# CHAPTER 3.
# EVOLUTIONARY ARCHITECTURAL APPROACH

*This chapter undertakes the mission of exploring in details the essence of evolutionary architectural approach which is the main digital approach of this thesis within its analytical part.*



**Keywords:**

EVOLUTIONARY ARCHITECTURE, EVOLUTIONARY DESIGN, GENERATIVE GENETICS, NATURE PRINCIPLES, ALGORITHMS, EVOLOUTIONARY SYNTHESIS.

# CHAPTER THREE

## EVOLUTIONARY ARCHITECTURAL APPROACH

### CHAPTER STRUCTURE: ANALYTICAL SECTION

## 3.1. INTRODUCTION

As the modern world develops and utilizes design technology for architecture, different design methodologies have emerged. Current design research has focused on computationally mediated design process, being essentially concerned with form finding and building performance simulation through the integration of physics and algorithms. Since its emergence, design practices are increasingly aided by and dependent on the technology and have resulted to major paradigm shift. (Al Qawasmi, 2004)

This chapter introduces the concept of evolutionary architecture indicating the nature of the biological and scientific analogies and related ideas to the wider context of present scientific discourse, exploring new computer-based techniques for design which models inner logic.

In the process, it relies heavily on natural science, algorithms, genetics, biological techniques, and the newer science of cybernetics, complexity creations and implementation of tools to assist in developing the model, suggesting design interacting and evolving in harmony with natural forces, including those of ecological theories. (Hensel, Menges, 2006)

It opens new territories and radically reconfigured the relationship between design and creating evolutionary architecture. This new design process enables architects to improve design quality which makes new levels of complexity and new aesthetics possible. These lead to evolutionary design processes which evoke new architecture forms. Evolutionary design process investigates fundamental form-generating processes in architecture, paralleling a wider scientific search for a theory of morphogenesis in natural world (Frazer, 1995); (Kolarevic, 2003); (Hensel, Menges, 2006); (Littlefield, 2008); (Datta, Hanafin, Pitts, 2009)

## 3.2. THEORETICAL APPROACH

### Definition of Evolutionary Architecture:

In general, ‘evolution’ means ‘the process by which the physical characteristics of types of creatures change over time, new types of creatures develop and others disappear’. In addition to this general term, it is more specified as ‘the use of evolutionary computation for design problem’ because computer generates all evolutionary process, in evolutionary design.
**In biology**, evolution is a change in the inherited traits of a population from one generation to the next. This process causes organisms to change overtime. Inherited traits are the expression of genes that are passed on to offspring during reproduction. Mutations in genes can produce new or altered traits, resulting in the appearance of heritable differences between organisms. Such new traits also come from the transfer of genes between populations, as in migration, or between species, in horizontal gene transfer. Evolution occurs when these heritable differences become more common or rare in a population, either no randomly through natural selection or randomly through genetic drift.

In computer science, Evolutionary Computation is a subfield of Artificial Intelligence and is the general term referring to several computational techniques which are based to some degree on the evolution of biological life in the natural world. Evolutionary Computation uses interactive progress, such as growth or development in a population. This population is then selected in a guided random search using parallel processing to achieve the desired end. (ESARQ, 2008)

According to French 1994, the crucial inventions of nature can be considered as their development in the fossil record. The organization of living material in a cell with a cell wall and a nucleus

As said by John Frazer who starts his book with these words:

> *"Architecture is considered as a form of artificial life, subject, like the natural world, to principles of morphogenesis, genetic coding, replication and selection. The aim of an evolutionary architecture is to achieve in the built environment the symbiotic behavior and metabolic balance that are characteristic of the natural environment."* (Frazer, 1995)

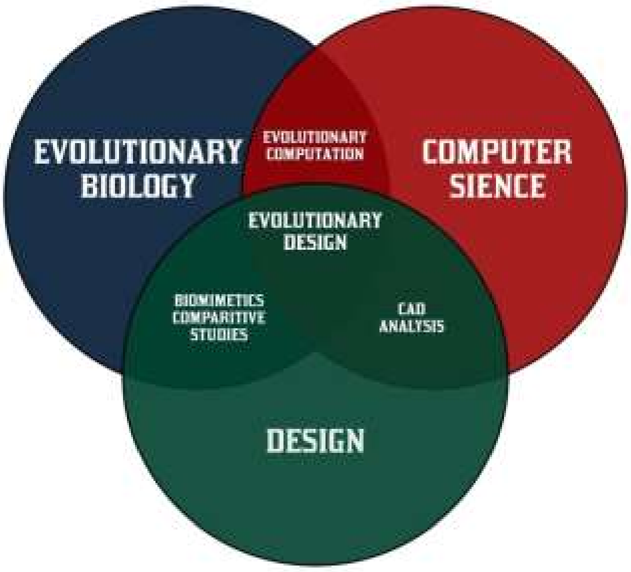


**Figure (3- 1): Evolutionary** design has its roots in computer science, design and evolutionary biology Source: Bentley 1999

Bentley's introduction on his book which borrows concept from natural evolution that it is in a sense; evolutionary computation is more specified term. To define the term "evolutionary design", Bentley classified four main aspects of evolutionary design. Which are evolutionary optimization, creative evolutionary design, evolutionary art and evolutionary artificial life form, these aspects are used in a combination with other categories and the combination defends on the purpose of design because the work of researchers does not specify within only one category. (Bentley, 1999)

To be clear it is stated in detail that as a matter of fact, Evolutionary Design is derived from biological principles, specifically those of the theory of natural selection. In this approach the evolutionary model of nature is applied as the "generative process for architectural form". The architectural concepts are being expressed in terms of genetics. The design rules are instructed as generative rules which are applied, evolved and thereupon evaluated by the use of computer models. The generative script of instructions produces numerous prototypical forms which are then evaluated on the basis of their performance in a simulated environment (Frazer, 1995)

## 3.3. NATURE OF EVOLUTIONARY MODEL:

**Evolutionary architectures have many proposes of its evolutionary model as the generating process for architectural form where we can state that the nature of evolutionary model is as following:**

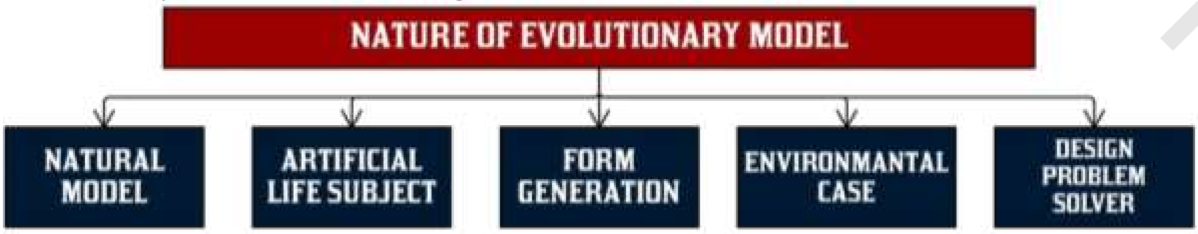


Figure (3- 2): Nature of evolutionary model chart
Source: Researcher 2014

### 3.3.1. Evolutionary Architecture as a Form of Natural Model

A. An evolutionary architecture investigates fundamental form generating processes in architecture. Paralleling a wider scientific search for theory of morphogenesis in the natural world
B. Evolutionary architecture proposes the model of nature as the generating force for architectural form
C. The profligate prototyping and awesome creative power of nature evolution are emulated by creating virtual architectural models which respond to changing environments successful developments are encouraged and evolved

### 3.3.2. Evolutionary Architecture as a Form of Artificial Life Subject

A. Artificial life subject like the natural world, principles of morphogenesis and genetic coding replication and selection

### 3.3.3. Evolutionary Architecture as a Form Generation

A. The rules are described in a genetic language which produces a code script of instructions for form generation
B. Space, structure and form are the traditional outward expressions of an architectural concept which has developed in the mind of architectural concepts are expressed as generative rules so that their evolution may be accelerated and tested

### 3.3.4. Evolutionary Architecture as a Form of Environmental Case

A. Natural eco-system has complex biological structure. They recycle their materials, permit change and adaptation, and make efficient use of ambient energy
B. by contrast, most man-made and built environments have incomplete and simple structures: they don't recycle their material, materials aren't adaptable and they waste energy
C. An ecological approach to architecture doesn't necessary imply replicating natural ecosystems, but the general principles of interaction with the environment are directly applicable through evolutionary process and algorithms

### 3.3.5. Evolutionary Architecture as a Form of Design Problem Solver

There is a number of evolutionary design systems used in architecture design to solve problems like design optimization and layout formation which are mostly best used in architecture design. (Gurer, Cagdas, 2006) (Johnson, 2001)

## 3.4. EVOLUTIONARY DESIGN BY COMPUTER – EVOLUTIONARY MODELS

*"... Evolving designs on computers now enables us to employ computers in every stage of the design process. This is no longer Computer Aided Design - this is becoming Computer Design where form follows process".* (Bentley, 1999)

The use of computer seems to be more extended in evolutionary design. It becomes more creative tool to design because the computer is generating and developing the initial concept, from phenotype to genotype with genetic algorithm. Even if it is mentioned that the use of computer is changed in a progressive way, it was pointed out that computer cannot be a designer. It is described as 'unconscious digital slave'; it can be only working with a logical programming language. These programs operate computers what, when, and how to do, Not the computer itself.

The meaning of computation in evolutionary design means the generative programming codes, so called Genetic Programming. Genetic Programming with Genetic Algorithms, which allows computer to evolve design, is the most widely used form of evolutionary computation so far. (Bentley, 1999)
The evolutionary models require an architectural concept to be described in a form of "genetic code". This code mutated and developed by computer program into a series of models in response to a simulated environment. The models are then evaluated in that environment and the code of successful models used to repeat the cycle until a particular stage of development is selected for prototyping in the real world.

## 3.5. GENERATIVE EVOLUTIONARY DESIGN SYSTEMS

As mentioned in chapter Two **(2.6.6)**: Evolutionary design approach is classified into two concepts generative genetics and parametric. We will introduce the design input methods to generate evolutionary design, Prior to embarking on an exploration of the generative and evolutionary systems in architectural evolutionary design, it is worth exploring the algorithms that are used as the main engine directing the workings of the computer's processing power.
These generative evolutionary design systems can be divided to be: see figure: (3-3)

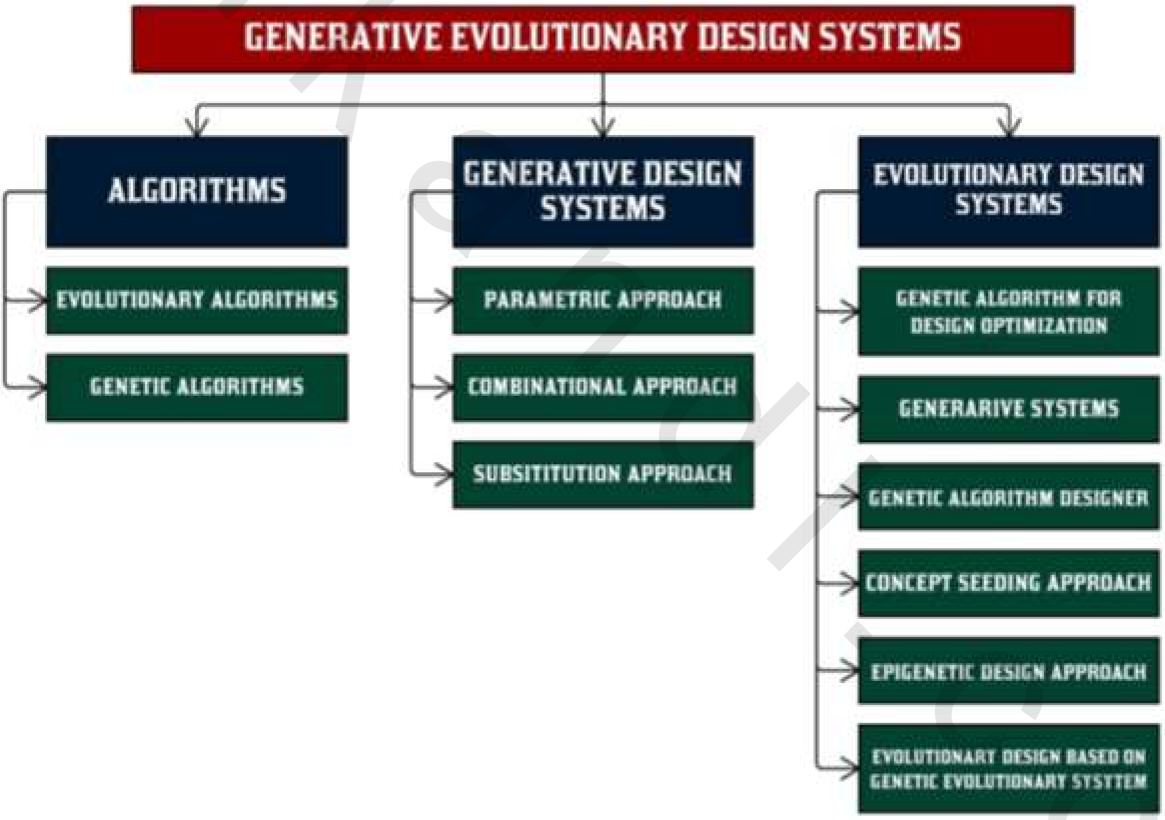


Figure (3- 3): Generative evolutionary design systems chart.
Source: Researcher 2014

### 3.5.1. Algorithms

The word algorithm itself believed to be derived from the Latin form (Algoritmi) of the Arab mathematician Mohammed ibn Musa Al Khowarizimi's name. Although algorithms are used for every computer process, they have been used as 'mathematical or logical mechanisms for resolving practical problems' for many years before the invention of the computer (Terzidis, 2006)
The focus should be on the algorithms that act as a 'language' for the human to computer interaction and which 'drive' and direct the computer's processing power.

Terzidis mentioned that:

> " … *Algorithmic tools are abstract, rational and intellectual in nature and therefore related to the human mind [which in turn means that] the output of an algorithm must be associated to a human mind, either the programmer or the designer*" (Terzidis, 2006)

This would suggest that for a designer to be able to gain as much as possible from the use of computers and to be able to direct the outcomes of the algorithms, he should understand and direct the actual processes of the algorithm.

An algorithm, which is a logical and mathematical process, written by a human developer, seeks to address a problem in a finite number of steps, some of these problems are:

- The wonderful flexibility of algorithms is that attempts can be made to solve problems whose target is known, but they can also be used to solve problems whose target cannot be defined. The algorithms thus 'become the means for exploring possible paths that may lead to potential solutions" (Terzidis, 2006)
- Another interesting feature of algorithms is that they can generate other algorithms, 'not only precise, identical, multiple copies of themselves but also structured text (i.e. code) that when executed will behave as an algorithm" (Terzidis, 2006)

The use of algorithms does not have to be limited to copying, simulating or replacing manual methods of design, but can also be 'studied as methodologies that operate in ways similar, parallel or complementary to that of the human mind". In addition because algorithms are able to generate novel concepts, new ideas and forms, they will affect the way a human designer thinks and so change his preconceptions and ultimately the way he designs.

#### 3.5.1.1. Evolutionary Algorithms

Nature is constantly changing and evolving and is used as a great source of inspiration for human designers and programmers alike. Individual organisms in nature contribute to the evolving and adapting of the whole population. Individual organisms are made up of both a genotype and a phenotype. The genotype consists of a set of genes (the DNA of the organism) and the phenotype is the fully-developed organism. The lifecycle of any organism can be seen in three phases1- reproduction 2- development and 3- survival. Through the three phases of the lifecycle of the organism, the evolution and adaptation of the population occurs.

The concept of evolution has been used in computation in the form of 'Evolutionary Algorithms' (EA), which, according to (Janssen, 2004), have been very successfully used in many domains and for a wide variety of problem types. Inspired by the processes of inheritance, mutation, selection and the crossover of natural evolution, EAs are, in many ways, analogous to the process of natural evolution.

**Evolutionary algorithms are used to:**

-Generate solutions to problems of optimization through a cyclical process that evolves a population of individuals in a continuous process of manipulation that allows the population to evolve and adapt as a whole (Janssen, 2004). The individuals that make up the overall population can be anything from a solution to a problem, a design that fulfills certain requirements or even a set of parameters in an equation (Janssen, 2004)

- Mirror nature: The genotype representation is a highly encoded version of the entity being evolved and the phenotype representation is a decoded version of the genotype. The reproduction step creates new genotypes; the development step transforms genotypes into phenotypes; and the survival step allows some phenotypes to survive.

- Although EAs 'mirror' and 'mimic' natural evolution, there is a key difference between EAs and natural evolution, EAs require an additional step: the evaluation process that assesses the performance of each individual of the population. In a natural evolution, the assessment of the performance is simply a matter of survival. If an individual (organism) does not die, then it will remain in the population. With EAs, on the other hand, the evaluation of fitness is performed explicitly to assess whether an individual can perform one or more of its objectives. (Janssen, 2004)

#### 3.5.1.2. Genetic Algorithms

Genetic Algorithms (GA), which can be seen as a special form of EA, use heuristics to search and mimic natural evolution. (Liu and Frazer 2002) explain the workings of genetic algorithms as being:

> *... Highly parallel mathematical algorithms that transform populations of individual mathematical objects (typically fixed length binary character strings) into new populations using operations patterned after (1) natural genetic operations such as sexual recombination (crossover) and (2) fitness proportionate reproduction (Darwinian survival of the fittest).*

GAs are used to simulate the concept of evolution with computer logic. GAs have a long history of development. They were first presented by John Holland in the 1960s in his investigation of the process of natural selection systems. (Holland, 1992)Much like nature, GAs begin with an initial population of individuals which can be generated at random or heuristically (Chan, 2008).At each step of the evolutionary process, the individuals in the current population are decoded and evaluated according to a predefined quality criterion, (usually referred to as the fitness, or fitness function).

> *"To form a new population (the next generation), individuals are selected according to a selection scheme, such as the roulette wheel selection scheme, where individuals are selected with a probability proportion to their relative fitness. This ensures that individuals with greater fitness have a better chance of 'reproducing'"* (Chan, 2008)

Genetic algorithms can be used to:

- Perform the evolutionary process of crossover to produce a new generation of individuals (or solutions).Crossover is the process through which parts of the 'parents' or genomes (the encoding) of two selected individuals are exchanged.

- By exchanging parts (usually the stronger or fitter parts) of the parents to the new individuals, 'offspring' or 'children' are, in effect, formed. At the most basic level, after a crossover point is randomly selected, 'substrings' can be exchanged. The crossover operation can be very effectively used in problems that involve the search for new aspects of a search space.

- Perform the mutation processes found in natural evolution. The mutation operator is processed by the genetic algorithm by 'flipping bits at random, with some small probability' (Chan, 2008)

The mutation operation of the GAs randomly samples new points in the search space, and can be used in the attempt to stop convergence to a local optima occurring prematurely, that is, a situation where the solution is clearly not the optimal solution, but no other alternatives present themselves.

Every process or operation of a GA undergoes a vast number of iterations at random, which means that convergence may never be reached. In cases where it is likely that there will be too many possible solutions and thus too many iterations for the GA to process, the designer must specify a point of termination for the GA. This condition for termination can take various forms, ranging from setting a maximum number of iterations or generations to setting an acceptable level of fitness (Chan, 2008)

The genetic make-up of the second (and all subsequent) generations in the evolutionary process is dependent on the process of selection that decides which part of the population will pass to the next generation. All solutions generated by the GA in the population are evaluated and the selection of the successful solution is made on the basis of fitness. Because the GAs process a vast number of possible solutions, as opposed to just one solution at a time, it is much more likely that an acceptable solution will be found (Kalay, Architecture's new media: Principles, theories, and methods of computer-aided design, 2004)

### 3.5.2. Generative Design Systems

Using performance-driven generative design methods actually enables the computer to become a 'design generator (Shea, Aish & Gourtovaia, 2005).Design in a general sense can be seen as the search for solutions to design problems. According Herr and Kvan to generative design solutions are the result of what they call the search for 'strategies to facilitate' that very search or exploration for solutions through the use of computers as 'variance-producing engines to navigate large solution spaces to achieve unexpected but variable solutions'. Computer-aided generative design systems make use of algorithms to generate a host of different solutions from a given set of design goals and constraints. The use of the computer in generating designs can be an invaluable aid to the human designer, because it is able to compute large and complicated sets of numerically formalized dimensions and constraints that a human designer would not be able to process with the same accuracy or speed. (Herr, Kvan, 2007)

### Categorization of Generative Design Approaches

Janssen identifies three main approaches to generative design that can be used within the developmental step of an evolutionary system to create alternative design models. He highlights three approaches; he acknowledges that other approaches and techniques to generative design exist. The three approaches that were focused on are the:
1) - Parametric approach, 2) - The combinatorial approach 3) - The substitution approach.
The combinatorial and substitution approaches are more flexible than the parametric approach. However, a combination of all three of these approaches can be used in a generative evolutionary design system.

#### 3.5.2.1. Parametric Approach

The parametric approach to generative design systems allows the generation of a range of forms by first assigning parameters to either a model or a procedure and then varying some of these parameters. It points out that the parametric approach does not only cover approaches where only dimensional parameters are varied, but actually encompasses rational modeling, variation design, constraint-based design and so on.
It was viewed from a wider perspective that the parametric approach can be seen as the 'varying of constraints, where dimensional parameters are just one type of constraint'. The parametric approach offers a great deal of control to the designer over the generation process.
Highlighting two techniques of parametric modeling are the 1) - variation technique and 2)- the history-based technique.
The variation technique is used in parametric modeling systems and must have a predefined model of the form to be generated. Using this technique, the constraints or parameters (including the dimensional parameters) of a model of the form that is going to be generated are defined as equations, which are subsequently solved 'simultaneously by a constraint solver' (Janssen, 2004)

The history based technique, on the other hand, generates forms incrementally through a series of operations that require certain data values. The manipulation of the generated form can be carried out by the operations or the data that is entered for each operation. This history technique generates forms through referencing the sequence of operations used to generate a form, while the variation technique does not make any reference to past operations.

#### 3.5.2.2. Combinatorial Approach

The combinatorial approach, which generates forms by combining a predefined set of elements, is the most general kind of approach to form generation. The combinatorial approach generates forms by combining and assembling elements (or components) of different types, although it is also possible that the components are of just one type. It was found that combinatorial approach identifies two combinatorial techniques, the first using an algebra technique and the second using a template technique.

#### 3.5.2.3. Substitution Approach

The substitution approach generates forms by beginning with a seed form that is provided. Modification and manipulation can be carried out by repeatedly substituting parts or components of that seed form into the new parts to generate a new form. The manipulation of the seed form is carried out by a set of transition rules that change or modify elements of the form.

The transition rules are, if then type rules, which specify 'that if a certain configuration occurs … then [the] configuration must be replaced by a new configuration'. The transition rules can be applied a number of times during the generative process and can also be applied to the generated forms they produce. Not only can a variety of forms be generated by modifying the seed form but Alternative forms can also be generated by modifying the transition rules, modifying the total number of steps carried out by the rules or modifying which transition rules to apply. The substitution approach can be used to generate very complex forms using only a small number of rules. Nevertheless the approach is both unpredictable and difficult to control.
The substitution approach to generative design can be achieved by using one of two techniques:1)- Generating forms by using a grid or 2)- generating forms by analyzing the shapes of the components.The 'grid-based' substitution technique uses a pre-defined cellular grid into which the substitution of the various parts of the form can be made. On the other hand, when using the 'shape-based' technique, the substitutions are 'performed based on the geometry of the individual shapes'. The design space, rather than being defined by a grid, is a continuous space. Growth is allowed in this continuous space where new shapes substitute existing ones. An example is shape grammars which are especially useful in designing forms that are differentiated primarily on the basis of form yet driven by function.

### 3.5.3. Evolutionary Design Systems

Evolutionary architecture is the study and practice of architectural form generating processes that use the model of nature and the morphogenesis found in nature as its inspiration. The term 'evolutionary architecture' (Frazer J. , 1974) (Frazer, J., Connor, J. , 1979) (Frazer J. , 1995)was used to describe the efforts to emulate 'the awesome creative power of natural evolution by creating virtual architectural models which respond to changing environments' (Frazer J. , 1995)The ultimate aim of evolutionary design is for the built environment to have the 'symbiotic behavior and metabolic balance' that is characteristic of the natural environment. As (Herr,Kvan, 2007)explain it, in evolutionary design approaches, the design process is organized as a cyclic process that generates increasingly appropriate solutions by way of repeated selection at every design cycle

Evolutionary architecture, unlike traditional architectural design systems, incorporates more than just the idea of spaces and forms, but includes a set of generative rules stated in a genetic language that produces instructions for the generation of those spaces and forms (Frazer, 1995). The generative rules used to evolve design solutions in evolutionary architectural systems are evolutionary algorithms (discussed in Section 3.4.1.1). These evolutionary algorithms seek to mimic the adaptive evolutionary processes of nature by applying the notions of natural selection, mutation and recombination to computer systems that are designed to generate architectural spaces and forms. The evolutionary algorithms are used to evolve a population of configurations in a process analogous to natural selection which uses objective functions, called 'fitness functions', to evaluate the evolved solutions.
The evaluation and selection of the generated design solutions is carried out by simulating the environment of the designed entity, which is thus developed out of a series of evolutionary steps based on the 'artificial' selection of solutions that best respond to predefined fitness function criteria, with the removal of poorly performing designs.
It was explained that this artificial selection as the selective breeding carried out by humans to produce a desired evolutionary response. Also evolutionary systems have been implemented in various fields of design, including graphic, media, art and industrial design and the main problem encountered has been that convergent mechanisms of standard evolutionary algorithms are difficult to achieve. (Gu, Tang and Frazer, 2006).
Evolutionary design systems and approaches are outlined to six key approaches. These are 1) - Genetic Algorithm for Design Optimization (GADO), 2) - Generative System (GS), 3) - Genetic Algorithm Designer (GADES) 4) - Concept-seeding approach and 5) - Epigenetic design approach 6) - Evolutionary design based on genetic evolution system (EDGE)

#### 3.5.3.1. Genetic Algorithm for Design Optimization (GADO)

The Genetic Algorithm for Design Optimization (GADO) uses a genetic algorithm for continuous design space optimization that uses new GA operators and strategies tailored to the structure and properties of engineering design domains.

**GADO is an evolutionary design system which:**

Focuses on the parametric design phase, where more detailed decisions about numerical aspects of the design are made, rather than on the structural design phase, which involves making high level decisions about the overall shape of the artifact (Rasheed, 1998)
GADO is a parametric evolutionary design system that can be used to optimize engineering design by using a parallel master-slave model and a steady state asynchronous evolution mode.
The GADO system uses and maintains an initial population, with specially-designed operators for crossover and mutation being applied to this population. Because designs are created through these operations working on the initial population, they are added to the population. A module has also been developed to ensure that diversity in the population is maintained (Janssen, 2004).

#### 3.5.3.2. Generative System (GS)

The Generative System (GS) is like GADO, also a parametric evolutionary design system. The GS focuses on the environmental performance of the building, rather than the engineering aspects. GS seeks to develop designs that can then be simulated using many applications. For instance, the DOE-2 application was used by the GS for lighting and thermal calculations and, sometimes Pareto multi-criteria optimizations were also used. With the GS, the systems follow the general synchronous evolutionary architecture, and uses standard rules and representations (Janssen, 2004)

#### 3.5.3.3. Genetic Algorithm Designer (GADES)

The Genetic Algorithm Designer (GADES) is a generative evolutionary system developed by Bentley. It can be used for a variety of design types. The GADES system follows the general synchronous evolutionary architecture, but uses specifically developed rules and representations. The evaluation phases or routines used by GADES are also customized and thus the system is supposed to be highly generic and applicable to any design domain

#### 3.5.3.4. Concept-seeding Approach

In Frazer's approach, the designer captures and codifies his design ideas, which are then entered into a computer program that can generate new designs that still embody the original design ideas. This approach of Frazer's is described as the concept-seeding approach, because it allows the designer to capture the design idea as a seed that can subsequently be manipulated and developed in response to specific problems

It can also be evaluated by the evolutionary system. The capturing of the seed is achieved by creating a set of rules and representations that encompass the designer's design concept or ideas. The seed of the concept can include any design considerations, such as the formal, structural, constructional, aesthetic or indeed any other consideration. (Janssen, 2004)

#### 3.5.3.5. Epigenetic Design Approach

The epigenetic design approach extends the role of the environment out of the evaluation phase of the design and allows designs to actually be generated in response to the design environment and can apply bio digital concepts in its design progress. A number of evolutionary design systems have the capability of using encoded environmental information and bio digital inputs in the evaluation step.

This encoded environmental information can take the form of encoded design constraints or encoded design context. The epigenetic design approach, on the other hand, uses this environmental information in the developmental stages of the design process that, in reality means that the ultimate successful design is even more adapted to its environment. It can be divided into two parts:

- A design which actually response to the design environment according to inputs or chosen environment
- A design applying bio digital, biological and genetic environmental concepts in evolutionary design system getting a design familiar with environment

#### 3.5.3.6. Evolutionary Design-based on Genetic Evolution System (EDGE)

The genetic/evolutionary design model (EDGE) is based on the concept of schemas which provides the representational framework for design knowledge. Two kinds of design schemas are suggested to formulate design knowledge and interpret the knowledge into genetic codes.

**The design rule schema**

The design rule schema plays the role of formulating design knowledge as design elements manageable by the design process; hence it comprises a target situation and a transformation operator. "The result of the rule application is not included in the schema because it is not manipulated by the design process but only appears in the phenotypic structure", In order to convey the required information to the genetic search mechanism for the identification and manipulation of the design elements, the design rule schema is being translated in genetic codes via the design gene schema. **The design** gene schema: A design gene schema can instantiate numerous design genes by assigning possible values to the components. One of the genetic model's constituents is the interpretation knowledge; this reflects the information specified by the user.

## 3.6. THE EVOLUTIONARY DESIGN APPLYING REQUIREMENTS

### 3.6.1. The evolutionary model requirements:

A design concept is described in a genetic code. The code is then mutated and developed in a computer program into a series of models in response to a simulated environment.

The models are then evaluated in the simulated environment and the code of successful models is selected.

The selected code is then used to reiterate the cycle until a particular stage of development is selected for prototyping in the real world. (Liu, Tang, and Frazer, 2002)

- The genetic code can be applied as one of the previous generative systems or through some inputs and plug-ins in software as genr8 in Maya or Grasshopper Galapagos in Rhino, etc.
- In order to create a genetic description, it is first necessary to develop a design concept in a generative manner capable of being expressed in a variety of forms in response to different environments. This is a manner in which many designers already work in the sense that they have a personal set of strategies that they adapt to particular design circumstances. This strategy is often very pronounced and consistent to the point where a designer's work is instantly recognized. (Graham, 1995) (Jian, 2002)

### 3.6.2. Evolutionary Design Process and Synthesis

There are a number of influential factors in the evolutionary design process. (Liu, Tang, and Frazer, 2002); (Räihä, Koskimies, Mäkinen, 2008)

#### 3.6.2.1. Functional requirement

A major problem in automated software architecture synthesis is the representation of functional requirements. Since the technique should be applicable in any domain, we cannot make assumptions about the actual semantics of the functional requirements. Yet, although software architecture design is usually driven by quality requirements rather than by functional requirements, the architecture is senseless without functionality. (Räihä, Koskimies, Mäkinen, 2008)

#### 3.6.2.2. Architectural patterns

An architectural pattern can be any general structural solution applied at the architectural level to improve some quality attribute of the system. Architectural patterns have been systematically catalogued as architectural styles and design patterns. (Räihä, Koskimies, Mäkinen, 2008)

#### 3.6.2.3. Architectural laws

The purpose of architectural laws is to prevent various kinds of anomalies in the architecture. Mutation and crossover operations are implemented in such a way that these laws always hold. (Räihä, Koskimies, Mäkinen, 2008)

#### 3.6.2.4. Initial population

An initial population is first produced, where only basic structures, such as class division and interfaces for the responsibilities are randomly chosen. To ensure as wide a traverse through the search space as possible, four special cases are inserted: all responsibilities being in the same class, all responsibilities being in different classes, all responsibilities having their own interface, and all responsibilities being as much grouped to same interfaces as the class division allows. (Räihä, Koskimies, Mäkinen, 2008)

#### 3.6.2.5. Genetic encoding of architecture

In order for the genetic algorithm to operate on software architecture, the architecture needs to be represented as a chromosome consisting of genes. For efficiency, in this experiment the architecture encoding is designed to suit the chosen set of architectural patterns. (Räihä, Koskimies, Mäkinen, 2008)

#### 3.6.2.6. Mutation and crossover operations

All mutations are implemented as either introducing or removing an architectural pattern. This ensures a free traversal through the search space, as moves that may have seemed good at one time can be cancelled later on. All mutations except for introducing a message dispatcher or a design pattern operate purely at supergene level by changing the value of one field. Introducing a new dispatcher to the system.(Räihä, Koskimies, Mäkinen, 2008)

#### 3.6.2.7. Fitness function

The fitness function is based on widely used software metrics, most of which are from the metrics suite introduced by Chidamber and Kemerer. These metrics have been used as a starting point for the fitness function, and have been further developed and grouped to achieve clear "sub-fitness's" for modifiability and efficiency, both of which are measured with a positive and negative metric. The biggest modifications to the basic metrics include taking into account the positive effect of interfaces and the dispatcher architecture style in terms of modifiability, as well as the negative effect of the dispatcher in terms of efficiency. A complexity metric is added to penalize having many classes and interfaces as well as extremely large classes.

Dividing the fitness function into sub-functions answers the demands of the real world. Hardly any architecture can be optimized from all quality viewpoints, but some viewpoints are ranked higher than others, depending on the demands regarding the architecture. By separating efficiency and modifiability, which are especially difficult to optimize simultaneously, we can assign a bigger weight to the more desired quality aspect (Räihä, Koskimies, Mäkinen, 2008)

## 3.7. APPROPRIATE ALGORITHMS

The word algorithm used in evolutionary design refers to a set of specific instructions, usually given to a computer. Many algorithms could be written to carry out the process of the evolutionary mechanism. Each GA brings together methods of translation, evaluation, selection, and variation. Some methods work together better than others, First the architect must choose an appropriate schema, or means of representing the phenotype as a genotype. When variation is introduced by mutation only, the genetic information can be encoded in any order. (Janssen P. , 2006)

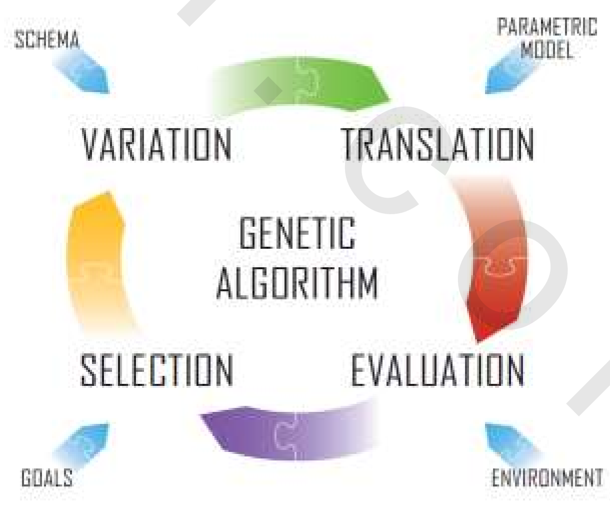


Figure (3- 4): An effective genetic algorithm needs to be assembled from pieces. Different methods are appropriate to different problems.
Source: (Janssen P. , 2006)

## 3.8. USES OF APPLYING EVOLUTIONARY DESIGN IN DIGITAL ARCHITECTURE

It was counseled that form follows function. Likewise, for achieving form akin to nature, our approach has been to have form follow process the growth and evolutionary process that occurs in the living world. Essentially, this approach of evolutionary process allows designers to capitalize on Nature's strategies which are the most compelling means of achieving Nature's outcome. Through all these previous evolutionary systems applying evolutionary design; evolutionary design processes is used in architectural digital design to form and solve many problems some of them are the following:

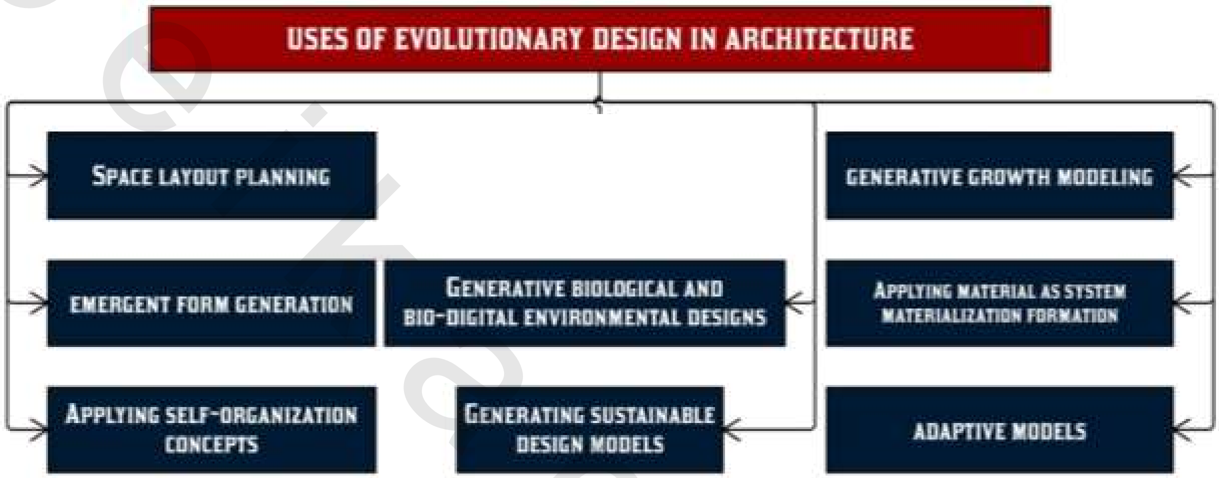


Figure (3- 5): what's the products evolutionary design uses to produce?
Source: Researcher 2014

## 3.9. WHY DO WE USE EVOLUTIONARY ARCHITECTURE APPROACH:

Because we can have:

• More diverse area of the design space can be traversed than with other methods
• Probabilistic selection method directs the random generative process towards meaningful and satisfactory solutions
•Suitable generated forms which are not necessarily globally optimal but satisfy a range of client, social and designer requirements.
• Pool or population of design solutions to be used rather than a single solution.
• Generations of new solution through mutations and crossovers of previous elites.

## 3.10. CHAPTER SUMMARY

This chapter introduces the evolutionary design concept by discussing its theoretical approach, and then the nature of the evolutionary model is presented. There is focus on the system used to apply evolutionary design and the mostly used algorithms which shows where, when and why to apply this algorithms, with a brief note on all their types.

The requirements for applying the evolutionary approach are discussed and the important terminologies and uses of the evolutionary design approach follows. All of the previous content helps in understanding the evolutionary process design approach and its ways for choosing the applying systems that will help in easily combining between this chapter and the next on chapter four.

However, the use of evolutionary models show promise in the formulation of computable models for architectural design. Due to their recognition of nature's means, they have embraced both growth processes as a means of deriving such form and evolutionary process as a means of exploring and creating ideal form. Thus, this will lead us to the analytical study of evolutionary design with its two main approaches in the following chapter then its applied examples and tools in chapters four and five.