy usage.*

<http://oee.rncan.gc.ca/publications/statistics/trends11/chapter3.cfm>

The U.S. Department of Energy provides a chart in which space heating is still the majority of energy usage, albeit less than in Canada.

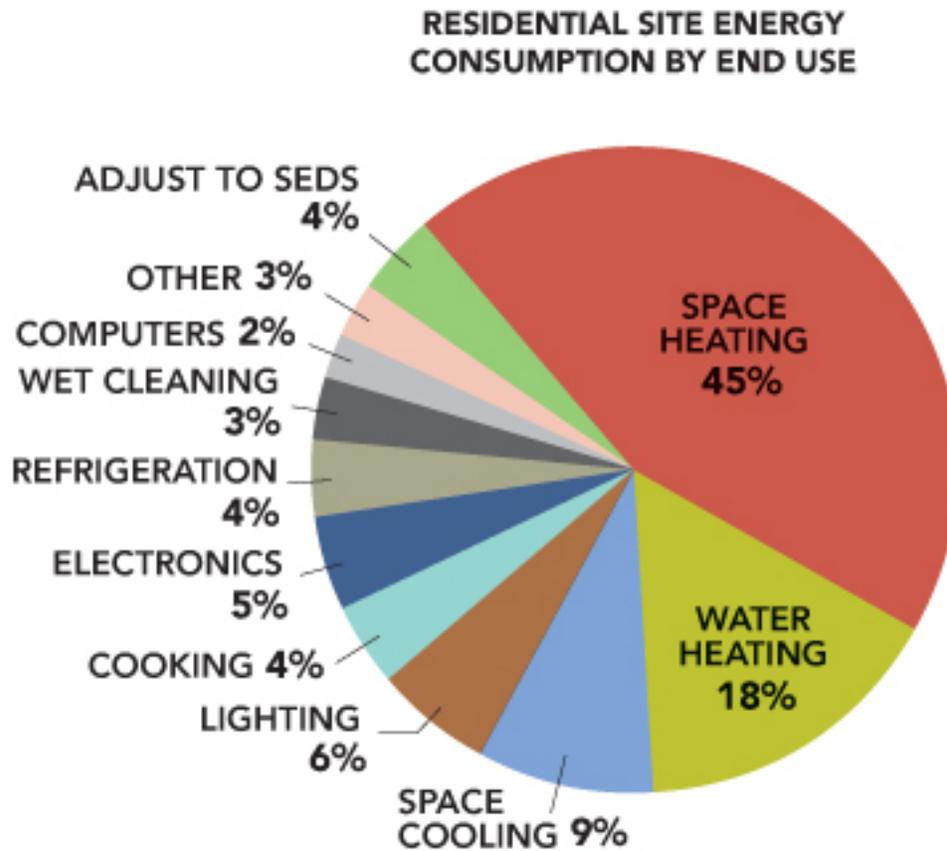


Figure 28: Pie Chart U.S. <http://buildingsdatabook.eren.doe.gov/ChapterIntro2.aspx>

Since space heating accounts for 63% of home energy usage in Canada and 45% (on average) in the United States, the next step is to investigate off-grid space heating methods.

2.1. Geothermal Systems

A geothermal system is made of five elements: the heat pump, the loop of pipe, the heat-exchange liquid in the loop (mixture of water and antifreeze), the pipe or ductwork within the house, and the electricity for the various pumps. This involves boring a hole between 130 to 500 feet down. Roughly six feet under the surface of the ground, the temperature of the soil remains constant all year round. Once the heat is extracted from the loop, it can be used for space heating and cooling, water heating, and radiant under-floor heating. Warm water from the loop is passed through a heat pump where it is raised to about 50 degrees C.

Geothermal systems are highly efficient; whereas coal has an efficiency of 70-90%, geothermal is around 400% efficient. The heat pump uses electricity to operate a compressor that extracts roughly two-thirds of a home's heating requirements. "For every kWh of electricity supplied to the heat pump, between 2.8 and 6.7 kWh of energy are supplied to the home" (Kemp, 2005, p. 232). According to the U.S. Environmental Protection Agency and Natural Resources Canada, "They are the most energy-efficient, environmentally clean, and cost-effective space conditioning systems available" (Kemp, 2005, p. 232).

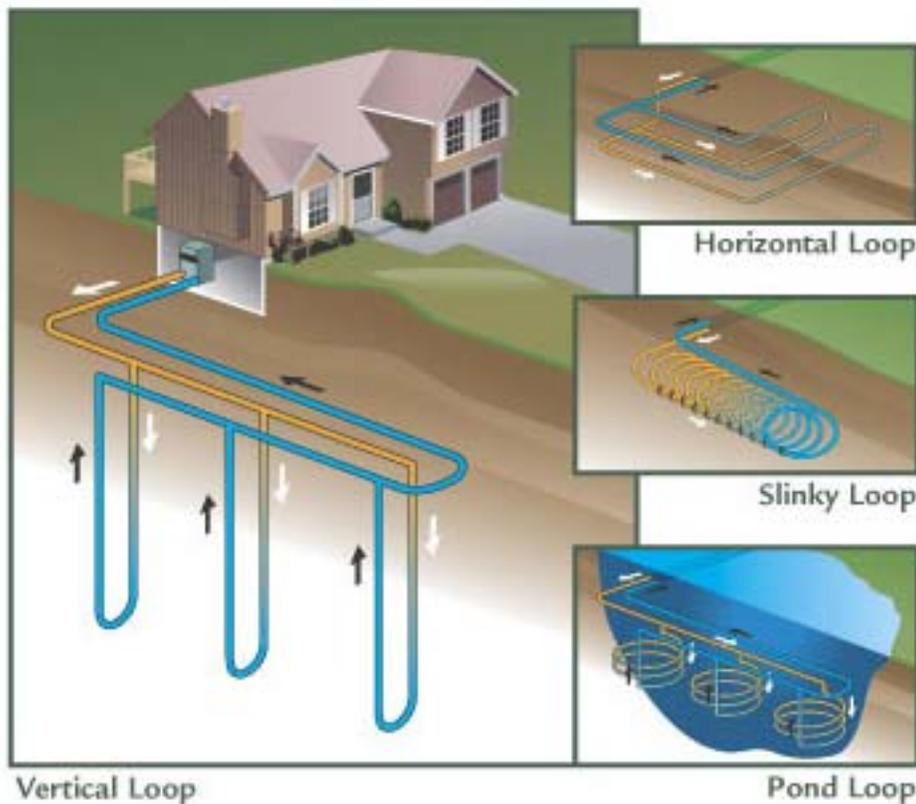


Figure 29: www.morrisheatingandair.com/Geothermal.aspx

Most sources say that geothermal can account for up to 80% of home heating, therefore some other tactics must be used to make up the difference. One interesting invention is the Solarwall® system courtesy of Conserval Engineering Ltd. It is basically an exterior wall-

cladding which is meant to be mounted on the East, West and South walls of the house which features tiny perforations which allow air to flow through the panel. “During the day, as outside air passes through the panel and along its inside surface, it absorbs the sun’s energy, warming the air which in turn rises. The preheated warm air is then drawn into the building’s ventilation system, reducing the load on the heating system” (Kemp, 2005, p. 167). During sunny days the cladding preheats air by between 17 and 28 degrees C, and on cloudy days it is reduced to 25% of what it would heat on a sunny day. Snow can reflect solar radiation up to 70% which increases its performance. The system can do the reverse in the summer by keeping the interior wall shaded and cool. Dampers allow it to switch from winter to summer positions (Kemp, 2005, p.168).

2.2. Passive Space Heating

Aside from active techniques to heat a house, there are the obvious passive methods. A combination of solar orientation, thermal mass, good quality windows, and proper insulation will drastically reduce the amount of energy needed to heat a house. Solar orientation simply means that the back of the house should be elongated and facing South. In addition, deciduous trees can be planted between the path of the summer sun and the house. In the summer they provide shade, and in the winter they lose their leaves and allow the sunlight through. Evergreen trees can be planted to the North and East to provide a wind buffer in the winter.

All windows should be triple-paned with argon or krypton gas and low-e coating except for the south-facing windows where heat gain is desired in winter for thermal mass. Wood or vinyl framing should be used, never metal (thermal bridging). The South face of the house should be mostly floor to ceiling glass to maximize the penetration of the low winter sun, while some kind of awning or upper deck should shield the windows from the summer sun. Sensors and actuators

allow the computer to open and close them as needed in the summer, along with a manual option.

2.3. Thermal Mass

The point of having large amounts of glass on the southern façade of the house is so that the sun can warm the thermal mass in the immediate interior during the colder months. This is basically free energy. The sun can provide roughly 1000W per square meter of window area (Kemp, 2005, p. 166). Thermal mass is a material's resistance to change temperature. When the floor is made of polished concrete, or stone, or certain tiles, heat from the sun is retained in the material and is released later when temperatures cool down. This helps keep the indoor temperatures constant, and reduces the heating load on the active heating systems. In order for the floor material to release the thermal energy that it has collected all day, the temperature of the house needs to cool down, otherwise the stored heat will remain in the floor. It also helps if the floor is a dark colour since dark colours attract light and light colours reflect it. A complex adaptive house can monitor the temperature of the floor and can lower the temperature accordingly to take advantage of this free energy. As a side note, water has the highest capacity for thermal mass, with concrete and stone being not too far behind.

The concrete floor slab must not be too thin because the heat will be released too quickly, and not too thick or the heat will be released too slowly. A study in New Zealand of 28 houses, where the predominant method of building is timber frame, found that the simple addition of a concrete floor slab in place of the timber floor reduced space heating by 40% (Henderson, 1993). According to Roaf et. al., (2007), a 10 cm thick concrete floor slab can be considered an appropriate thickness for thermal mass. They also state that "Today's passive solar heating systems can typically provide 30-70% of residential heating requirements, depending upon the

size of the passive solar system, the level of energy conservation being employed in the building envelope, and the local climate” (Roaf et. al., 2007, p.191). Kwok and Grondzik (2011) state that the general rule of thumb about a concrete slab is that it should be four to six inches in thickness and roughly three times the area of the windows. The Sustainable Buildings Industry Council (SBIC) says that the thermal mass should be six times the area of the windows, and that the floor slab should be four inches. They also state that mass thicker than four inches will not release heat fast enough to be effective. Four inches of thickness for the concrete slab seems to be the consensus. In order to perform the most accurate calculations possible for this exercise, a passive solar thermal performance calculator can be utilized for free at:

http://www.borstengineeringconstruction.com/Passive_Solar_Thermal_Mass_Performance_Calculator.html.

Trombe walls can be very effective containers of thermal mass. This is basically an interior wall which is positioned so that the winter sun can warm it. Again, they have no moving parts (unless the designer wants them to), and they gather free energy. Controllable vents can be implemented to allow convection to take place and to circulate the heat if desired.

A greenhouse can serve the double function of growing food but also as an area that can gather and store thermal mass. This gives the designer the option of whether to share some of that heat with the rest of the house or not.

Before moving on, it is worth noting that a backup wood-burning fireplace can be included as well. Paper log makers can be purchased to make bricks from compressed newspapers, woodchips and sawdust. Advanced combustion units are extremely efficient as they allow

oxygen into the smoke and then give the smoke a secondary burn before it reaches the chimney, which gives extra heat and no creosote (Kemp, 2005, p. 212).

3.0. Ventilation

Cross ventilation is another energy-efficient strategy which takes advantage of the solar chimney effect. This is simply a way to exhaust hot air in the summer by channeling heat upwards and ejecting it while taking in cool air from the north side at the bottom of the house. All the computer needs to do is open and close a couple of windows or vents.

As far as ventilation is concerned, a high-efficiency air-to-air exchanger or heat recovery ventilator (HRV) draws stale air into the unit and passes over a membrane before being exhausted from the house. At the same time, cold fresh air from outside comes in to the unit and passes over the opposite side of the membrane causing waste heat to be transferred to the incoming cold air, which is then warmed and distributed in the central furnace air duct. HRVs automatically adjust to current conditions. They also filter dust and pollen. The disadvantage is that they are energy consuming; a typical unit operating 12 hours a day can consume 2000 Watt-hours of energy. However, the 120 Volt fans can be replaced with lower-flow, high efficiency 15 Watt computer-style fans. Two 15 Watt fans operating for 10 hours a day lowers HRV energy consumption to 300 watt-hours (Kemp, 2005, p.43). The computer can monitor and regulate this.

4.0. Insulation

All these energy-saving solutions would be useless if the house was leaking heat. A well-insulated house is one of the best energy-saving solutions. The minimum recommended insulation values are: for walls: 23R minimum, the basement walls: 13R minimum, the roof should be 40R minimum, and the floor: 30R minimum (Kemp, 2005, p.33). A vapor barrier of 6

5.0. Water

Water heating is the next biggest consumer of energy in a typical house, accounting for 18% in the U.S. and 17% in Canada. Some statistics put water heating as high as one third of total energy usage (Kemp, 2005, p.171), depending upon one's habits.

5.1. Solar Water Heating

According to Kemp (2005), "throughout the densely populated northeastern section of the United States and Canada, homeowners can achieve an average of 55% of their water heating demand from solar" (p. 171). All solar thermal systems work in conjunction with an existing hot water tank. It pre-heats the incoming cold water which drastically reduces the demand on the hot water heater, making it very energy efficient.

A solar thermal system consists of a tube array, which must be mounted at the angle of incidence (angle at which the sun hits the target directly at midday). The system consists of five primary elements: a collector, a hot water storage cylinder, a batch of controls and sensors, one or more pumps, and pipe-work. The sun heats the absorber in the collector, the heat is transferred to a fluid, a pump pushes the hot fluid to a heat exchanger in a storage tank, and the water from the storage tank is used either directly as hot water or for heating a space. Evacuated tube configurations operate at higher thermal efficiency than flat plate collectors and are better suited for colder climates.

The most common configuration of solar thermal systems are active closed-loop systems. "A drain back system features a standard hot water tank with a non-pressurized secondary water storage tank mounted on top. Thermal sensors mounted at the solar collector panel and the hot water tank are connected to an electronic device known as a *temperature differential controller* which measures the difference in temperature between the solar collector and the hot water tank.

When the collector temperature is higher than the hot water tank and below the desired or maximum water temperature of the system, the circulator pump is activated, transferring heat to the tank. When the solar collector cools at night or on a cloudy day, the differential controller will turn off the circulation pump, causing water to flow back into the secondary storage tank. As the main water heater tank is equipped with a heat exchanger, water in the secondary tank does not come into contact with the potable hot water supply” (Kemp, 2005, p. 179).

However, for extremely cold climates, the *closed-loop glycol system* is more prudent. In this configuration “a mixture of food-grade propylene glycol antifreeze and water continuously remains in the solar collectors, plumbing lines, and heat exchanger. A differential electronic controller and temperature sensing devices are connected to an alternating current circulation pump in the same manner as in the drain back system. It is also possible to replace the alternating current circulation pump with a direct current model connected to a small photovoltaic panel mounted in the vicinity of the solar thermal collectors. This configuration eliminates the need for temperature sensors and the differential electronic controller by activating the circulation pump whenever the sun shines (Kemp, 2005, p. 180-181).

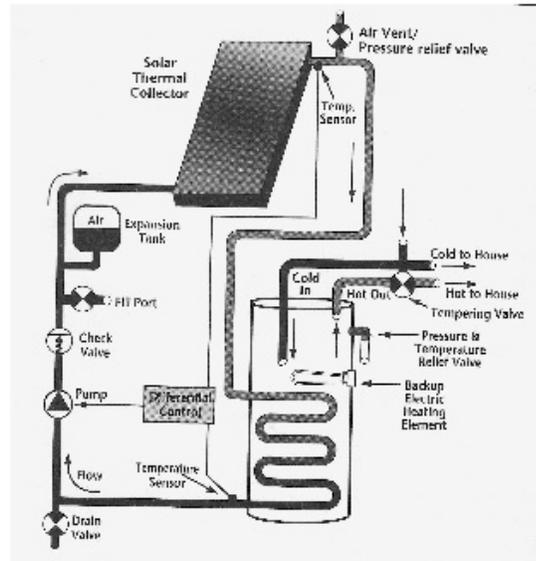


Figure 31: Solar thermal diagram

A solar thermal system requires *centrifugal-type hydronic heating pumps* to circulate a heat-collecting fluid. They have low power consumption, high reliability, and low maintenance.

5.2. Other Components of a Solar Thermal System:

“A *one-way check valve* is installed on the outlet port of the circulation pump to prevent the convection flow of warm fluid from the hot water storage tank to the cold solar collectors at night. Check valves may be spring or gravity operated” (Kemp, 2005, p.184). In AC circulator pumps, the spring operated check valve is better.

Fill and drain valves must be included. This allows the system to fill with glycol. “In order to fill and pressurize the circulation system, a positive-displacement high-pressure pump is

required to overcome the vertical ‘head pressure’ related to the force of gravity and the expansion tank air pressure” (Kemp, 2005, p. 184).

“An *expansion tank* allows the fluid in the closed-loop system to expand and contract as a function of its temperature. Without an expansion tank the buildup of pressure would cause the plumbing lines to burst when the circulation fluid is heated by the sun” (Kemp, 2005, p. 184).

“*Pressure relief valves* are designed to prevent the catastrophic explosion of plumbing lines in the event of excessively high-pressure buildup” (Kemp, 2005, p.185). This is equipped with a drain port connected to a capture bucket or drain. This emergency valve is manual, but with an actuator the computer can determine whether the valve needs to be released.

A *pressure gauge* is also required. The operating pressure is usually about 15 psi. It may also be useful in the detection of leaks, indicated by “a slow continuous drop in static pressure over time” (Kemp, 2005, p.185). In the context of the complex adaptive house, the house will monitor all gauges.

An *air relief valve* is also needed because any trapped air will lower the efficiency or stop the circulation of fluid altogether. These can be automated.

A *double-wall heat exchanger* is preferred to a single. “Heat exchangers work when the circulation fluid is at a temperature higher than that of the water contained in the storage tank” (Kemp, 2005, p.186).

“*Temperature sensors* are located on the hot fluid outlet of the solar collector and the bottom or cold heat transfer pipe of the water storage tank. These sensors, or ‘thermistors’ change their electrical resistance as a function of temperature, thereby feeding information to the *electronic*

controller module” (Kemp, 2005, p.187), otherwise known as a *differential temperature controller* (the brain of the solar thermal system). It “monitors the circulating fluid temperature at the outlet of the solar thermal collector and on the cold side of the water storage tank. When the temperature of the solar thermal collector outlet exceeds the cold side temperature of the storage tank by a predetermined number of degrees, the circulation pump is started and the heat transfer process begins. A differential temperature in the order of 5 degrees C to 14 C is required in order for adequate heat transfer to begin. Electronic controls also contain a high-temperature safety limit or cut-off circuit, which will shut the system down once the water storage tank reaches a predetermined setting (normally 77 C)” (Kemp, 2005, p.188). A third temperature gauge should be installed in the hot water storage tank. The computer can monitor and control all of this. Note: testing the antifreeze solution every two years is recommended.

On extremely sunny days, the system might produce more hot water than the inhabitants are capable of using. Therefore, a bypass valve can redirect the heat to other areas such as a baseboard hot water radiator, or can be used for radiant floor heating (glycol) in areas where hardwood flooring is employed rather than thermal mass flooring.

5.3. Power Pipe™

The Power Pipe™ is an invention which can be installed with any faucet or shower that wastes hot water down the drain (using heat recovery exchange). Basically the main drain pipe has a smaller copper pipe coiled around it. The bottom of this spiral pipe is connected to the incoming cold water supply. As the warm drain water pours down the larger vertical pipe, heat is transferred to the spiral pipe. Potable water exiting from the top of the spiral is thus pre-warmed and is sent to the hot water heater or plumbing fixtures, reducing their energy requirements.

Courtesy of *Renewability Energy Inc.* (Kemp, 2005, p.173-174).

5.4. Water Collection

Since this theoretical house is off the grid, a rain/snow catchment system is necessary to acquire water. Cisterns are an advanced rainwater collection system that involves large underground storage tanks with an arrangement of physical and UV filters, and one or more pressurized delivery outlets.

There are six main components of a system: “1) *Catchment area*: the surface upon which the rain falls, which could be a roof or impervious pavement, or grates in a patio with piping beneath. 2) *Conveyance*: channels or pipes that transport the water from the catchment area to a storage tank. 3) *Roof washing*: the systems that filter and remove contaminants and debris, which includes first flush devices. 4) *Storage*: cisterns or tanks where collected rainwater is stored. 5) *Distribution*: the system that delivers the rainwater, either by gravity or pump. 6) *Purification*: includes filtering equipment, distillation, and additives to settle, filter, and disinfect the collected rainwater” (Levario, 2007, p.14). Note: water should not be collected from surfaces containing zinc coatings, copper, or asphaltic compounds. An acceptable roofing material for systems is Raincoat 2000.

Rainwater is slightly acidic, which means it will dissolve and carry minerals into the storage system from any catchment surface. The water should be tested to determine its content and what items need to be removed by a filtration system. Sensors will be needed to know when to change filters, and to provide a warning when water levels are too low. The computer will need to know when to pump water from the roof catchment (for potable usage), and ground catchment (irrigation and greenhouse), as well as grey water. The computer must also inform the user when a part needs replacing or cleaning.

Dual potable water tanks are desirable so that if one needs cleaning or servicing, the other tank services all water needs. In terms of purification, potable water treatment systems require filters, disinfection, and buffering for pH control. Filtration can include in-line or multi-cartridge, active charcoal, reverse-osmosis, nano-filtration, mixed media, or slow sand. Disinfecting can include boiling or distilling, chemical treatments, ultraviolet lights, and/or ozonization. When water is being used from storage, sediment filtration should be a maximum of 5-microns, followed by a 0.5-micron carbon filter or an equivalent 1-micron absolute filter. These filters will remove Giardia and Cryptosporidium. Most filters have been designed to be used with municipal supplies and do not have a convenient method of monitoring when they are overloaded or need replacing (Levario, 2007, p.30-31).

Medium sized tanks are between 5000-10,000 gallons, and large tanks are 10,000 to 25,000 gallons. It is best to have more than enough, especially since the winter months give us snow which is more difficult to harvest (14 inches of snow equals 1 inch of water. As it becomes more compacted, 3 to 4 inches of snow will yield 1 inch of water) (Levario, 2007, p. 41). An excess of water should be harvested for droughts.

A water budget is used to determine the amount of rainwater that can be collected in the catchment area, and to determine if the amount will meet the user's needs. This should be analyzed as both monthly and annual figures. Monthly and annual rainfall figures should be analyzed to determine the size of the catchment area.

The first step is to multiply the catchment area (square feet) by the amount of rainfall (feet). Then multiply the efficiency percentage of the catchment surface (roofs and other smooth

surfaces are 90% efficient). Then multiply by 7.48 (number of gallons in a cubic foot). The resulting number is the amount of rainwater collected (Levario, 2007, p. 37).

Alternatively, if the catchment area is yet to be defined, then an alternate formula must be used. If the required amount of water per day is known, then multiply the gallons per day by 365 days for annual water requirement. Then multiply the annual rainfall in inches by 0.623 (which converts inches of rain into gallons per square foot. Then divide the total gallons by the gallons collected per square foot per year. This gives the catchment area required in square feet (Levario, 2007, p. 37).

(Ottawa- 36.2 inches water 22.55 = 2266 s.f. for two people. 4 ppl: 102,200 = 4532 s.f. ----40 x 80 ft house is 3200 s.f.)

So how much water does the average person need? According to Kemp (2005), the average North American uses 100 gallons or 382 liters per day (p. 89). Levario (2007) says that the average person uses 70 gallons per day. It is estimated that 40% of water is used for toilet flushing, 35% for showering, 20% for laundry and dish washing, and 5% for drinking and cooking (Kemp, 2005, p. 90). Clearly this is all a matter of lifestyle.

According to the USGS Water Science School, people use 80-100 gallons per day. A bath uses on average 36 gallons, old showers use 5 gallons per minute whereas new water-saving shower heads use 2 gallons per minute, new Energystar dishwashers use up to 6 gallons, dishwashing by hand can use 8 to 27 gallons depending on one's technique, a typical faucet flows at 1 gallon per minute, newer washing machines use 25 gallons/load, and newer low-flush toilets use 1.6 gallons per flush as opposed to the old 4 gallon flushers. Outdoor watering with a hose is roughly 2 gallons per minute (<http://water.usgs.gov/edu/qa-home-percapita.html>).

Since weather is unpredictable, and increasingly so due to climate change, alternate measures can also be investigated, such as digging a well, or if there is a lake, stream or river nearby, water can be pumped to the house. There are also other methods of collecting rainwater other than from the roof such as upside-down umbrella contraptions that funnel water into a hollow tube and into a container, as well as fog mesh.

5.5. Grey Water

If indeed 40% of potable water is flushed down the toilet, then the water requirements of the house can be reduced by 40% (as well as the catchment area) by installing a grey water system. Grey water is defined as “untreated household wastewater specifically from non-kitchen sinks, showers, baths, and washing machines or laundry tubs. It does not include water from kitchen sinks, dishwashers, or toilets; these sources fall under the heading of black water or sewer water” (Levario, 2007, p.202). It must be noted that Kwok and Grondzik (2007) state that kitchen sink water is perfectly acceptable as grey water (p. 235). Furthermore, grey water is first used for toilet flushing, and the remainder is diverted for outdoor irrigation.

The system uses a surge tank, which holds but not stores the initial surge of water. The surge tank must be emptied completely each time the grey water is sent to be treated before it is ready to be used for toilet flushing or irrigation. Treatment of grey water may include: filtering, settlement of solids, flotation and separation of lighter solids, anaerobic or aerobic digestion, chemical or UV disinfection (<http://www.level.org.nz/water/wastewater/on-site-wastewater-treatment/greywater-recycling/greywater-systems/>).

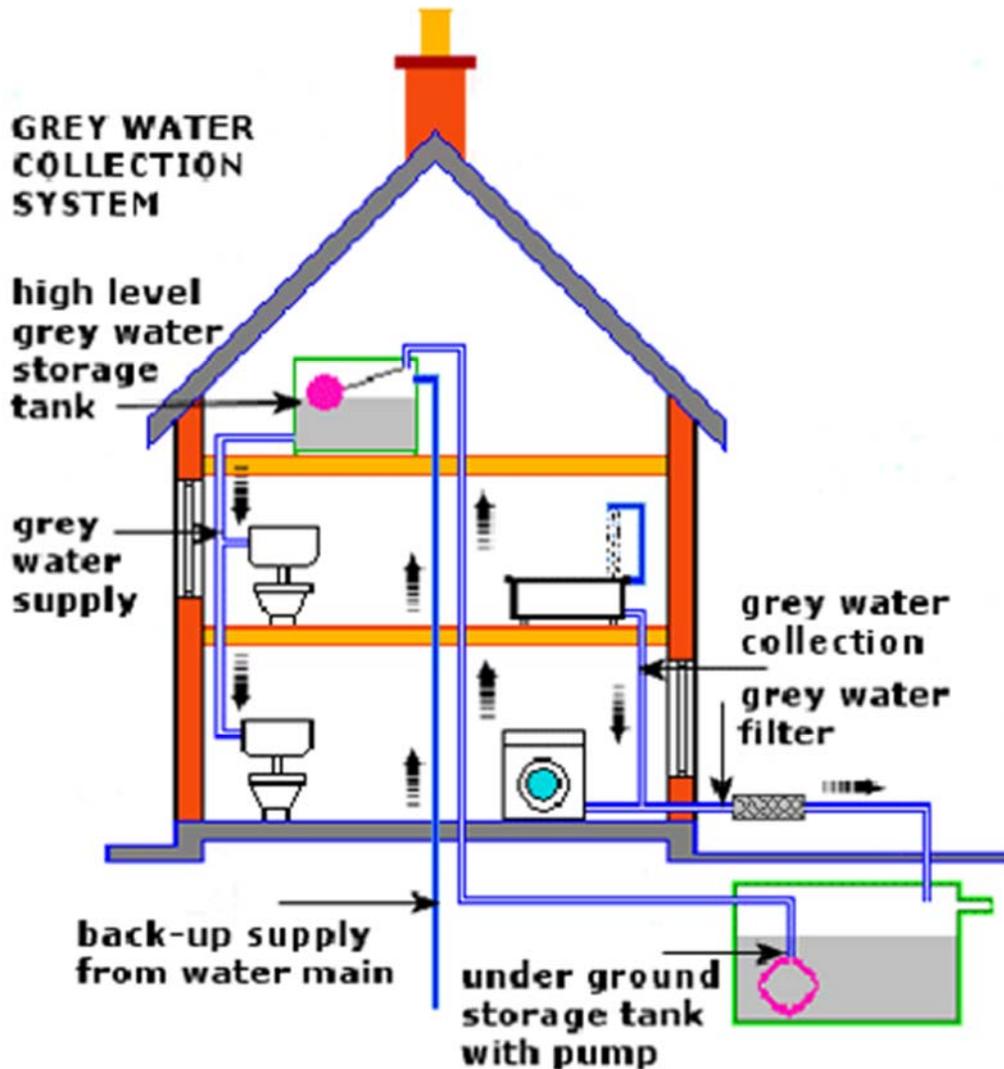


Figure 32: Diagram courtesy of <http://www.theconstructioncentre.co.uk/sustainable-homes/water/indoor-water-use.html>

5.6. Aquaculture

Aquaculture was pioneered by Jack and Nancy Todd which they developed at the New Alchemy Institute. This system is comprised of translucent cylindrical solar-algae ponds approximately five feet high and five feet in diameter, where fish such as tilapia feed on algae. They produce roughly 60 pounds of fish per year. In the summer these half grown fish are transferred to an outdoor pond where they are allowed to mature. If such a tank is located in a

greenhouse, the water soaks up the heat from the sun which acts as thermal mass. As a result of these aquaculture tanks being contained ecosystems, the fish water also serves as an excellent source for crop irrigation due to the nutrients they contain. John and Nancy Todd report from their experiments that irrigating crops with fish water increased yields by 120% (Todd, 1984, p. 67).

Aquaculture is more commonly referred to by Nancy and John Todd as ‘living machines,’ which like conventional machines, are “comprised of interrelated parts that function together in the performance of some type of work. Living machines can be designed to produce food or fuels, treat wastes, purify air, regulate climates, or even do all these things simultaneously” (Todd, 1994, p. 167). They further describe that in order to create such a machine, “organisms are collected and reassembled in unique ways depending on the purpose of the project. The parts or living components can come from almost any region of the planet and be recombined in any number of ways. Appropriate assembly is based on a knowledge of the niches and the natural history of the organisms that make up the constituent parts as well as a calculated determination of their individual fit into the unique constellation of organisms being assembled by the designer” (Todd, 1994, p. 167).

6.0. Waste

Since the complex adaptive house does not connect to any municipal sewer system, rather than direct all black water to a septic tank, a simple anaerobic digester system is preferable. Waste is directed to a heated, oxygen-free environment where anaerobic bacteria feast on the waste which converts it into fertilizer, as well as methane, carbon dioxide and other gases. Any leftover liquid can be extracted and reused as irrigation for non-food crops unless it is treated

further. The nutrient-rich fertilizer can be easily scrubbed of pathogens and returned to the greenhouse, while the methane collected can be used for heating or some other fuel source.

7.0. Greenhouses

In an ecosystem, nothing is wasted; the waste of one organism is food for another. Waste equals food. This concept will be incorporated into the Complex Adaptive House by way of anaerobic digestion. With this in mind, by 2050 the population of the world is estimated to be 9 billion (Ponce et. al., 2015), and feeding this many people poses daunting challenges. Climate change poses further problems of unpredictability to the agriculture sector. In the near future growers will increasingly need to rely on growing crops in indoor climate-controlled environments. The automation of such environments has been the subject of research and experimentation since computers entered the mainstream in the 1970s.

7.1. Automated Greenhouse Precedents

In the late 1970s, microprocessors began to be implemented along with an on/off control strategy for their programming. This system was soon replaced by PID controllers (proportional integral derivative). “These type of controllers show less fluctuations and a faster response getting the set point but the operation range of these controllers limits their functionality because not every state can be reached, so they have to be implemented for specific purposes with limited ranges” (Ponce et. al., 2015, p. 32). The implementation of sophisticated algorithms to optimize the efficiency of automated climate controls and maintenance began in the early 90s (Nelson, 1998). Fuzzy logic began to be incorporated into programming methods. Fuzzy logic is a computing method based on ‘degrees of truth’ rather than ‘true or false’ (1 or 0) Boolean logic, which is the basis for most modern computers.

In 1996 Putter and Gouws developed a non-linear and multi-variable method to control the main parameters in an ideal greenhouse, such as temperature, relative humidity, light intensity and carbon dioxide concentration. This incorporated PIDs as well as Fuzzy logic techniques. Their experiment met with reasonable success, although in that same year Occhipinti and Nunnari (1996) used an artificial intelligence technique to model and control greenhouse variables due to the fact that non-linear dynamics associated with temperature and humidity are difficult to model with traditional techniques. They successfully demonstrated that the use of 'Multi-Input-Multi-Output (MIMO) fuzzy logic controllers performed better than traditional techniques in terms of precision, energy and robustness.

Caponetto et. al. (2000) also demonstrated the validity to this approach with a new distributed PID controller along with a fuzzy logic controller they developed in 1998. This was further optimized with the implementation of a genetic algorithm. Lafont & Balmat (2002) concluded from their study of fuzzy logic in greenhouse climate control that a decentralized structure with a hierarchical organization would be the most optimal solution.

Later in 2006, Pucheta et. al. devised a 'Neuro-Dynamic Programming-Based' (NDP) optimal controller that enabled the system to learn how to decide which solution was optimal by observing itself and how to improve upon itself through reinforcement signals. Similarly, Fourati and Chtourou (2007) used an 'Elman Neural Network' within a closed loop control using a 'multilayer Feed-Forward Neural Network,' to simulate the dynamics of a greenhouse. Their findings demonstrated that Neural Network strategies demonstrate excellent performance for the control of complex systems such as greenhouses.

Kolokotsa et. al. (2010) developed two different fuzzy logic controllers [Fuzzy P (proportional)

and Fuzzy PD (proportional-derivative)]. The system monitored indoor illumination, temperature, relative humidity, CO₂, and outside temperature. It also included actuators for heating units, motor-controlled windows and shading curtains, artificial lighting, CO₂ enrichment bottles and water fogging valves. The system successfully achieved environmental set points based on specific crop requirements. Also in 2010, Yousefi et. al. proposed a hybrid Neuro-Fuzzy approach. Cluster centers were initially implemented for the Neural Network to be trained and optimized using a self-organized feature mapping (SOFM) algorithm.

Control systems based on artificial intelligence, fuzzy logic and neural networks increase efficiency in cultivation (Ponce et. al., 2015, p. 141). Shihua, Shyan, and Limei (2008) proposed an adaptive intelligent greenhouse control system with two main tasks: fuzzy control and forecasting. It controls temperature and humidity using fuzzy control but carbon oxide control is achieved with a classic PID since it is reasonably accurate and time delay is negligible. The system also forecasts and has a graphic user interface that draws historical data, and monitors and controls temperature, relative humidity, light intensity, and carbon dioxide. In contrast, Fang et. al. (2006) proposed an interesting technique for optimizing a fuzzy control system using a genetic algorithm. Genetic algorithms can “operate in high dimensional search spaces with nonlinearities, avoiding local minima and finding the optimal values for fuzzy parameters at the current set-point (Ponce et. al., 2015, p. 142). This also reduces the energy requirements of greenhouse operations.

Researchers have been experimenting with fuzzy controllers and neural networks programming.

As to which is more effective, Ponce et. al., (2015) offer this explanation:

“If experience is needed, fuzzy systems are better for controlling systems. In terms of knowing how the response is acting depending on several variables, fuzzy controllers are well determined. However, if the number of variables increases, rules grow

exponentially, making them intractable. On the other hand, neural networks are better for making a generalization of systems (the control system does not depend on specific environmental or geographical situations) and then these controllers can be used in any place, with the condition of monitoring before using them. Actually, an algorithm is needed for modeling the behavior of the greenhouse. Then, the neural control can be effectively used. One advantage of neural networks is that the model can be adapted for the current environmental states and adjust its parameters (weights) for a better response. In comparison with fuzzy controllers where membership functions are tuned for some specific set points, neural networks can be generalized and adapt to other circumstances” (p. 171).

As these articles indicate, the automation of greenhouses has been a subject under investigation before inquiries began concerning the automation of smart houses. A greenhouse is a complex structure that would benefit from the self-organizing principles of complex adaptive systems. In terms of the Complex Adaptive House, a greenhouse can be considered a CAS within a CAS.

7.2. Greenhouse Basics

The objective of a greenhouse is to provide the most preferable micro-climate for plants to achieve their maximum possible growth. Of all the factors to consider when designing and maintaining a greenhouse, air temperature is the most critical. A ‘greenhouse effect’ is achieved by an enclosed structure with transparent covering which allows solar radiation to enter, and traps it within. “The inside air temperature of a passive greenhouse directly depends on the ambient air temperature, the solar radiation, the overall heat transfer coefficient, the covering material and the wind velocity. Total solar radiation received by a greenhouse at a particular time and location also depends on its shape as well as its orientation which ultimately determines the inside temperature” (Sethi, 2009). Colder climates present more of a challenge at maintaining warm air temperatures, so a suitable heating system(s) must supplement the greenhouse effect.

7.3. Orientation and Layout

Greenhouses located above 40 N latitude in the northern hemisphere should run lengthwise East to West due to the low angle of the sun in the winter. Below 40 N latitude, greenhouses should be oriented from North to South since the angle of the sun is higher (Ponce et. al., 2015, p. 45). Ideally, a work space should be located somewhere in the North side of the structure so that nothing obstructs light from reaching the plants. Planting beds can be in-ground or raised depending on the height of the plant in question.

7.4. Electric requirements

<i>Electrical Power</i>	<i>Greenhouse Size (m²) Requirement (kW)</i>
500	15
500–2000	24
2000–3000	36
3000–4000	48
4000–8000	96
8000–12000	145

Figure 33: Electrical requirements for different sizes of greenhouses (Ponce et. al., 2015, p. 49).

<i>System</i>	<i>Actuator</i>	<i>Unit Power (W)</i>	<i>Total (W)</i>
PassiveVentilation	3 Motors	370	1110
ActiveVentilation	2 Motor (Fan)	370	740
Shading System	1 Motor	370	370
Heater	1 Heater	2000	2000
Fog System	1 Pump	370	370
Irrigation System	1 Pump	370	370
	Total	4960	

Figure 34: Electric loads for hi-tech greenhouses, (Ponce et. al., 2015, p. 51).

7.5. Ventilation

Ventilation involves allowing fresh air to enter and expelling stale air. Ventilation is necessary for reducing temperature when needed, replenishing CO₂, and controlling relative humidity.

During hot summers, passive cooling systems may not be sufficient, and so active cooling systems can reduce excess heat problems. Two such evaporative cooling systems are fan-and pad, and fog, which absorb heat during the evaporation of water. Winters can sometimes be excessively cold and it is sometimes necessary to temper the air before it enters the plant area. Convection-tube cooling and horizontal airflow cooling are two systems to accomplish this (Ponce et. al., 2015, p. 203). In the winter, additional heat will most likely be needed to supplement the heat obtained from thermal mass. Strategically placed unit heaters are generally cost effective. Alternatively, heat can be pumped from the house.

Taller greenhouse structures (3 meters) benefit the plants by providing a buffer against external temperature fluctuations, improving air movement, improving transpiration and reducing disease (Ponce et. al., 2015, p. 59). The volume of air that requires heating can be reduced at night by pulling a thermal screen across, just above the plants, and heating under the screen, which also reduces the amount of heat loss through the outer cladding. For effective ventilation, the ventilation area should be roughly equal to 15% to 30% of the floor area (Ponce et. al., 2015, p. 63).

7.6. Irrigation

In an automated greenhouse, irrigation will take place according to sensor input. In order to program the system efficiently, according to Ponce et. al. (2015), three rules must be taken into account.

- 1) Use a well-drained substrate with good structure- use a substrate that drains well and is aerated.
- 2) Water thoroughly each time- the water requirements for soil substrates is 20 liter/m² of bench or 0.3 to 0.35 liters per 0.165m diameter pot.
- 3) Water just before initial moisture stress occurs- Since overwatering reduces the aeration and root development, water should be applied just before the plant enters the early symptoms of water stress. The foliar symptoms, such as texture, color, and turgidity can

be used to determine the moisture stresses, but they vary with crops. For crops that do not show any symptoms, color. Feel and weight of the substrates are used for assessment (p. 215).

Time of irrigation requires sensing devices such as a tensiometer, an evaporation simulator, and a soil conductivity meter (for moisture levels). “The length of interval can be increased during periods of low transpiration rates and decreased during periods of high evaporation and thus high transpiration. Porous soils which have high infiltration rate and low water holding capacity require small and frequent applications if losses from deep percolation are to be avoided. Soils of much better water holding capacities can be given larger amount of water less frequently. In general it can be said that shallow root crops (vegetables) are likely to be sensitive, specially trees will be less affected. It has been suggested that the minimum time of application based on providing water for the plant when it can use it and allowing for an adequate factor of safety, would be 6–10 hours per day for porous soil and 10–18 hours per day for soils of good water holding capacity. For peak periods, the duration could be increased to 20–22 hours” (Ponce et. al., 2015, p. 215).

7.7. Materials

Greenhouses have concrete foundations and floors, which is advantageous in terms of thermal mass. Metal frames are built on top that can vary from galvanized steel, low-carbon steel AISI 1010, high-strength low alloy HSLA 340, and others. Galvanized steel has the most advantages over the other metals. This frame is then sheathed with a transparent material which will determine the amount of solar radiation penetrating the structure, and the amount of heat loss at night. Glass has classically been the cladding material of choice, but its main disadvantage is its weight and high cost. Other materials have since been tried such as polyvinyl chloride (PVC), which is the most widely used due to its versatility, malleability and mechanical properties. It is

lightweight but not environmentally friendly as it is slow to degrade. Vinyl sheets are heavier than PVC and more durable but they last only as long as five years and they attract dust. 5mm thick mylar, although expensive, has all the advantages of glass and is considerably lighter. Fiberglass reinforced panels (FRPs) are durable, retain heat better than glass, and are lightweight, resistant to breakage, and last up to 15 years (Ponce et. al., 2015, p. 68). There are several other cladding materials used in a variety of greenhouses all over the world, however the best possible skin with which to wrap a greenhouse is ETFE (Ethylene tetrafluoroethylene).

ETFE is a highly durable, lightweight plastic that typically comes as pneumatically inflated pillows. This material, which was developed by Dupont 40 years ago, has proven itself the ideal choice for greenhouses and other architectural projects that require a large skylight, as it transmits light better than glass. It is featured prominently in the Eden project in Cornwall (U.K.), the Alliance arena for the 2006 World Cup of soccer, and the Water Cube at the 2008 Beijing Olympics. Dust or dirt will not stick to it, it will not weaken or become discoloured over time, and it is recyclable. The pillows are filled with low-pressure air, providing thermal insulation and structural stability against wind or snow loads, and they can range from 5 to 15 wide and 200 feet in length. It is highly transparent (90 to 95%) and allows for the passing of UV light which is responsible for photosynthesis in plants. Although it is very elastic and strong, sharp objects can tear it at which point a simple patch can be applied. The pillow system is supplied by one or more inflation units, which consist of two redundant blowers as a backup system. Air is dried as it enters the machine to avoid condensation inside. Pressure sensors monitor the pillows and maintain them between 5 to 6 PSF. However in case of high wind or snow loads, they can adapt to up to 30 PSF. They run on 110V and consume less than 1kW/h.

7.8. Micro-Climate of a Greenhouse

In order to build a successful greenhouse it is necessary to understand the ideal micro-climate of plants. The five main elements of such a climate are temperature, relative humidity, light, CO₂, and root media (Ponce et. al., 2015, p. 171).

7.8.1. Air Temperature

Air temperature affects the development of plants. Allbright et. al. (2001) found that leaf growth is increased when air temperature is raised from 24 C to 27 C. Plants should be grown in cool temperatures at night to keep their respiration rate down, and in warm temperatures during the day to enhance photosynthesis. Ponce et. al. (2015) state that “as a general rule, greenhouse crops are grown at a day temperature 3°C to 6°C higher than the night temperature on cloudy days and 8°C higher on clear days. The night temperature of greenhouse crops is generally 7°C to 21°C” (p. 172). Different species of plants often are accustomed to varying temperatures, which makes growing tropical species alongside species that are native to temperate climates rather tricky. In this case it may be efficacious to separate them into ‘zones.’

7.8.2. Relative Humidity

Relative humidity is a product of water content and temperature and is dependent on “damping temperature changes, increase or decrease of the transpiration, the growth of tissues, pollen viability for greater percentage of ovarian fertilization of flowers, and in the development of diseases and pests (Ponce et. al., 2015, p. 174). At high temperatures, air increases its capacity for retaining water vapor which decreases relative humidity, and at low temperature, the opposite occurs. “Low relative humidity leads to reduced plant growth, presumably by causing stomata closure to conserve water. High relative humidity can reduce transpiration, limit

calcium uptake, induce physiologic disorders in some plant species, and promote fungal diseases and insect infestations” (Albright et al., 2001). Every species of plant has an ideal preference concerning humidity.

To adjust relative humidity, the employment of a fogging system is desirable. This system, aside from raising humidity, helps regulate temperature along with the ventilation system. Increasing relative humidity decreases the temperature and vice versa. This system requires a storage tank of roughly 1000 liters, a domestic water pump, a piping network to cover the growing area, and the requisite number of foggers.

7.8.3. Light Intensity

Light is directly related to photosynthesis, and thereby growth. The photosynthetic response to light is very linear. Light intensity varies by location, but it generally varies from zero in the morning to about 100 to 150 kLux (1000 lumen/m²) at noon. Light intensity on cloudy days is much lower which leads to poor photosynthetic process. Optimum light intensity for a plant is 32 kLux, but below 3.2 kLux and above 129 kLux are possibly detrimental for the plant (Tiwari, 2003). Light transmittance is higher for an East-West oriented greenhouse in the winter, and lower in the summer than a North-South orientation. Greenhouses with arched roofs transmit light better than pitched roofs (Tiwari, 2003). Since a greenhouse is an artificial method of keeping plants alive through the winter when they would normally die, the fact that they are enclosed and kept at an optimal temperature does not excuse the fact that light levels are much lower in the winter. Therefore supplementary lighting is required. There exist three categories of lamp for this purpose:

“**Incandescent** – These are generally not used because they generate excessive heat, poor radiation, and produce far-red and infrared wavelengths not desirable for plants. In terms of

energy they are very inefficient. Incandescent lights are low cost. The luminous efficacy of a typical incandescent bulb is 16 lumen/watt.

Fluorescent – They are the most common lamps used in growing rooms because they offer a variety of wavelengths that can be used for different situations. Contrasting to incandescent, fluorescent lamps are more efficient and they do not produce much heat. Fluorescent lamps offer around 40 to 60 lumen/watt.

High intensity discharge (HID) – These lamps convert electrical energy into radiation very efficiently. They are more used in Europe than America. HID lamps have a higher cost than fluorescent but need less maintenance” (Ponce et. al., 2015, p. 177).

7.8.4. Carbon Dioxide

Roughly 40% of matter within a plant is carbon and is an essential nutrient. Plants obtain carbon from the CO₂ in air. “CO₂ gas diffuses through the stomatal openings in leaves when they are open. Once inside the leaf, carbon from CO₂ moves into the cells, where, in presence of light and water, it is used to make carbohydrates (sugars) and oxygen. The process whereby CO₂ is utilized by the plant is photosynthesis and it occurs in the green chloroplasts within cells. The process is summarized in the following equations:

CO₂ + Water + Energy from Sunlight → Carbohydrate + Oxygen

3CO₂ + 6H₂O [Photosynthesis Process] Sunlight C₆H₁₂O₆ + 3O₂“ (Ponce et. al., 2015, p. 178).

“Air, on average, contains slightly more than 0.03% CO₂. The average level at the present time is 345 ppm. Due to combustion and deforestation, the level of CO₂ has been increasing 1 or 2 ppm per year since 1880, when the average level was about 294 ppm. The CO₂ level will also be higher in areas such as swamps and riverbeds, where large quantities of plant material are decomposing. Microorganisms feeding

upon plant or animal remains respire CO₂ gas, much as humans do when they use plant and animal derived foods. Plants also produce CO₂ gas through the process of respiration which is opposite to photosynthesis. A CO₂ level of 300 ppm is sufficient to support plant growth. Most plants, however, have the capacity to utilize greater concentrations of CO₂ and, in turn, attain more rapid growth” (Nelson, 1998).

According to Albright et. al., (2001), in a greenhouse, concentrations of CO₂ of three to five times ambient levels are advantageous, but higher levels than that are deleterious. Enclosed air in a greenhouse can have a CO₂ concentration of 1000ppm from respired CO₂ trapped overnight. As the sun rises, photosynthesis begins which quickly depletes the CO₂ supply, and levels can dip below 300 ppm before noon. Plants can become CO₂ deficient if they do not receive an additional supply. CO₂ enrichment is not performed during ventilation or in the summer. Typically this is done between October and March (Ponce et. al., 2015, p. 180). Three enrichment methods are:

“**Combustion** – A hydrocarbon such as natural gas, paraffin oil or kerosene, when burnt in the presence of sufficient oxygen, the CO₂ and H₂O (water) is produced as a result.

Liquid CO₂ – When dry CO₂ is filled in bottles and tanks under high pressure, it liquifies. The CO₂ gas from these pressurized tanks is released with the help of a set of regulating valves so that gas at low pressure is spread inside the greenhouse.

Solid CO₂ – CO₂ under low pressure and low temperature gets solidified and it is popularly known as “dry ice”. It can be practiced for enrichment of CO₂ in greenhouse system if the quantity to maintain a particular level, is known. Air circulation system will be required for even distribution of CO₂. The amount of CO₂ uptake by leaves

depends upon several factors:

- Plant species and variety.
- Temperature.
- Radiation intensity.
- Wind velocity.
- Water stress.
- CO₂ concentration in air.
- Resistance to CO₂ diffusion through the stomates.
- Previous history of plant.
- Leaf area” (Ponce et. al., 2015, p. 180).

7.8.5. Root Medium

Soil has two basic purposes: it acts as a reservoir to retain nutrients and water, and it provides physical support for the plant through its root system (Ponce et. al., 2015, p. 180). Soil-based growing is much less expensive than hydroponics, however soils can get depleted of their nutrients over time and are susceptible to pathogenic organisms. Hydroponics can produce better yields, but for the purposes of this exercise it will not be discussed here.

7.9. Automation Controls

Automation controls in greenhouses consist of lights, fans, pumps, heaters and mist systems which must be turned on and off at specific times. This requires clocks, photocells, thermostats, and humidistats. Naturally a thermostat can turn on a heater when the temperature drops, and can open vents and turn on fans. Humidistats regulate humidifiers. The automation controls should regulate ventilation, humidity, irrigation and nutrient distribution with the help of

sensors, actuators and a controller. Maintaining a greenhouse is not without complexity due to the fact that plants differ in the amount of water and light they require, as well as nutrients, therefore a system of automation is preferential. According to Tiwari (2003), some of the advantages of hi-tech greenhouses are: “1) the computer always knows about the functioning of individual system, and if programmed properly can coordinate this system to provide the optimum environment. 2) The computer can record environmental data which can be displayed, stored, processed to provide history of cropping period. 3) A high-speed computer can control several remotely located greenhouses. By placing the computer in a central area, the results can be monitored frequently. 4) With proper sensing apparatus and programming, the computer can anticipate weather change in advance and make adjustments in heating and ventilation systems, thus saving energy. 5) The computer can be programmed to sound an alarm if conditions become unacceptable and warn of sensor and equipment failure, if not managed properly” (Tiwari, 2003).

For the automation of a greenhouse, the controller mechanism must be linked to a real-time operating system (RTOS), since different parts of the greenhouse have varying timing requirements. Artificial neural networks controllers have been tested in different greenhouse conditions. Some examples provided by Ponce et. al. (2015) are: Albus’s cerebellar model articulation, which is a neural network based on the function of the cerebellum, Kawato et. al.’s hierarchical neural network controller, and Psaltis et. al.’s multilayered neural network controller which is a feed-forward neural network and runs in real-time.

8.0. Sensor Monitoring

Sensors convert physical variables to an electric signal for processing, control through actuators, or display. Optimal crop growth in an automated system depends on gathering and disseminating the right information to make the right decisions. Various sensors and actuators should be present both inside and outside the greenhouse. The following figure shows a list of possible sensors. Sensors should be situated logically according to true conditions. For instance, temperature, relative humidity, and light sensors should be placed just above the crop canopy, not in shadowed areas or near vents. EC and pH sensors must be submerged with the nutrient solution. Outside sensors, or a weather station, should be on top of the greenhouse (Ponce et. al., 2015, p. 187). Actuators consist mainly of ventilation and cooling technologies, heating, and irrigation, composed usually of electrical motors or pumps.

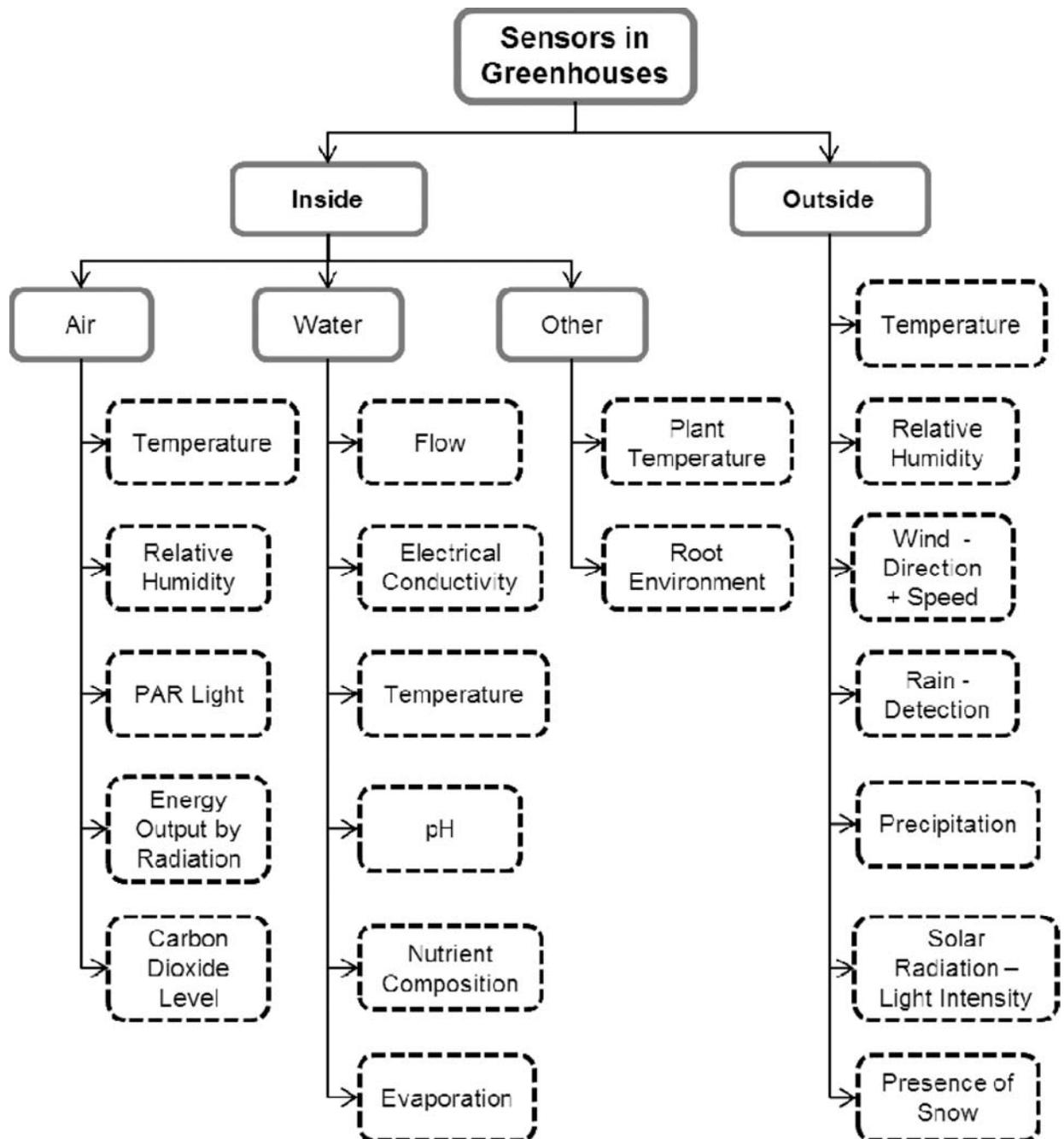


Figure 35: Sensors in greenhouses (Ponce et. al., 2015, p. 183).

8.1. Microprocessors

Dedicated microprocessors are basically simple computers which typically have a keypad and a few lines of liquid crystal display for programming. They have many output connections and can control up to 20 devices which makes them an economical choice. They can receive all

manner of signals such as temperature, light intensity, rain and wind speed, and they can be integrated with a wide range of other devices which is not possible with a simple thermostat. From the wind direction or speed, a microprocessor can operate ventilators, or it can be set to activate the CO₂ generator when the light intensity exceeds a given set point. Microprocessors are much more accurate than other methods because they employ a thermistor rather than a thermostat (Ponce et. al., 2015, p. 233).

8.2. ANFIS

Although there are many types of control systems, one example is given here due to the fact that it seems to be the closest fit for the complex adaptive greenhouse. Adaptive neural fuzzy inference system (ANFIS) is a graphical network representation of T-S fuzzy systems. This model combines the benefits from an artificial neural network with that of a fuzzy inference system. This system has gained in popularity because it is a fast and accurate learner and has a great capacity for data management.

“The main objective of ANFIS is to optimize fuzzy system’s parameters (to obtain an accurate answer to a problem) through a learning algorithm implementation and a set of inputs and outputs which are responsible of the learning process. Those sets are used to build a fuzzy inference system, from here the membership function parameters are adjusted by a hybrid training algorithm, which combines gradient descent and the least-square method. The least-squares method is actually the major driving force that leads to fast training, while the gradient descent serves to slowly change the underlying membership function that generates the basic functions for the least-squares method. This type of adjustment allows the fuzzy model to learn the data set that is provided. The adaptive neuro-learning works in a similar form as a neural

network. The adaptive neuro-learning model provides a procedure of fuzzy modeled to learn information from a set of data” (Ponce et. al., 2015, p. 247).

9.0. Sensor Networks: An Overview

Sensors are the intermediaries that allow computation to be derived from the environment. Each sensor is developed for a specific measurement application. Some examples of sensor specification are temperature, light, fluid or gas flow, motion, chemical, pressure, displacement, electrical resistance, electrical potential, and myriad others.

The basic functions of a sensor are as follows: first there is a stimulus from the environment which excites the sensor which leads to an input circuit which converts the signal into whatever form (usually voltage) that the following circuits require. Then, signal processing, conditioning and/or amplification is required. This may include frequency-selective filtering by means of mathematical operations such as differentiation, integration, ‘logging’ or ‘anti-logging’ depending (Carr, 1993, p. 4). For this to be useful there must be a display function with data storage, or a control function. Many types of outputs and displays are available depending on the specific situation. They are generally generated as digital signals to be stored in a computer.

Many materials change their electrical characteristics when subjected to changing environmental conditions. Sensors are made to predict these changes within a specific range. A thermistor, for example, “is a variable resistor that changes smoothly with temperature. An analog-to-digital convertor (ADC) then converts the voltage drop into a binary number that a microcontroller can store or process” (Nirupama, 2005, p.7).

Sensors and actuators have recently been revolutionized by the advent of micro-electro-mechanical systems (MEMS) technology. This fancy term simply means that mechanical

devices such as barometers, accelerometers, and movable mirrors are now miniaturized. Lithographic processes are used for etching transistors onto silicon so that very small mechanical structures can be carved “such as a microscopic springboard within an open cavity. Gravitational forces or acceleration can deflect this cantilevered mass, causing powerful internal forces that cause changes in material properties or delicate alignments, which can be amplified and digitized” (Nirupama, 2005, p.7). These MEMS devices are getting smaller and smaller which results in lower power consumption and better performance.

MEMS sensors are cheap and efficient at sensing a plethora of phenomena such as various physical forces, chemical concentrations, and environmental factors and fluctuations. They consume a couple of milliwatts and do not even require being turned on most of the time. They turn on when they need to and return to dormancy. With ADCs becoming as efficient as they are, the sensor subsystem has become similar to a processor making MEMS sensors critical components for embedded devices.

In 2003, MIT’s Technology Review magazine referred to sensor networks as “one of the ten technologies that will change the world” (www.techreview.com). Until recently the development and adoption of sensor technology has been hampered due to cost. Thanks to Moore’s law whereby computers double their processing speed and storage capacity every year or two, it is becoming increasingly cost effective to utilize sensor technology and wireless sensor networks (WSN’s). Miniaturization and manufacturing have ensured that sensors and sensor networks are becoming cheaper as well as commonplace. In a few years, “complete systems with computing, storage, communication, sensing, and energy storage could be as small as a cubic millimetre leading to very cheap, small form factor sensor devices” (Nirupama, 2005, p. 2).

Sensors can be deployed throughout any physical space to measure any type of phenomena as long as the sensor is specifically calibrated to measure a specific phenomenon. Every sensor is different to some degree in that respect. Once the sensors senses something, it can send the raw data to a computer, or as is more common now, it processes and communicates the information in coordination with other nodes. Combining this with the system technology of the internet makes it possible to equip the world with artificial senses that we can measure from afar either to better understand our world, or to facilitate our lives within it. Sensors can be thought of as ways to conduct scientific research or for the automation of infrastructure and/or its maintenance. They can monitor (a) space, they can monitor (b) things, and they can monitor (c) the interactions of things with each other and the surrounding space. The first can include such things as environmental and habitat monitoring, precision agriculture, indoor climate control, surveillance, treaty verification, and intelligent alarms. The second can include structural monitoring, ecophysiology, condition-based equipment maintenance, medical diagnostics, and urban terrain mapping. The third can include wildlife habitats, disaster management, emergency response, ubiquitous computing environments, asset tracking, healthcare, and manufacturing process flow” (Nirupama, 2005, p.2). It is maintained in certain circles that “large-scale, distributed sensor networks will eventually cover and instrument the entire planet. They will continuously monitor and collect information on diverse phenomena, including endangered species, soil and air contaminants, patients, and man-made environments” (Nirupama, 2005, p.2).

It is important to note that individual sensors in a network are resource constrained: their processing speed is limited, storage capacity, and communication bandwidth. But when combined within a network their collective ability is substantial. In most situations, the network

operates for long durations in a wireless setting which offers its own set of challenges such as energy consumption. They generally use batteries or can harvest energy from the immediate environment or both. Reducing energy consumption is an issue that will be explored. Another challenge is that networks can experience large variations in connectivity and can be subject to environmental conditions (especially if outdoors). The high degree of interactivity with other nodes implies that sometimes they may not always work in unison. One bad node can distort the overall data, but where there is a programming will, there is a programming way to compensate. Manually configuring large networks of small devices is impractical so they must be adaptive and self-configuring. Ideally, they must process the sensor data close to where it was harvested and filter out the uninteresting events from the interesting ones.

9.1. Scale

An important determinant of a wireless system design includes the spatial and temporal scale which are concerned with “the sampling intervals, the extent of overall system coverage, and the relative number of sensor nodes to input stimuli” (Nirupama, 2005, p.5).

The nature of the phenomena measured naturally dictates the scale of the sampling of the sensor network. High-frequency waves such as acoustic vibrations require higher temporal and spatial sampling than things such as temperature or light levels in a room. The sampling scale is determined by the application. For example if we were monitoring a structure to detect structural faults and seismic impacts we could use ‘coarsely-grained’ data. If we are monitoring it more closely for profiles of structural response, a ‘fine-grained’ data stream is needed which may require innovative, collaborative signal-processing techniques.

The spatial and temporal scope of systems varies from micro to macro. At the macro scale there are environmental monitoring systems such as those for tsunamis which can span for miles, or

they can be at the scale of a building or room, and at the micro scale, the surface of a piece of machinery. Either way, systems that are meant to operate over extended amounts of time and space must be self-organizing because as the system grows it becomes impossible to manually configure and control the environment.

System density is a measure of the number of sensors within the measured footprint. “High-density systems are better for exploiting redundancy to eliminate noise and extend system lifetime. A high density of nodes to stimuli enables a large number of independent measurements to eliminate channel or coupling noise. Similarly, where density is high enough to allow oversampling, nodes can go to sleep for long periods and extend coverage over time” (Nirupama, 2005, p.5).

9.2. Variability

Variability is another important characteristic of sensor systems. Whereas static systems can be optimized during the initial design phase, more variable systems must be programmed to be self-organizing which could induce limitations into their longevity and variability, whether it be system structure, task and/or mobility.

As far as system structure is concerned, monitoring systems for bridges, buildings, or airplanes must use a static structure. Conversely, if we were attempting to study bio-complexity we would require an ad hoc sensor network where sensors are scattered randomly, and possibly in remote regions over extended areas.

“Variability in system task determines the extent to which we can optimize the system for a single mode of operation. Even a structurally static system might perform different tasks over

time-for example, a structural system that periodically generates a structure profile could act as an even monitoring system” (Nirupama, 2005, p.5-6).

In most sensor networks usually the nodes are placed into a fixed position. However sometimes sensors are placed in vehicles or on people or animals which requires a different set of parameters than monitoring a static object or space.

9.3. Autonomy

Autonomy is one of the most important aspects of the success of a system. “Greater autonomy increases the need for multiple sensor modalities, and translation between external requests and internal processing and the internal computation complexity” (Nirupama, 2005, p.6).

“Truly autonomous systems depend on multiple sensor modalities for robustness. Different modalities provide noise resilience to one another and can combine to eliminate noise and identify anomalous measurements” (Nirupama, 2005, p.6).

Autonomy also requires computational complexity. Low autonomy systems simply deliver data to a human user or program. Event detection requires autonomy because there must be programming involved to differentiate between interesting and non-interesting events, and must process more detailed computations to process detailed measurements.

9.4. Embedded Devices

The most unique aspect of the devices used in WSNs is the integration of wireless communication, sensing, and computation. “A sensor network node’s hardware consists of a microprocessor, data storage, sensors, ADCs, a data transceiver, microcontrollers, and an energy source” (Nirupama, 2005, p.9). Two possible options for wireless integrated networked

sensors are a) having more computing and storage capacity onto a chip or b) putting the same capacity into a progressively smaller area with reduced energy consumption.

9.5. Operating Systems

Many operating systems have emerged with the hopes of simplifying the programming of embedded devices. At first a slew of programs emerged such as Vxworks, GeoWorks, Chorus, compact Linux variants, and Windows CE, who scaled down their footprints and added TCP/IP capabilities (Nirupama, 2005, p.11). It was determined that “to make the networked, embedded node an effective vehicle for developing algorithms and applications, modularity, in addition to efficiency, would be critical. A modular, structured runtime environment should provide the scheduling, device interface, networking, and resource-management primitives for the network. Moreover, it must support several concurrent flows of data from sensors to the network to controllers” (Nirupama, 2005, p.11). An operating system called TinyOS was designed to provide “a framework for dealing with extensive concurrency and fine-grained power management, while providing substantial modularity for robustness and application-specific optimization” (Nirupama, 2005, p.11). TinyOS also uses its own language for creating a system made from smaller components. Soon after, other programs emerged such as Mantis and Contiki which is more flexible than TinyOS.

9.6. Distributed System Architecture

The bulk of the energy consumed in WSN technology is from wireless communication.

Longevity within an autonomous system is unrealistic if every node streams sensor data to a computer, so distributed systems must be constructed “whose outputs are at a higher semantic level-compact detection, identification, tracking, pattern matching, and so forth” (Nirupama, p.

12). Two important types of architecture have emerged to support this notion: self-configuring networks and data-centric systems.

9.7. Self-Configuring Networks

If we wished to track a moving object within a distributed wireless sensor network, there are a few requirements. Firstly, the nodes have to cover the desired area, they must begin ‘bootstrapping’ by coordinating access to the network, they must be aware of their physical positions in time and space by synchronizing with each other, any faults must be detected and corrected, and they must form a connected network by establishing appropriate routes for reporting and receiving data.

“The primary objective of self-configuration is to let sensor devices autonomously coordinate among themselves to ensure adequate deployment coverage, form a connected network, establish a spatial coordinate system, synchronize themselves in time, and calibrate their sensor readings. An extremely important secondary objective is to exploit the node redundancy for energy conservation and management to achieve longer, unattended system lifetimes”

(Nirupama, 2005, p.13).

9.8. Data-Centric Systems

This type of system promotes in-network processing “by building on a data-centric instead of an address-centric architecture for the distributed system or network” (Nirupama, 2005, p.13).

Address-centric refers to the ad hoc routing protocols of the traditional internet model of sending data from one edge of a network to another. Directed diffusion separates data identity from node identity using “a simple typing mechanism and encoding of attribute-based naming with simple matching rules. Names are sets of attributes. Each attribute is a tuple, including keys, values, and operations. User-provided code or filters can be distributed to the sensor

network to perform application-specific, in-network processing tasks such as data aggregation, catching, and collaborative signal processing” (Nirupama, 2005, p.14). As soon as in-network processing is established, programming abstractions must be deployed in order for the entire sensor network to be programmed and not just each individual sensor node.

9.9. Energy Harvesting

For a sensor network to be truly autonomous, it relies on operation over extended branches of time without human intervention. Energy supply is obviously a critical factor in the design of the WSN. Unlike handheld devices such as smart phones which can be plugged in a regular intervals for recharging, sensor nodes often cannot be continuously plugged in or returned periodically to some docking station. Although miniaturization decreases energy consumption and nodes can go into sleep mode when not in use to save power, energy harvesting techniques embedded into the sensor node technology are most advantageous to ensure longevity.

However, this requires design changes which encompass the hardware of the node, power management, and energy scheduling at the level of the entire network. For instance, if the energy consumption of a node is to be matched to the environmental energy available where it is physically situated, then it must be equipped with the appropriate hardware to measure environmental input rather than simply measuring its own battery life. Also, power-management decision-making in a network is altered with energy harvesting. For example, supposing a network encompasses solar harvesting as an energy supply. At a given moment, the network may have two or more different data routes, but the harvested energy available at nodes along these routes may differ due to a shadow at a particular time of day. This is when a harvesting-aware power-management strategy, along with an environmental-energy-input measurement allows it to choose the appropriate route to avoid the shadow.

Typically an energy-harvesting node will contain, apart from the energy-harvesting component, an energy-storage mechanism such as a battery or an ultra-capacitor. This is then referred to as a harvesting network which requires further considerations than just the individual node. Such a network may be heterogeneous and the environmental energy harvested may differ as well as the availability of energy from node to node.

9.10. Methods of Harvesting Energy

Solar or other light source can be converted to electricity with commonplace solar cells. The amount of energy generated by these cells varies from $15 \mu\text{W}/\text{cm}^2$ in noon-time sun exposure to $10 \mu\text{W}/\text{cm}^2$ in indoor incandescent lighting (Nirupama, 2005, p. 121). With circuit design, a solar cell can be modeled as a voltage source with an internal resistance. The output voltage is constant (at around 0.6V for a single cell) and the supply current varies with light intensity.

There is always the advantage of adding as many solar cells as is required to achieve any desired voltage.

Vibration energy is of interest in many environments including commercial buildings, parking structures, aircrafts, trains, industrial facilities, and residential buildings wherein $300 \mu\text{W}/\text{cm}^2$ are available on average (Nirupama, 2005, p.121). Vibrations can be detected and harvested from things such as heavy machinery, appliances, HVAC, movement of people or vehicles etc. converting this energy into electricity can be classified as either electromagnetic, electrostatic, or piezoelectric.

Electromagnetic conversion uses vibrations to move a conductor in a magnetic field but the output is too low to be usable. Electrostatic conversion uses vibrations to move conductors of a charged capacitor but a separate voltage source is required to charge the capacitor.

“Piezoelectric conversion uses materials that when mechanically deformed, generate an electric

potential. This method combines the advantages of electromagnetic and electrostatic conversion but is difficult to implement at a micro scale. However, with current technology, this method has the greatest available energy density of the three methods” (Nirupama, 2005, p.121).

Although wind and water are widely used to make electricity, micro-scale technologies to extract these energies are non-existent. “Thermoelectric generation using the Seebeck effect (flow of current in a loop made of two wires of certain metals when a temperature difference is applied to the wire junctions) and other methods have been demonstrated to yield 10 to 40 $\mu\text{W}/\text{cm}^2$ using a 5°C to 10°C temperature gradient” (Nirupama, 2005, p.122). Pressure variations of a fluid or gas in an enclosed space can generate energy from a rise and fall in temperature throughout the day (through expansion and contraction). However converting this into electricity is problematic.

Since harvesting energy from the environment is highly variable, reliable system performance is a key design criteria. Another key is determining whether the energy harvested is sufficient to run the system indefinitely. Analysis must be undertaken to determine the energy usage of the devices versus the amount of harvested energy needed to run the network or each individual node.

In a distributed WSN with multiple harvesting nodes, a harvesting-aware power-management system is needed to allocate the workload to the nodes in a way that maximizes the performance of the overall system. This requires detailed knowledge of what resources are available at every node and all the tasks that the system needs to perform. Certain tasks such as routing data packets from node to node may require energy consumption at intermediate nodes. In a distributed system, scalability concerns require that the nodes not send all their information to a

central node for scheduling decisions, but be coupled among themselves. In most situations, communication itself is the energy consumer so the decision-making must be distributed, not centralized. Distributed power management is budding area of research. Suggested methods are distributed battery-powered systems “which attempt to reduce and balance energy usage across multiple nodes to ensure system-wide sustainability rather than minimize the total energy consumption, which could cause certain nodes to deplete faster and make the overall system useless” (Nirupama, 2005, p. 128). Conversely in harvesting networks, rather than balancing the energy consumption of each node, the goal is “to share the workload in proportion to the harvesting opportunity available to each node” (Nirupama, 2005, p.128). The workload distribution may be application dependent.

“One of the applications for sensor networks is to monitor a deployment scene for specific events and to report the occurrence of such events with low latency. The sensing transducers consume minimal energy and are kept active, while the power-intensive modules, the processor and the radio, are woken up when an event is triggered by a transducer. The event must then be communicated with low latency to a central base station” (Nirupama, 2005, p. 128). The point of power-management is for nodes and their individual harvesting points to adapt their energy consumption and provide an energy-efficient communication topology.

10.0. Smart Home User Interaction and Interface Precedents

Since the Complex Adaptive House is highly computerized, it is worth exploring how the human inhabitants might possibly interact with it. The house not only self-regulates its systems, but must also respond to the habits of those who reside within it. A user interface must also be established via a series of touch screen panels dispersed throughout the domicile. To get an idea of what this might look like, it is worth exploring an overview of progress being made in the

realm of smart home research so that it may be appropriated.

Many tech companies are transitioning into the smart home business, and they all offer easy to use, simple graphic user interfaces. The touch screens are mounted to the walls in various parts of the house and display the house plan with the position of all electrical devices displayed as icons. The user can touch an icon and get a real-time status report. The touch screens can also be used as intercoms.



Figure 36: Touch screen on a wall

Fig. 41: <http://ptdiyen.com/product-touch.html>

Fig. 42: <http://hiddenwires.co.uk/resourcesnews2011/news20110401-03.html>



Figure 37: Example of a simple interface

Simpson et. al (2006), present some notes, criteria and challenges for user interfaces for smart homes. Although much of the data is generic, there is the question of who should receive what information? In some cases, data should perhaps be delivered on a need-to-know basis in terms of privacy and security. How should the data be presented? The goal of the smart home will determine the nature of the interface. The user may not wish to sift through hours of sensor data and may prefer to be notified when a sensor malfunctions. In contrast, a user may wish to see how the sensor network performs after or during a perturbation event. If the user wants the home to monitor and adapt to human behaviour, then it must possess two main faculties: *plan*

recognition and knowledge representation (Simpson et. al, 2006).

Plan recognition is recognizing what a person is doing (or failing to do) based on observations from the sensor network. Since human behaviour can be complicated, this can present more of a programming challenge than monitoring for weather perturbations. Many smart home applications rely on Bayesian networks or Hidden Markov Models for this, and a probabilistic approach allows multiple plans to be calculated simultaneously.

Knowledge representation is the method of representing whatever knowledge is accumulated by the house. A number of constraints must be satisfied:

- “1. The representation must provide a computational surrogate for real-world entities, such as people, sensors, projectors, lights, and appliances.
2. The representation implies a set of ontological commitments, which define how a smart home looks at the world. For example, is camera input interpreted as people or as blobs of colour being tracked through a room?
3. The representation defines how inferences are made, and determines what kinds of inferences are possible.
4. The representation collects, organizes and links together large quantities of information” (Simpson et. al, 2006). Also, “in order for a smart home to make use of plans, the smart home must either be given the plans or given the knowledge necessary to construct the plans” (Simpson et. al, 2006). For the Complex Adaptive House, it is the latter that must be applied.