A DISCUSSION ON ARCHITECTURE AND AI STUDIES THROUGH THE SEEK AND CITYMATRIX PROJECTS

by
Taner İzzet Acarer

Submitted to Graduate School of Natural and Applied Sciences
in Partial Fulfillment of the Requirements
for the Degree of Master of Science in
Architecture

Yeditepe University
2021

A DISCUSSION ON ARCHITECTURE AND AI STUDIES THROUGH THE SEEK AND CITYMATRIX PROJECTS

APPROVED BY:

Assoc. Prof. Dr. Ece Ceylan Baba ....
(Thesis Supervisor)
(Yeditepe University)

Prof. Dr. Emin Erkan Korkmaz ....
(Yeditepe University)

Assoc. Prof. Dr. Ümit Arpacıoğlu ....
(Mimar Sinan Fine Arts University)

DATE OF APPROVAL: ..../..../2021

ACKNOWLEDGEMENTS

I would like to express my gratitude and special thanks to my supervisor Assoc. Prof. Dr. Ece Ceylan Baba for her guidance, criticism, encouragements and supporting suggestions. It has been a great honor to work with her.

I present my special thanks and love to my precious parents Sultan Acarer, Basri Acarer and my brother Caner Ceyhun Acarer for their endless love, generous support and encouragement on every decision that I take throughout my life. This thesis is dedicated to them.

ABSTRACT

A DISCUSSION ON ARCHITECTURE AND AI STUDIES THROUGH THE SEEK AND CITYMATRIX PROJECTS

Architectural design process involves a set of complex problem-solving processes due to its interdisciplinary nature. This complex process requires an intense study by architects, collection of different data, and creation of design alternatives within the scope of concepts, and selection of most ideal design among such different alternatives.

While artificial intelligence (AI), which has been studied for a long time and has become a focus of interest today, was used to make electronic data transfer and to make complex computations before, it can gather information regarding events/instances and make decisions about events/instances today. AI studies brought along machine learning as a result of development of artificial neural networks. Hence, AI can also learn the relations between events/instances. The ability of AI to analyze events/instances, make decisions, and learn the relations between them rapidly is thought to play an effective role in architectural processes.

This thesis examines the effects of the use of AI studies, which stand out in many fields today, in solving the complex structure of the architectural design process and consideration of AI as a decision-making mechanism on “the role of architect in architectural design process and his/her effect on the design processes”. For this purpose, firstly, the maps of the design process arisen especially with the influence of modernism and gained importance in the post-World War II period, were evaluated, and then the definition, historical development, and working mechanism of AI technology have been examined and the "CityMatrix" project, an up-to-date project created as a thesis study at MIT in 2017, and the "SEEK" project, one of AI's first exemplary works in the field of architecture created in 1970 by Nicholas Negroponte and the Massachusetts Institute of Technology (MIT) Architecture Machine Group (AMG) have been comparatively examined in terms of their AI technologies and their general structures. In the conclusion of this study, potential effects of the AI technologies that attained a place also in the architecture field, as in all fields, and the predictions on the change of architectural design process as a result of such effects have been addressed.

ÖZET

SEEK VE CITYMATRIX PROJELERİ ÜZERİNDEN MİMARLIK VE YAPAY ZEKA ÇALIŞMALARI ÜZERİNE BİR TARTIŞMA

Mimarlığın disiplinlerarası bir alan olması nedeniyle, mimari tasarım süreci bir dizi karmaşık problem çözme sürecini içinde barındırmaktadır. Bu karmaşık süreç, mimarlar tarafından üzerinde yoğun bir şekilde çalışılmasını, farklı verilerin toplanarak bağlamlar kapsamında farklı tasarım alternatifleri oluşturulmasını ve bu farklı alternatifler arasından en ideal tasarımın seçilmesini gerektirmektedir.

Uzun zamandır üzerinde çalışılmakta olan ve günümüzde yoğun bir ilgi kaynağı haline gelen yapay zeka (YZ), önceleri sadece elektronik veri transferi yapmak ve karmaşık hesaplamaları gerçekleştirmek üzere kullanılmaktayken, günümüzde olaylar/durumlar ile ilgili bilgileri toplayabilmekte, olaylar/durumlar hakkında kararlar verebilmektedir. YZ çalışmaları yapay sinir ağlarının geliştirilmesiyle makine öğrenmesini de beraberinde getirmiştir. Bu sayede YZ olaylar/durumlar arasındaki ilişkileri de öğrenebilmektedir. YZ'nin olayları/durumları hızlı bir şekilde çözümleme, karar verme ve aralarındaki ilişkileri öğrenebilme yeteneği, mimari tasarım süreçlerinde efektif bir rol oynayabileceği düşünülmektedir.

Bu tez kapsamında, günümüzde pek çok alanda kendini gösteren YZ çalışmalarının, mimari tasarım sürecinin karmaşık yapısının çözümlenmesinde kullanılmasının ve YZ'nin bir karar verme mekanizması olarak değerlendirilmesinin "mimari tasarım sürecinde mimarın rolü ve tasarım süreçlerine etkisi" üzerinde olan etkileri incelenmektedir. Bu amaç doğrultusunda, ilk olarak, özellikle modernizm akımının etkisiyle oluşmaya başlayan ve II. Dünya Savaşı sonrası dönemde önem kazanan tasarım süreci haritaları değerlendirilmiş, ardından, YZ teknolojisinin tanımı, tarihi gelişimi ve çalışma mekanizması incelenmiş ve YZ'nin mimarlık alanındaki ilk örnek çalışmalarından biri olan, 1970 yılında Nicholas Negroponte ve Massachusetts Institute of Technology (MIT), Architecture Machine Group (AMG) tarafından oluşturulan "SEEK" projesi ile güncel bir proje olan, 2017 yılında yine MIT'de bir tez çalışması olarak oluşturulan, "CityMatrix" projesi kullandıkları YZ teknolojileri ve genel yapıları ile karşılaştırmalı olarak incelenmiştir. Çalışma sonucunda her alanda olduğu gibi mimarlık alanında da kendisine yer bulan YZ teknolojilerinin mimarlık mesleği üzerindeki olası etkileri ve bu etkiler sonucunda mimari tasarım sürecinin değişimine dair öngörüler irdelenmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS/ABBREVIATIONS

AI	Artificial intelligence
AMG	Architecture Machine Group
ANN	Artificial neural network
ARPANET	Advanced Research Projects Agency network
BIM	Building information modeling
CAD	Computer aided design
CRT	Cathode ray tube
DARPA	Defense Advanced Research Projects Agency
DL	Deep learning
GA	Genetic algorithm
HPP	Heuristic programming project
KB	Knowledge base
MIT	Massachusetts Institute of Technology
ML	Machine learning
NLP	Natural language processing
NLS	oN Line System
PARA	Perceiving and recognizing automaton
SAINT	Symbolic automatic integrator
SIR	Semantic information retrieval
TUI	Tangible user interface
VDI	Verein Deutscher Ingenieure
WWW	World-Wide-Web
YZ	Yapay zeka

1. INTRODUCTION

Architecture, since its beginning, has been one of the most basic needs of human being. First architectural products produced for protection, accommodation, and survival purposes, especially upon adoption of a settled life, enabled the creation of living quarters based on cultures and needs later on. The increased level of complexity due to consistently developing human life and varying needs did not only lay the foundations of architecture as a sole discipline, but also caused it to have an interdisciplinary structure. Increased structural variety and gradually increased problems addressing to such structures inevitably caused architecture to have a complex structure. As the level of complexity increased, some studies have been conducted on the design processes.

Examining design in an interdisciplinary framework, the studies on the design process intent to analyze the structure of the design, and to provide the designer with auxiliary tools to cope with potential difficulties to be encountered when solving the problems that the design product is desired to address. The evolution of insuperably complex structures of new design problems, which cannot be handled with conventional approaches, into a more quality design understanding may be considered a basis for the design process studies. Design process studies also have a structure that is shaped according to the social, cultural and economic structures of the period in which they were created.

The change movement, which is experienced in almost every sphere of life by the formation of the modern world order, influenced every sphere of daily life, and ultimately the architectural design as well. While Taylorism, under the influence of the modernist movement, was effective on production and management models, the Fordist movement based on Taylorism, but enabled Taylorism to evolve into a real-time life transformation, caused issues such as mass production and standardization to gain importance and affect the whole world [1], and made an overwhelming impression in almost every sphere of daily life [2-3]. These movements, which were considered an easy way to rebuild the social and economic structure that collapsed after World War II, led to the complete adoption of standardization and mass production models, and even the design processes were influenced by these movements.

In the post-World War II period, the studies of design process models that aim to systematize design with certain rules have gained momentum. The created models propose to divide the design into certain phases, and to follow these phases just like the steps in a mass production system.

Computer systems, which gained importance during the war period with the developing technology, played a key role in the development of a new field in the 1950s. This field was the Artificial Intelligence (AI) named in the conference held in 1956 predicting that the machines can think like humans, struggle with problems, make reasoning and inferences. The projects that some architects designed after the 1950s started evincing optimistic points of view toward technological developments. Upon AI technologies came into prominence in different fields, the studies were conducted aiming to engage AI technologies into the field of architecture that has an interdisciplinary structure. The aim of these studies was to benefit from the computation power of the machines also in the field of architecture, as in almost every field. The reason underlying the use of AI technologies in architecture can be argued to be the major effects imposed on the architectural design problems by increased and developed complexity in human life as well as the emergence of new needs, alongside the fact that the architect has to struggle with all these problems all alone in the processes of working, collecting data, and producing a design product.

1.1. AIM OF THE STUDY

The aim of this study is to question the validity of the efforts to make the design process a systematic structure, which were created especially in the post-World War II period, in parallel with the developing technology, and to investigate the possible effects of developing technologies on the architectural design process and the professional role of the architect, within the strict rules of the modernist movement.

The architect/designer having certain steps and defined tasks based on the design process maps, is inadequate to cope with the uncertainties regarding the adaptation of the design to the unexpected and changing life dynamics. The main motivation of this study is comprised of the design process maps with strict rules within certain limits, the doubts on the impossibility of the design product to respond to the dynamism of developing technology and changing human life. AI technologies are likely to have effects that can interfere with the design product when necessary in the design processes and/or afterwards in line with the needs of the design, resulting in a change in the design process and the role of the architect. This thesis examines the cases of AI technologies that are tried to be applied to the architecture, and examines the changing design process maps and the architect's professional role.

This thesis also includes the benefits that the inclusion of AI technologies in the architectural design process can provide to the designer, and the a partial assessment of the pros and cons of these benefits. The outcomes of the research, including the changing structure of the design process and a limited review of the professional role of the architect, are discussed in the last chapter of this thesis.

1.2. SCOPE OF THE STUDY

This study focuses on the 1960s and later period, when the creation of design process models had increased importance with the effect of the modernist movement in the mid-20th century. The Design Methods Conference, held in 1962, caused the desire to create a systematic design process map to gain momentum. In this thesis, the structures of the design process models created in the mid-20th century and later on, and their way of handling the design were investigated. During the conduct of this research, the book of Nigel Cross named "Engineering Design Methods: Strategies for Product Design" [4] and the book of Bryan Lawson named "How Designers Think" [5] were used as the main sources of design process research and design process maps. This study is based on the division of the design process into two individual structures as "descriptive" and "prescriptive" in Nigel Cross's book "Engineering Design Methods: Strategies for Product Design".

In addition to understanding the computing power of the computer and the thought that it can contribute to production along with the computer technologies that gained importance in the World War II, it has been understood that intelligent machines can be created and the idea that intelligent machines are suitable for use in almost every field has spread. In the thesis, the structural features and working mechanisms of AI technologies, in which the intelligent-machine concept is considered as a specific branch, and the approaches used are discussed. Focusing on AI, which can be considered as a separate subject on its own, the AI approaches used by SEEK and CityMatrix projects examined in this thesis, the short historical processes and operational structures of these approaches are examined. Complete and detailed examination of AI technologies, a branch of computer technologies, is excluded in the conduct of this thesis, aiming to make the thesis easy-to-understand for the architects. All the information in this thesis, regarding the definitions, historical development processes, and approaches of the concept of AI, are expressed without technical structures of computer science and tried to be transformed into an easy-to-understand content.

SEEK project created by the Massachusetts Institute of Technology (MIT), Architectural Machine Group (AMG) under the leadership of Nicolas Negroponte in 1970 [6], and the CityMatrix project created in 2017 as a thesis at MIT [7], both in which AI was involved, are examined in this thesis. The primary purpose of selecting these projects is that SEEK and CityMatrix projects have different approaches of AI, and the secondary purpose for the same is the varied intellectual reflections of such two projects, which were created at different times, in including AI technologies in architectural design processes. The general structure of the projects, the methods they use, and the intent behind their creation were examined during this investigation. It was also addressed to that which of the "descriptive" and "prescriptive" design processes were used in structural terms in the creation of these projects. At the end of the study, the requirements of the design process maps, the change of design process maps in design models created with AI, and the possible effects of this change on the architect were examined.

1.3. METHODS OF THE STUDY

The initial step of this thesis was literature reviews on the concept of design and design process maps. The main method of this thesis is the qualitative method, and the historical analysis, architectural analysis, interpretation and comparison methods were employed under the qualitative method.

In this research, the possible change of design process maps with the inclusion of AI technologies in architecture, and the possible effects of this change on the professional role of the architect were discoursed. In the second chapter of this thesis, the design and design process have been conceptualized, and the structures of the design process maps, which emerged in the Design Methods Conference held in 1962 and the period later on and endeavored to transform the design into a systematic structure, and the intellectual background that these structures use to bring the design process into a systematic structure were investigated.

In the third chapter of this thesis, the historical processes and functioning structures of the computer technologies, which started to be used effectively in the World War II period and started to be implemented in all spheres of life due to their computation power and their contribution to the productive activities, and the AI studies conducted with the intent to create intelligent machines were addressed. Besides, the approaches that shaped AI studies were addressed and technical and structural information regarding these approaches were included.

In the fourth chapter of this thesis, the reason why AI technologies may be needed in the field of architectural design was investigated within the framework of the general features of architectural design, then, the features of the SEEK project created by AMG at MIT in 1970 and the CityMatrix project created as a thesis at MIT, which involved AI systems in the architectural design process, and the AI systems employed by such projects were examined comparatively.

The fifth chapter of this thesis is more of an epilogue rather than a definitive conclusion. In such chapter, the possible effects of AI technologies on the architectural design process and the professional role of the architect were investigated in the light of the information obtained throughout the thesis and a model that does not intent to systematize the design has been proposed in parallel with the design process maps.

2. EXAMINATION OF ARCHITECTURAL DESIGN PROCESS

Design is a phase that requires intensive knowledge, skills and, experience to find new solutions through testing, to meet the new necessities, through the evaluation of pre-existing things [8]. Norman Foster (2000) argues that everything around us that is made by humans and that we can perceive with our senses depends on the design process that requires careful choices and decisions [9]. The design itself is not just an act of creation. The structure of the design embodies an intense evaluation and decision procedure, and due to this procedure, the design becomes a process.

Weijnen et al. (2008) argue that the design process is considered among complex systems due to the fact that the infinite number of design tasks, restrictive elements, requirements, all other areas of design, the starting point of the design are not completely determined and the design cannot be directly modeled [10].

The design process is a practical and intellectual process that involves collecting data, identifying and analyzing design problems, reasoning, creativity, and finding solutions to problems. Bayazıt (1994) explains the design process as a sequence of actions consisting of techniques and tools used in the design phase [11]. Based on this statement, a design process can be described as a step-by-step flow of actions using technical knowledge and various tools to analyze all data about the design and design problem. The design process covers all activities from the thinking phase of the design to the production of the product.

Lawson (2005) defines the design process as a permanent and infinite procedure, as it does not have an exact and comprehensive definition and may have an unknown number of solutions. The design activity does not end at a certain time frame and can be constantly improved to produce better products within the process extending from the optimal solution to the best solution. In the design there is no sequence of actions that can help the designer find an optimal result in a suitable way. Hence, the designer needs to improve his/her ability to control and change the design process [5].

The architectural discipline, which benefits from many different fields such as humanities, social sciences, physical sciences, technology and creative art, includes an intensive and complex design process due to its interdisciplinary structure. While the architectural design process benefits from different fields due to its interdisciplinary structure, this benefiting method may differ depending on the designer's knowledge, understanding, correlation and cognitive abilities. In an architectural design process, the technical knowledge of the architectural profession and the knowledge of other related professions are taken into consideration to create habitable and nature-compatible spaces, as well as all other designs related to human life, and a reliable design from macro to micro scale.

The interdisciplinary nature of architectural design leads the design process to have a complex structure, besides, the infinite number of alternatives of design requires the designer to follow a certain set of rules during the design process. This set of rules followed by the designer throughout the design process takes place in the natural flow of the design process phases. Design researchers have tried to develop design process maps that explain the design process scientifically and systematically and investigate the design process with a prescriptive method in order to define the set of rules that designers follow voluntarily and / or involuntarily after the World War II.

These design process maps, which were created with the effect of technology, industry and modernism movements developed in the post-war period, differed according to the field in which the design will be made. Such differences are expressed with the phases in which the intellectual and prescriptive activities are systematized and divided into processes in the creation of design process maps.

2.1. MAPS OF DESIGN PROCESS

When the basis of the design process models is examined, the principles in Taylor's book The Principles of the Scientific Management, published in 1911 can be said to be effective. Taylor argues that the definition of maximum welfare can be achieved by increasing individual productivity at the maximum level, that is, by establishing the relationship with manpower, machinery, resources and structure, at the lowest cost. Systematic production is considered as a method that can maximize productivity through the interaction of new and unskilled workforce with the machine, leaving aside conventional methods [12]. Summarizing his twenty years of experimental studies and the practical studies of other researchers, Frederick Taylor argues that there is only one way to organize the business activity, and the purpose of rationalization is to find this way [1]. Taylor's study can be regarded as an effort to systematically create a productivity process that can be put into practice by ensuring a systematic division of labor.

Based on Taylor's scientific production model, a serial model first appeared in 1914 with the production system that Henry Ford implemented in his factory in Dearborn, Michigan, and the whole system, in which Taylor argued that each labor process can be radically increased by dividing them into separate actions and organizing these divided actions in accordance with the strict standards of time and motion research [3], was effectively implemented in this factory by Henry Ford. In the factory established by Henry Ford in Dearborn, Michigan, a labor process that is subdivided is transformed into a sequential production process, allowing workers to deal only with the jobs that are defined to them along a line. The main principle underlying the Fordist trend is to increase productivity through the creation of a system that defines the division of labor in an organizational scheme based on the Taylorist scientific management idea and is based on the continuous repetition of a defined task for each worker [13]. By virtue of this factory, which ensured the full adoption of mass production in the industrial field, the production standards have changed and a uniform design, uniform work and uniform worker models have been adopted.

In addition to all these, Fordism is quite far from a structure that only focuses on production, unlike Taylorism. Having a centralist structure, Fordism stipulates implementation of decisions unquestioningly, without giving the right to decide or have a say to the groups that are outside the upper levels of the hierarchical structure. Fordism argued that the products produced in a factory with low cost and labor force could be marketed even if they had a low quality, and that this could be achieved by the method of giving workers an opportunity to spare free time for themselves in addition to the wage they earned [2].

Taylorism, which tries to standardize production by establishing a completely systematic production style, and Fordism, which aims to transform everyday life into a systematic structure, not just limited to production, can be regarded as a reflection of the modernist movement, which includes the idea of putting aside all areas that are deemed old and outdated [2-3].

While America and Europe had different economic and daily life organizations in the Pre-World War I period, labor was cheap and raw materials were expensive in Europe, whereas it was the opposite in America. Gaining strength in the Post-World War I period, the American economy led to the adoption of similar business model and management systems in Europe [1]. The standardization of American production systems has played a role in the adoption of similar systems also in Europe to reduce the difference that has arisen due to mass production efficiency and labor savings between Europe and America.

When these business models and management systems are introduced to Europe in the 1920s, these systematic structures were opposed in cultural and sectoral means. In the 1930s, systems adapted to local conditions and demands made it possible to use mass production lines and models in Europe as well. The adoption of mass production systems and Fordist model in order to respond to the necessary needs for cultural and social revival of Europe's broken economy and lifestyle in the Post-World War II period was only possible with the loss of power of the labor and trade union classes [1-3].

This new systematic order that was adopted by Europe has reflected in other spheres of life as well as in industry, as the biggest reflection of the economic sphere. Here, it is possible to mention the effects of the modernist movement and Fordist thought, which has a wide impact in the economic sphere, on daily life. The thoughts aiming to handle the entire system as a whole have resulted in efforts to include the needs of daily life in this system. Design is one of such areas that are tried to be systematized. It is possible to talk about a deterioration in the structure of the design in a natural flow as a result of these developments, which anticipate that the processes that the design must address to should be formed within a certain systematic structure. Although the method the designer follows in creating a product is generally actualized within the framework of certain data, the process generally involves the intuitive actions of the designer. The systematic design process, on the other hand, may create a process that limits the intuitive actions of the designer.

In order to solve the socio-economic problems and design problems that came up as a result of the World War II and to meet the user needs, the concept of design has started to be considered as a problem-solving/decision-making action and the scientific developments brought about by the war aimed to contribute to the solution of design problems. After the World War II, the methods and techniques used in the development of war technologies started to attract the attention of many design researchers. The commencement of testing of the methods developed in the fields such as cybernetics, work study, ergonomics, applied psychology, activity research, system analysis in design in the periods following the World

War II has gathered interest in the perception of design as a science [14]. Although there is a lot of research on the design and design process in the Pre-World War II period, the design process researches, which gained importance in the post-war period, started to rise after many researchers started to conduct researches on design methods with a scientific and systematic study and presented their design method theories for the first time at the Design Methods Conference held in 1962 [15]. Underlying the introduction of new approaches to design is that designers can no longer rely on their ability to focus solely on the product as the center of a design task, and the necessity to emphasize the interest in considering human needs rather than equipment and form, due to the interest in mass production brought by technological developments. These new design ideas giving human needs prominence required a different perspective to design methods [16]. With these studies, the researchers have argued that, primarily, the functioning of the design process should be more important than the final design product. Stating that the design process should be able to keep up with technological developments to develop more effective design products, they have conducted studies on problem- and solution-oriented quests in the design process.

Although the design process models that came up with these studies have gained importance particularly in engineering fields, they also affect architecture due to the interdisciplinary nature of design. In the early examples of design process models, the structure of design act and design process researches has been formed in similar frameworks, although the engineering design process and the architectural design process are different fields. Jones and Thornley, editors of the Design Methods Conference in 1962, argue that the two processes have similar characteristics when describing the engineering and architectural design process [17]. Many researchers who conducted studies on the design process examined the design process in three main stages: analysis, synthesis and decision-making.

Jones (1984) explained these stages in an early example of design process models as follows:

> Analysis: listing of all design requirements and the reduction of these to a complete set of logically related performance specifications.
>
> Synthesis: finding possible solutions for each individual performance specification and building up complete designs from these with least possible compromise.
>
> Evaluation: evaluating the accuracy with which alternative designs fulfill performance requirements for operation, manufacture and sales before the final design is selected [18].

In the 1970s, firstly, Hillier et al. argued that the difference between architectural and engineering design processes models should be emphasized, stating that the designer should not only solve the problems, but also produce solutions by reconstructing the design problems. They suggested that the basic analysis-synthesis design model is considered as an epagoge formed by observations and that the information obtained by the designers from experience for solutions should be taken with bias. Hillier et al. therefore defended that a design model based on hypothetical methods might be more appropriate for architecture [19].

Nigel Cross and N. Roozenburg argued that two processes differ significantly as the engineering design process is a linear and sequential process and the architectural design process has a cyclical structure. While models of the engineering design process tend to emphasize the sequence of stages in which a project is expected to progress, architectural design models, on the other hand, emphasize the cycle of cognitive processes that the designer must fulfill. While the engineering design models, which emphasize the sequence of activities that occur during project development, exhibit a prescriptive approach, the architectural design process has a descriptive structure because it reveals the thought processes that should be used by the designer [20].

The basic idea under the design process maps is to draw a route from defining the design problem, which is the first stage of the design process, which proceeds in a predictable and definable logical pattern and consists of a series of different and definable activities, to solving the problem, which is the last stage [5].

In the following two sections of the chapter, prescriptive and descriptive design models discussed by Nigel Cross and N. Roozenburg are examined, and with which methods the design process models deal with the design process are investigated.

2.1.1. Descriptive Models of Design Process

Descriptive models of the design process often define the importance of creating a solution concept in the early periods of the process and thus reflect the solution-oriented nature of design thought. Such first solution assumption is then subjected to analysis, evaluation, idealization and improvement [21].

Markus (1969) placed the evaluation stage in the design process map in addition to the three basic stages that are predicated upon in design process models. Markus summarized the architectural design process in the table shown in Figure 2.1. The vertical dimension of Markus' architectural design process model shows the total framework of the design management, and the horizontal dimension of the model shows the stages of the design process [22]. Markus' architectural design process model showed that the stages of the design process follow an endless cycle in the management of the design. In the design process, each proposed step can reveal new and complex problems, and each new problem can trigger a new and more complex problem.

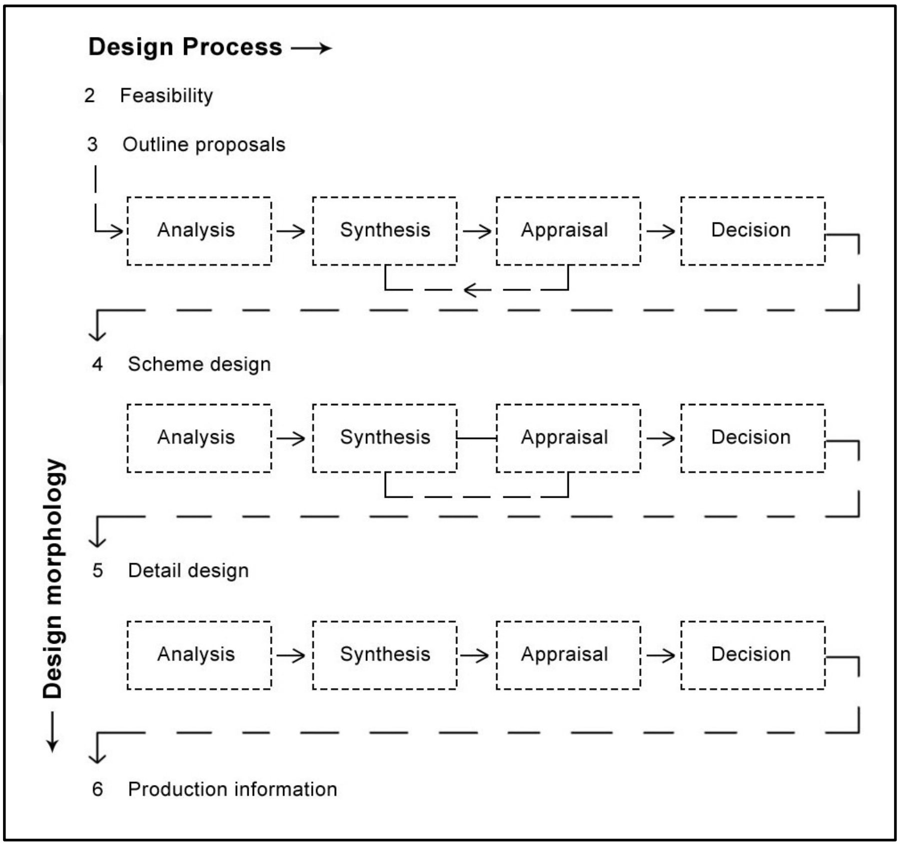


Figure 2.1. Markus's model of the architectural design process [5].

Maver (1970) describes the design process as an interactive and cyclical activity that includes analysis, synthesis, evaluation and decision that can be applied in a series of stages in a sequential design morphology, similar to Markus's design process map. As reported by Maver, the analysis phase is the main step of the design process. The analysis step involves clarifying objectives, identifying problems, quality of challenges, researching relationships, goal setting, and generating layouts from random data that contributes to the later stages of the design process. The synthesis phase includes the creation of partial solutions, and the combination of partial solutions. The evaluation phase includes evaluation, controls and tests, application of criteria, limitations and restrictions, and consistency tests. The decision phase involves choosing the best solution from a designated progression to the next morphological phase [5].

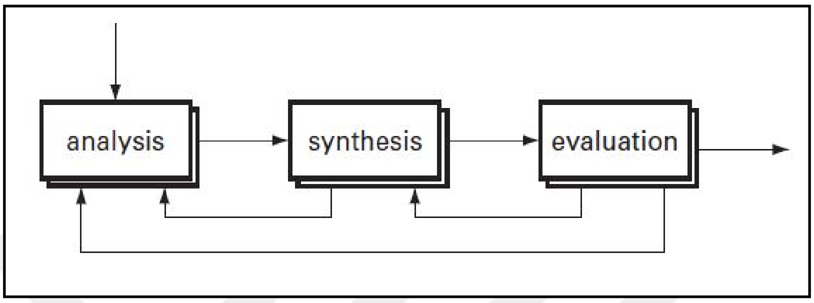


Figure 2.2. Lawson’s design process [5]

Lawson (1980) mentions that all early design process studies basically refer to three main stages as analysis, synthesis and evaluation, but that all stages must follow each other in an unsequenced manner from the beginning of the design process to the achievement of the design product. He argues that the process starting from draft proposals and extending to the detail design should actually continue with a circular feedback method as a whole (Figure 2.2) [23].

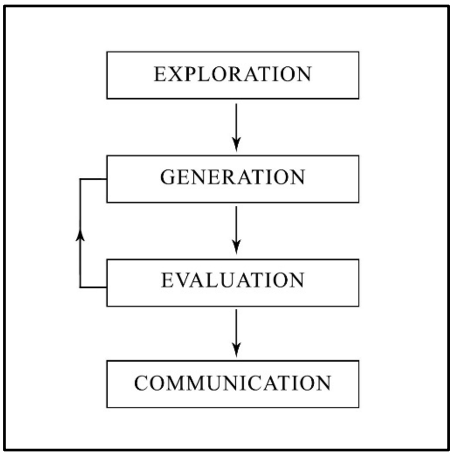


Figure 2.3. Cross's model of the architectural design process [4].

Cross (1942) explains the design process with a model consisting of four steps, taking into account the activities performed by a designer in the design process (Figure 2.3.). In Cross's model, the data available to the designer in the exploration phase does not allow the designer to fully understand the design problem, and the designer is only interested in understanding and solving the problem at this stage and tries to produce workarounds. In the generation phase, the designer puts forward the raw design proposal before making analysis and evaluation. Evaluation phase is the stage at which the designer decides on the product in general terms. Cross argues that the designer cannot ensure that the final design product comes out in the evaluation phase and the designer may always need to be in an iterative feedback loop that requires him/her to return to the production phase to produce a better final product. The presence of new problems seen in the design as a result of evaluation requires returning to the generation phase and producing new solutions. The communication phase is the most important step for designers to explain their work to everyone other than themselves in the most accurate way and to put their work into practice [4].

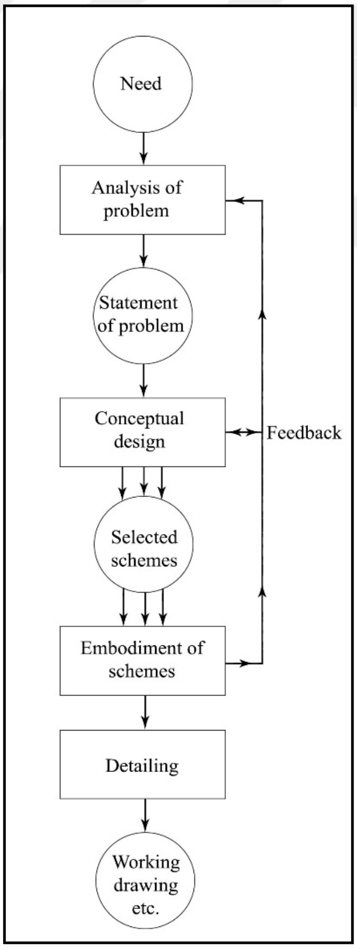


Figure 2.4. French's model of the architectural design process [24].

Bringing a different perspective to the design process models, French (1985) mentions two different phase models determined by circles and rectangles in his model (Figure 2.4). In this model prepared by French, those expressed with circles express the achieved phases, while those expressed with rectangles refer to the ongoing phases. French argues that it is not right to include the evaluation phase in any part of the process in a design process. Because the evaluation is a phase that is active in all rectangles in progress. French states that the design process begins with a "need". The "need", which is the first stop of the design process, is analyzed in the "analysis of the problem" phase, which is the first phase that needs to be worked on actively. French explains the ongoing phases that he described with rectangles in his model as follows: The phase in which the analysis of the problem is made is the most difficult process that involves the exact definition of the design problem. The analysis of the problem phase consists of determining the necessity to meet what is desired exactly as possible.

> The analysis of the problem is a small but important part of the overall process.The output is a statement of the problem, and this can have three elements:
>
> - a statement of the design problem proper,
> - limitations placed upon the solution, e.g. codes of practice, statutory requirements, customers' standards, date of completion, etc.,
> - the criterion of excellence to be worked to [24].

French states that the criterion of excellence in design is to evaluate the cost in the most reasonable way. Each new decision made after the design process is a factor that significantly affects the design cost, therefore, getting the right feedback throughout the design work is an important point to consider about the analysis of the design problem. The phase in which solutions are produced and the greatest demands are imposed on the designer is the conceptual design phase. French states that all aspects of the design must be brought together and important decisions regarding the design must be taken at this stage. At the phase of embodiment of the schemes, all the schemes created are examined in more detail and a final choice is made among the schemes. The resulting final product is usually a set of general arrangement drawings. This phase is in an iterative feedback loop with the conceptual design phase and may require re-evaluation of the conceptual design phase. French suggests that the detailing phase is the final stage in which the basic points that are likely to significantly affect the overall design structure are kept being decided. A poor quality work to be done in the detailing phase may reflect on the design as delay, expense, and failure [24].

Descriptive design process models are the design process models created to comprehend the design process itself and the structure of the system created. Besides, descriptive models have a more permeable structure than engineering models in terms of incorporating the relations of the system with the environment into the intellectual processes.

2.1.2. Prescriptive Models of Design Process

In addition to explanatory models that deal with the intuitive and conventional structure of design thinking, researchers have conducted some studies to create prescriptive models of the design process. These prescriptive models created are intended to enable designers to adopt newly developed models of study, rather than descriptive models. Prescriptive models usually offer a more algorithmic, systematic procedure to follow and are generally considered to provide a specific design methodology. These prescriptive models emphasized the need for more analytical studies before solution concepts are formed. The purpose is to ensure that the design problem is fully understood, that no important factor thereof is overlooked, and that the main problem is identified. Prescriptive models propose an alternative, basic structure to descriptive design models that include analysis-synthesis-evaluation phases [4].

Archer (1984) has developed a model, which is more detailed compared to early examples, consists of various inputs and outputs, has a feedback loop, includes the designer's education and experience and the interaction of other sources of information with the world outside the design process. Archer described this model in six stages [25]:

> Programming: establish crucial issues; propose a course of action.
> Data collection: collect, classify and store data.
> Analysis: identify sub-problems; prepare performance (or design) specifications; reappraise proposed programme and estimate.
> Synthesis: prepare outline design proposals.
> Development: develop prototype design(s); prepare and execute validation studies.
> Communication: prepare manufacturing documentation [25].

Archer summarized this process as dividing into three broad phases: analytical, creative and executive.

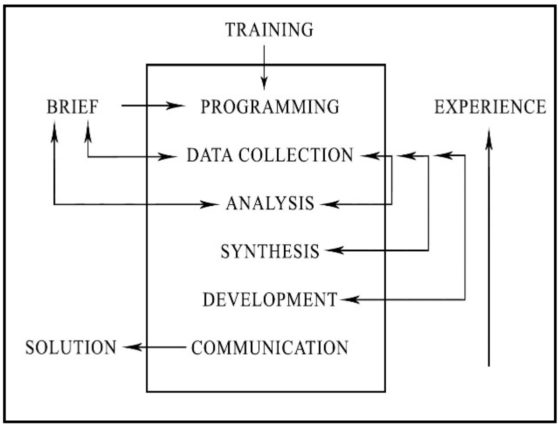


Figure 2.5. Archer's design process model [26].

He suggested that:

> One of the special features of the process of designing is that the analytical phase with which it begins requires objective observation and inductive reasoning, while the creative phase the heart of it requires involvement, subjective judgment, and deductive reasoning. Once the crucial decisions are made, the design process continues with the execution of working drawings, schedules, etc., again in an objective and descriptive mood. The design process is thus a creative sandwich. The bread of objective and systematic analysis may be thick or thin, but the creative act is always there in the middle [25].

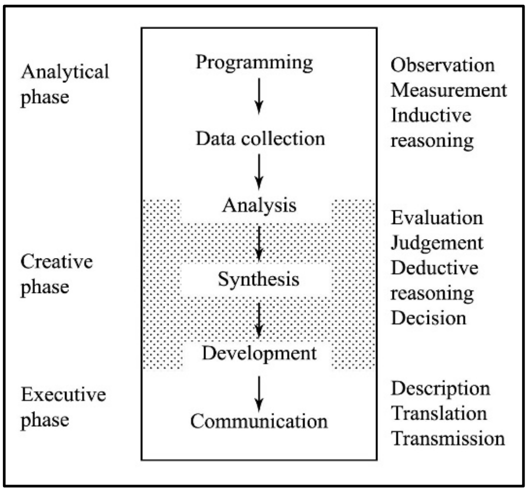


Figure 2.6. Archer's three-phase summary model of the design process [26].

Pahl and Beitz (1984) proposed a more detailed design process model in which the design process continues to preserve its general nature (Figure 2.7.):

> Clarification of the task: collect information about the requirements to be embodied in the solution and also about the constraints.

Conceptual design: establish function structures; search for suitable solution principles; combine into concept variants.

Embodiment design: starting from the concept, the designer determines the layout and forms and develops a technical product or system in accordance with technical and economic considerations.

Detail design: arrangement, form, dimensions and surface properties of all the individual parts finally laid down; materials specified; technical and economic feasibility re-checked; all drawings and other production documents produced [27].

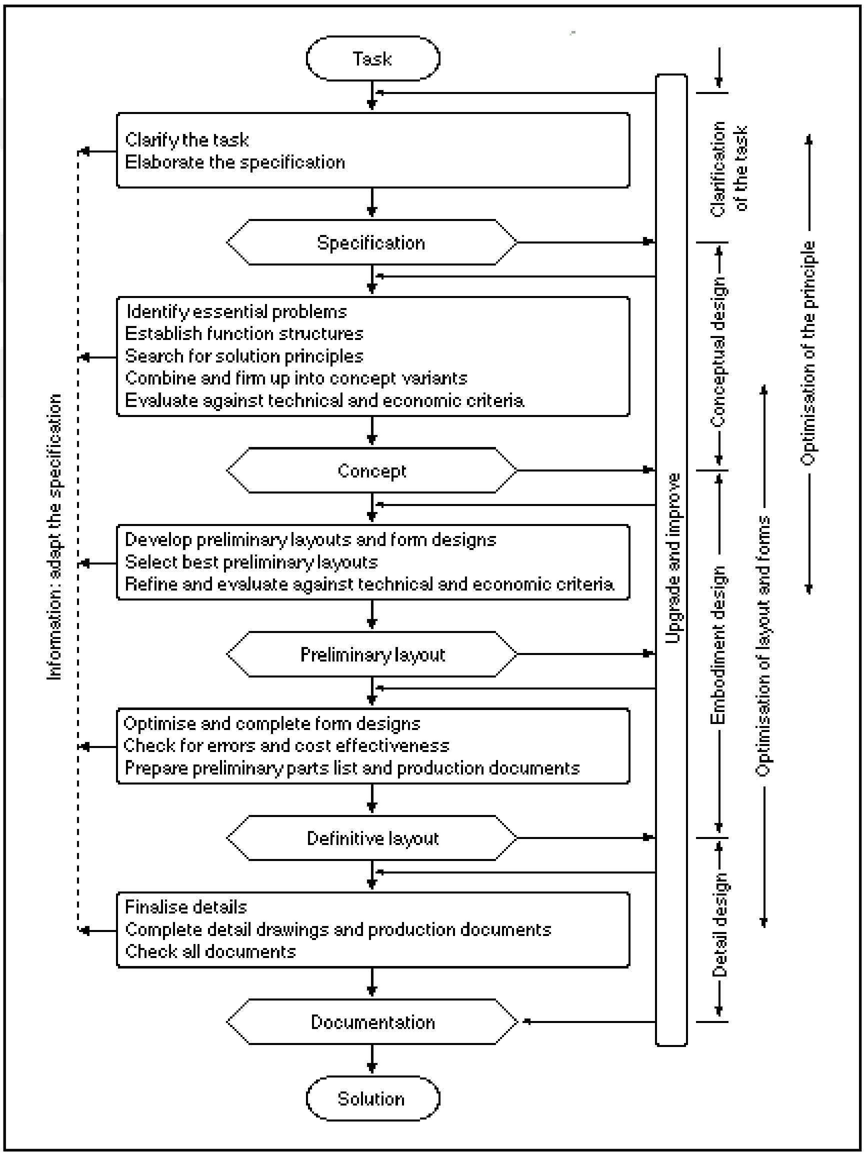


Figure 2.7. Pahl and Beitz's (1984) model of the design process [27].

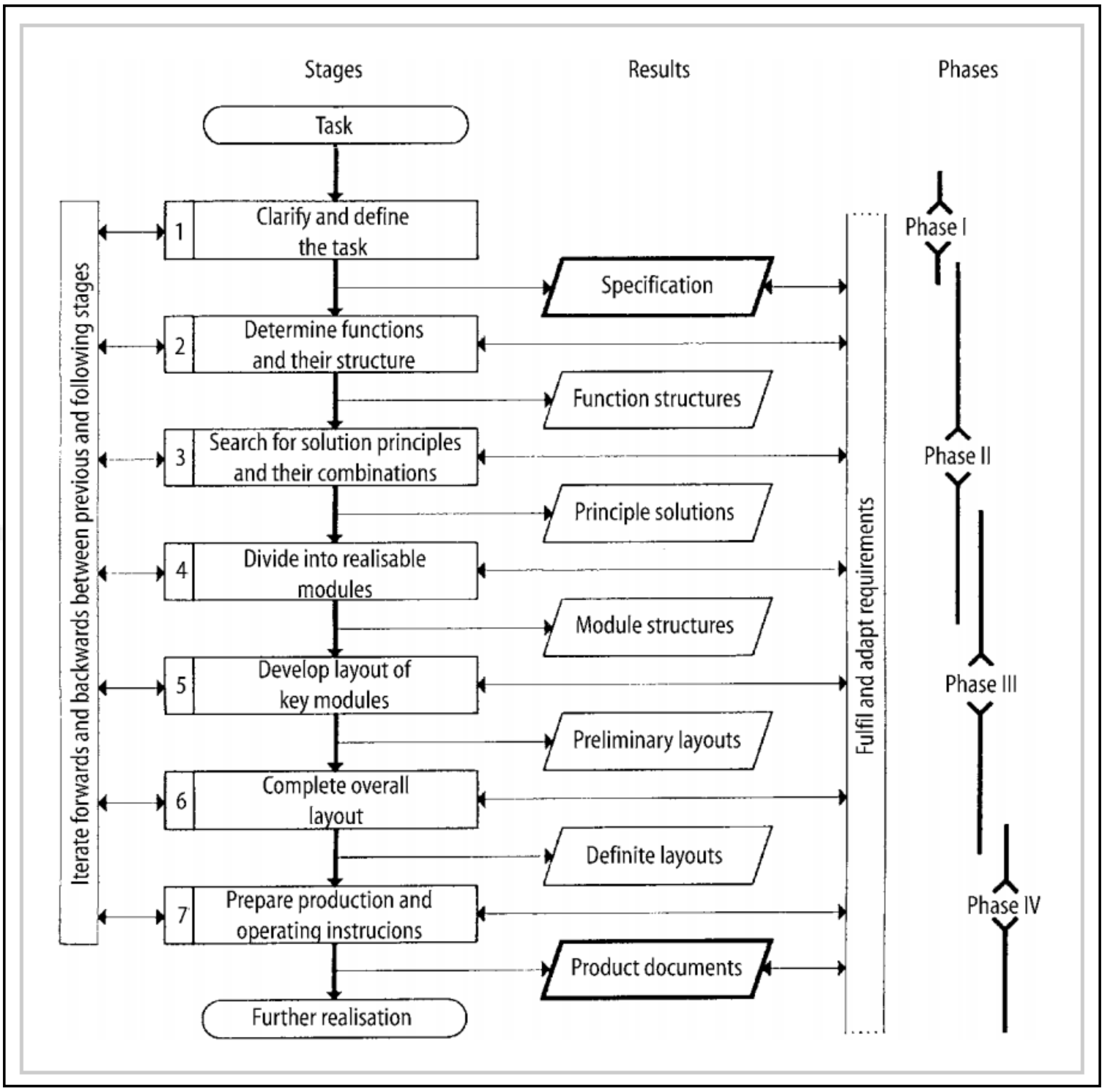


Figure 2.8. The VDI 2221 model of the design process [28].

In their guide named VDI 2221: Systematic Approach to the Design of Technical Systems and Products, the German Society of Professional Engineers (Verein Deutscher Ingenieure-VDI) has proposed a model that deems the design process as a part of product production, divides the design process into general working phases and makes the design approach transparent, rational and independent (Figure 2.8.). The output obtained in the first stage of this design process model is the specification. The specification is considered an important output throughout the entire process and is constantly reviewed and updated. In the second stage of the design process, the necessary functions and the design structure of the design are determined. In the third stage, a research is made for the solution principles in all sub-functions and these are combined as a basic solution in accordance with the general function structure. In the fourth stage, the process is divided into feasible modules and the structure of the modules is created. In the fifth stage, the layouts of the key modules are developed and a set of preliminary layouts are created. In the sixth stage, these are refined and developed as a precise layout, and in the seventh stage, the final product documents are produced. In the guide, it is emphasized that the prepared model is not a uniform model that follows one after another, as in all other design process models. The model tries to develop the design iteratively within the framework of a certain optimal shape [28].

> The VDI Guideline follows a general systematic procedure of first analysing and understanding the problem as fully as possible, then breaking this into sub-problems, finding suitable sub-solutions and combining these into an overall solution. This kind of procedure has been criticised in the design world because it seems to be based on a problem-focused, rather than a solution-focused approach. It therefore runs counter to the designer's traditional ways of thinking [4].

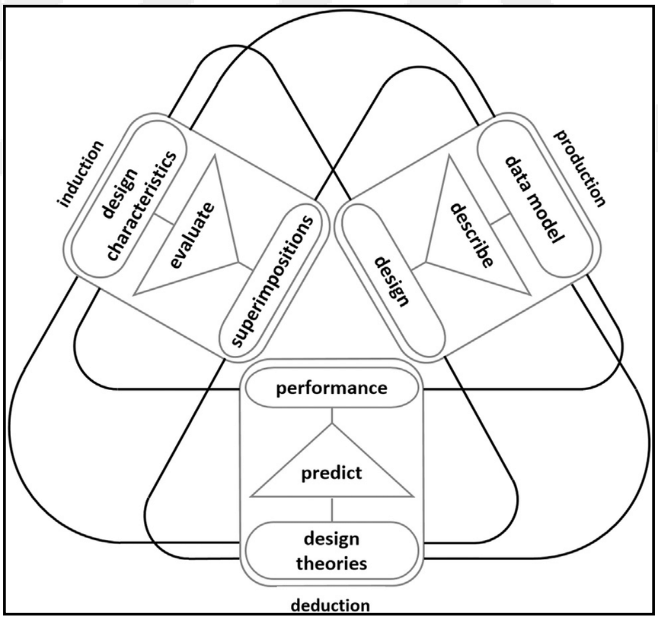


Figure 2.9. March's model of the design process [29].

March (1984) proposed a more radical model of the design process, describing the solution-oriented nature of design thinking (Figure 2.9). March argued that traditionally understood forms of reasoning apply logically only to evaluative and analytical types of activity in design. He stated that the type of activity particularly associated with design is the synthesis, which is not a generally accepted form of reasoning. The model of March has a nature of hypothesis on what the synthesis may be, which is central for the design. Based on the way designs are produced, March preferred to name this model “productive reasoning”. In this model, stage one, the productive reasoning, utilizes some presuppositions about the types of solutions to produce or explain a proposal statement and a design proposal. From this proposal and from an established theory, it is deductively possible to analyze or predict the performance of the design. From these anticipated performance characteristics, it is inductively possible to evaluate other assumptions or possibilities leading to changes or improvements in the design proposal [30].

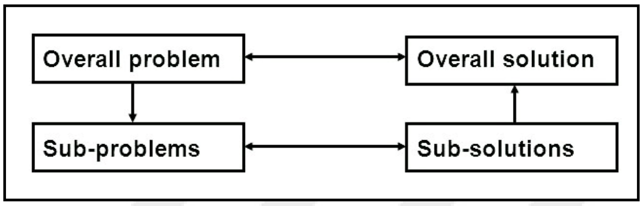


Figure 2.10. The symmetrical relationships of problem/ sub-problems/ subsolutions/ solution in design [29].

Prescriptive design process models, which deal with the whole operation of the systems in an algorithmic and systematic way, aim to ensure that every stage of the design is meticulously incorporated into a specific system. Thus, it is possible to achieve the final design product within the framework of the model created without skipping any details.

The systematization and rationalization processes envisaged in the organization of production and management in the early 1900s initiated the first steps of a transformation that will take place in almost every area of life with the effect of modernism. The systematization and rationalization that spread all over the world, especially with the social and economic changes experienced with the World War II, affected almost every sphere of life, including the daily dynamics thereof. Like almost every sphere of life, design was also tried to be included in these systematic processes and the design process models were created as a result of these efforts. These design process models, which are created by dividing design into specific processes, are categorized as prescriptive models with a precise and uncontroversial structure and descriptive models with limited flexibility.

Compared to descriptive models, prescriptive design process models can be seen as design process models that cannot be possible to be used in the fields of architecture and art because they are closed systems, but can be used to produce specific design products for specific problems with the help of computation. However, a design process created with a descriptive model represents processes that are intuitive except computation and in which each stage is in constant interaction with itself and with other stages, and where the decisions made and even the final product can be evaluated at certain stages. While prescriptive models offer a specific and almost precise design methodology, descriptive models, on the other hand, are close to intuitive and conventional processes.

The design process models created with an effort to systematize the design are examined in this part of this thesis. In the further chapter of this thesis, the problem solving methods in architectural design process will be discussed.

2.2. PROBLEM SOLVING IN ARCHITECTURAL DESIGN PROCESS

The main problem in all the design process maps examined in the preceding section is that the problem requiring design action cannot be completely solved at the beginning of the design. Thus, the design process models created by the researchers to make the design process more defined suggest that the designer continually goes back to previous design process phases and completes the design in an iterative feedback loop. Researchers who approach to design thinking from a more systematic perspective and produce problem-oriented and solution-oriented design process models have tried to fully analyze the complex structure of the design process.

Bruce Archer stated that unsatisfactory conditions that may hinder the achievement of the design objectives should be regarded as a problem and therefore design is a target-driven problem-solving activity. Archer states that a "problem space" must first be defined in order for the design process to begin [31].

Newell and Simon explain the problem space, with a perspective similar to Archer's approach, as a field where problem solving activities take place. Newell and Simon argue that the process of problem solving requires collecting information that can directly affect the design, such as what, under what conditions, through which tools, methods and processes, starting from which initial information and which resources, is the information requested. The designer classifies all this information he collected in the most appropriate way to the target he has determined in the problem space. There are two different types of problems that are addressed by Herbert Simon and Alan Newell in problem solving studies. They define the first type of problem as a well-defined problem. In well-defined problems, the solutions are known beforehand, allowing problem solving to have a definite starting point. Besides, in a well-defined problem, the solution paths can be clearly recognized in the definition of the problem, thereby they can provide the problem solver's operators with a direct start to the solving process. Another type of problem in design is ill-defined problems. Ill-defined problems are the problems where the goals, problem solver's operators, and the current situation are not certain and do not have sufficient inputs to reach a satisfactory solution [32].

According to Jonassen (2000), the problem solving process is to find the unknown. Due to the limited understanding of problem solving processes by humans and the wide range of problem solving activities, problem solving is considered to be the most important cognitive activity in a daily and professional context. Jonassen argues that the problem has two critical features: the first of these is that the problem is an uncertain entity, and the second is that the human cannot have all the social, cultural or intellectual values required to find the unknown. The design problem is defined by researchers as the most complex and poorly structured problem due to the identification of unknown number of objectives, the uncertain solution path, and the need for integration into multiple information fields. Designers should structure the problem by defining the nature of the work that will meet undefined requirements. Designers must set up personalized systems for evaluating their products as criteria for acceptable solutions are not always clear [33].

Kalay (2004) defines design as a cyclical relationship between two paradigms: design as problem solving in which the designer tries to find solutions to ill-defined problems, and design as a puzzle, a discovery process in which given parts are synthesized into a new and unique whole [34].

Lawson defines the design process as creating a negotiation environment between problem and solution through analysis, synthesis and evaluation activities [5]. However, not all the data and features required to solve a design problem can be determined clearly especially in the architectural design process. Wagter and Vries (1991) argue that the architectural design process has three features that differ from all other design processes. Wagter and Vries suggest that the first feature that distinguishes the architectural design process from all other design processes is the fact that the architectural design process does not have a good structure. A designer can create solutions to potential problems by adhering to the restrictions in the design process, but cannot identify a set of steps that will lead to a successful solution in a design process. The second feature is that the architectural design process is open-ended. During the design process, a design can always be developed until the deadline or budget limit is reached. Restrictions and improvements applied to the design will positively or negatively affect other aspects of the design process. The applied restrictions, themselves, can also be improved. Therefore, the design process always continues even if the deadline or budget limit is reached. The third feature is that the designer does not have a definite and fixed starting point at the beginning of the design process. At the beginning of the design, the designer must make a series of evaluations, including the assumptions that may change at the end of the design and the phases of the design [35].

The design process models developed starting from the 1960s have provided a set of working methods to solve design problems and organize the structure of the design process. The idea of underlying the conduct of these studies is gaining the ability to intervene, in the early stages of design, to the situations specific to design such as multidisciplinary, complexity, unpredictability, and hierarchical order that are experienced in the design process.

Multidisciplinary structure

Design is a wide range of activities consisting of two different extremes: engineering and art. The multidimensional nature of the design requires the designer to collaborate with teams from many different disciplines. Each individual in such teams plays an important role in design within the period from the conceptual design process to the technical implementation of the ideas [5]. Lattuca and Knight (2010) argue that interdisciplinarity is the process of evaluating a subject, which is too comprehensive or too complex to be handled by a single discipline, from a more comprehensive perspective by using the perspective of a different discipline [36]. The design of a simple building concerns many different disciplines such as architecture, static, mechanics, electricity, infrastructure, landscape design, etc. The interdisciplinary nature of architectural design is likely to play a role in the problems that arise in design studies. These problems can be grouped into complexity, unpredictability, and hierarchical order.

- Complexity:

Alexander points out that the functional problems that have become more complex make it more and more difficult for the designer to grasp what the design problem is [37]. Jones argues that the information technologies, changing economic, cultural, educational characteristics, lifestyle preferences of growing community populations, and many other characteristics of societies cause design problems to be addressed at larger scales and become more complex each time [38]. The design, in which the designer aims to bring an important situation that needs improvement to an improved state with insufficient information resources, is considered a kind of problem solving activity. Studies in cognitive science and cognitive psychology fail to provide sufficient resources for explaining the complex design problem that the designer struggles with, as they are successfully applied only to well-defined problems. Due to the infinite number of improvements that can be applied at minimum and optimum level on the current situation in design problems, it is not clear if there could be a well-defined problem space and a final improved situation. Almost no improvement can be considered the ultimate or even the most perfect situation ever discovered [39].

- Unpredictability:

Design is an exciting, creative and unpredictable process as not every project worked on can achieve success with similar processes, and every project is unique [40]. Design processes are inherently complex. The complexity results from different factors such as the interdependencies and coordination needs among the tasks performed by different design disciplines, the iterative nature of the design process that designers seek to satisfy solutions, and unpredictable nature of design criteria [41]. All the steps that are tried to be predicted in the process aspect of the design can be excluded from the design process if the design is deemed inadequate and/or inappropriate in the later stages. As with well-defined problems, the fact that the process phases and the problem solution are not final is likely to cause the designer not to have a sufficient prediction about the product to be obtained at the end of the design.

- Hierarchical order:

Herbert Simon (1969) argues that the complex systems are included in the hierarchical systems set and the components of the complex systems perform certain sub-functions that contribute to the general function of the system. As the sub-function is insufficient for the whole system, each sub-function in the system can manage the task assigned to it alone up to a certain level. In cases the sub-function fails to perform the next task, it needs a parent function more capable than itself [42]. Each of these concepts incorporated in the structure of the design and causing design problems closely concerns the other. These concepts are utilized in design process models to make the information underlying design thinking processable and to discover computable aspects of design.

These concepts refer to the problems faced by an existing design problem. When the methods used to achieve a design product are considered within the framework of the dynamic structure of life in a design problem, it becomes more complex due to increasing and developing needs. While increased complexity makes predicting design problem solutions more complex, it increases the hierarchical relationships between solution processes. These difficulties encountered in solving design problems require the design processes to be handled with new tools and technologies.

Experiencing various difficulties in responding to problems due to the transformation of the studies required to meet the increasing and changing needs over time in architectural design processes into more complex, unpredictable and multidisciplinary structures can be shown as the reason behind the consideration of design as a specific systematic and hierarchical process. Design researchers, who approach to these systematic and hierarchical structures by creating design process maps, aimed to use also in the field of design the similar productive activities, which are a reflection of the modern world order. Thus, it can be discussed that they aimed to create optimal solutions to the design, which can be considered a problem-solving activity. In the third chapter of this thesis, the functioning and field of use of AI technologies, which are considered to be more successful than designers in solving certain problems will be discussed.

3. ARTIFICIAL INTELLIGENCE

The researchers have suggested many different definitions for artificial intelligence: Bellmann (1972) suggests that AI can be defined as "activities such as automation of activities that we associate with human thought, decision making, problem solving, learning..." [43]. Haugeland (1985) defines AI as "an exciting new effort to make computers think that have minds in real terms" [44]. Kurzweil (1990) defines it as "the art of creating machines that perform the required functions" [45]. Rich and Knight (1991) define it as "the effort of ensuring to make computers capable of doing things at which humans are better for now" [46]. Charniak and McDermott (1985) define it as an "investigation of mental abilities by use of computational models" [47]. Winston, (1992) defines it as "examining the computations that enable perception, causes, and actions" [48]. Schalkoff, (1990) defines it as "a field of study that aims to explain and imitate intelligent behaviors in terms of computational operations" [49]. Luger and Stubblefield (1993) define it as a "branch of computer sciences engaging in automation of intelligent behaviors" [50].

Roger C. Schank (1987) argues that there is no single final definition of AI, as what AI is depends on the purpose of the researcher using it and the methods used in creating AI models. Because intelligence is a term that is very difficult to define. In addition to this uncertainty about the definition of artificial intelligence, researchers who develop AI technologies have jointly adopted the objectives of creating an intelligent machine and finding the nature of intelligence. Schank argues that in order to be able to define an entity as intelligent, it must have certain features: communication, internal knowledge, world knowledge, intentionality, and creativity. [51].

Stuart J. Russell and Peter Norvig (2009) points out in their book 'Artificial Intelligence: A Modern Approach' that AI definitions can be categorized under 4 headings: systems that think like human, systems that act like humans, systems that think rationally, systems that act rationally [52].

- Systems that think like human:

In order to create a thinking entity, an approach like human mindset is required. For this purpose, the mindset of the human can be examined with psychological experiments and/or introspection method, and as a result of the examination this structure can be transferred to an entity that is able to think [52].

- Systems that act like humans:

Turing defined intelligent behavior as the ability to deceive an entity, which questions itself, in all cognitive tasks and to have a human-level of performance [53]. Turing proposed a test, known as the Turing Test, in which a subject and interrogator can communicate through a terminal to investigate whether a system is intelligent. In this test, it is determined whether the subject (machine) passed the test according to the ability of the interrogator to understand whether the subject is a human or a machine.

- Systems that think rationally:

A rational system is based on a logical approach. This logical approach aims to make inferences from situations, to create solutions with the help of the rules it creates within the framework of inferences.

- Systems that act rationally:

Systems that think rationally determine a purpose for themselves with the inferences they have made with this method and try to achieve this goal. This situation causes systems to behave in accordance with the purposes they believe in. In AI, it is not enough alone for a rational agent to make correct inferences. In such cases, AI based on the laws of scientific thought has the advantage to produce solutions in some situations that are not gained through inferences [52].

3.1. A BRIEF HISTORY OF AI

3.1.1. The Gestation of AI 1943-1956

The foundations of systems known today as AI were first laid in 1943 by Warren McCulloch and Walter Pitts, who created artificial neuron models with their approaches based on knowledge of the basic physiology and function of neurons in the brain, characterized by the transition to the "on" state in response to stimulation by a sufficient number of neighboring neurons in which each neuron is qualified as on or off, and the formal analysis of Russell and Whitehead's propositional logic and Turing's computational theory. They proposed that a computable function can be computed with an artificial network of neurons connected to each other with logical connectors, and that these artificial networks can perform the learning action similar to the biological structure of human.

- In 1949, Donald Hebb developed a simple set of updating rules that could change the connecting forces between neurons, making it easier for neurons to learn by themselves [52].
- The Turing test is a test that first involves investigating whether it is logically possible to say that a machine can think in an article titled Computing Machinery and Intelligence by the famous British mathematician and computer scientist Alan Turing in a philosophy journal named Mind in 1950. According to the Turing test, the machine and a human volunteer are hidden somewhere away from the interrogator's field of vision. The interrogator tries to determine which one is a human and which one is a computer by simply asking questions [53].
- In 1951, Marvin Minsky and Dean Edmonds, who were graduate students in Princeton mathematics department then, set up the first neural network computer, which they named SNARC (Stochastic Neural Analog Reinforcement Calculator), which consists of 40 artificial neural cells [52-54].
- While continuing his studies at Dartmouth College, John McCarthy, a Princenton-based researcher, held a 2-month workshop in Dartmouth with the support of researchers such as Minsky, Claude Shannon and Nathaniel Rochester in 1956 with a group of 10 US researchers as Trenchard from Princeton, Arthur Samuel from IBM, Ray Solomonoff and Oliver Selfridge from MIT, who are interested in automata theory, neural networks and intelligence studies. John McCarthy was the first name who used the term "Artificial Intelligence (AI)" there [52-55].
- In 1956, Alien Newell and Herbert Simon stood out amongst other names in the workshop with the AI program, which they developed, named Logic Theorist (LT) that imitates the problem-solving skills of human and has non-numerical thinking ability. This two-month workshop held at Dartmouth College ensured that the studies, which were considered a new field of research and did not have a defined name until that day, were collected under the title of "Artificial Intelligence (AI)”,,

which was introduced by McCarthy. This workshop also provided the opportunity for those, whose names will be frequently mentioned in the next two decades of Artificial Intelligence (AI) studies, to meet each other and share their studies with each other [52].

3.1.2. Golden Age of AI

This era is perceived as the beginning of developments that can no longer be prevented, where AI studies have gained a great momentum with the support of various institutions and organizations. Developing systems, such as disease diagnosis in particular, AI experts have laid the foundations for a long and exciting adventure, of which outcomes are eagerly anticipated today. This period also covers the years in which artificial neural network (ANN) studies were conducted.

- In 1957, Frank Rosenblatt introduced Perceptron, an algorithm for the supervised learning of binary classifiers in machine learning. Rosenblatt proved the perceptron convergence theorem and showed that the learning algorithm can adjust the coupling strengths of a perceptron matching with any input data, provided there is such a match [56].
- In 1957, the General Problem Solver program, which was created by Newell and Simon after Logic Theorist, embodied the protocol of systems that think like human by imitating the human problem-solving protocols within the limits of the program, the sub-goals of the solution, and the possible actions that are likely to take place [52].
- In 1957 H. Gelernter and N. Rochester created the Geometry Theorem Prover, which has a semantic structure, using axioms that are clearly represented, just like in Logic Theorist. H. Gelernter and N. Rochester discovered as a result of their studies that there could be more than one possible logic path and added to their program the ability of creating a numerical representation of the diagram. Thereby, the program had the ability to check the accuracy of the diagram for a specific situation before proving anything.
- 1958 was a brilliant year for John McCarthy. First, McCarthy invented LISP, the second oldest language which is commonly used currently [57]. He then created the

Advice Taker, a hypothetical program that can be regarded to be the first full AI system, and introduced the Advice Taker with an article titled Programs with Common Sense [58]. McCarthy designed the Advice Taker to use knowledge to seek solutions to problems as in Newell and Simon's Logic Theorist and H. Gelernter and N. Rochester's Geometry Theorem Prover. However, unlike all other studies, the Advice Taker, having supported by new axioms while the existing axioms continued, was able to develop itself to solve problems in new areas without the need for reprogramming, and thus to have general knowledge of the world.

- In 1961, James Slagle, using the SAINT (Symbolic Automatic Integrator) program, on which he worked in the thesis he prepared with Marvin Minsky at MIT, has succeeded in solving college freshman level math problems [59].
- In 1961, Tom Evans's program called ANALOGY succeeded in solving geometric analogy questions of IQ tests, in which the elements that are similar to each other are defined [60].
- In 1962, Minsky and Papert defined limited areas requiring intelligence as micro worlds (block worlds) [61].
- J. C. R. Licklider introduced a series of notes that will later become known as the Internet, which is a worldwide medium for collaboration, information sharing, dissemination and interaction among individuals, regardless of geographic location [62].
- 1964 - Bertram Raphael's SIR (Semantic Information Retrieval) succeeded to accept input expressions in a very limited English linguistic subset and answered questions to them [63].
- Created in 1964, the STUDENT program has reached a level that can use the natural language well enough to solve algebraic word problems [64].
- In 1965, Lotfi A. Zadeh introduced the Fuzzy Logic theory, which, unlike classical logic, uses approximate values instead of absolute values, in which the values have a degree of membership in the range of [0-1] and which can express each logical expression within these features [65-66].
- In 1965, D. Engelbart developed the computer mouse as a way of implementing NLS (ON Line System) hypertext and common workspace [67-68].

- ELIZA, an early natural language processing program created by J. Weizenbaum in the MIT Artificial Intelligence Laboratory in 1965, was able to have a human-level conversation by simulating a psychotherapist [69-70].
- Developed by Feigenbaum et al. at Stanford University in 1967, the DENDRAL program is the first AI assistant designed and put into use. DENDRAL is the first program to bring the knowledge of an expert to the computer environment by means of AI. The main purpose underlying the creation of the DENDRAL program was to analyze the data to be obtained from a mass spectrograph with a chemist's logic, by developing a chemistry program that would accept physical data and generate sufficient hypotheses to explain the data. DENDRAL is the pioneer of next generation AI programs called knowledge-based systems. Feigenbaum et al. made observations by meeting with chemists to analyze specific situations such as how chemists think, how they answer questions, and how they approach problems. As a result of these observations, they developed the method of representing specific behaviors in a large database with DENDRAL. The significance of DENDRAL was inarguably that it is the first successful knowledge-intensive system [52-68].

The time span from the development of AI technologies to DENDRAL was a general-purpose search mechanism, called weak methods, that attempted to put together basic reasoning steps for problem solving that came up in the first decade of AI research [52].

- Created by Joel Moses at MIT in 1967 and based on the studies of James Slagle, MACSYMA is the first knowledge-based AI math assistant program designed to solve problems in symbolic algebra [71].
- Mac Hack VI, a later version of Mac Hack (The Greenblatt chess program) developed by Richard D. Greenblatt at the Massachusetts Institute of Technology in 1967, is the first chess program that played in human tournament conditions, ranked in chess rank, and won against a person in tournament play [52-72-73].
- In 1969, ARPANET (Advanced Research Projects Agency Network) was established, which forms the basis of today's internet concept, where multiple different networks can come together and interact under a general network [68].
- In 1971, artist Harold Cohen created AARON, a computer program that creates original artistic images [74].

3.1.3. The First Winter of AI 1974-1980

In the 1970s, AI came under criticism and financial disruption. AI researchers could not appreciate the difficulty of the problems they faced. Researchers' optimistic attitudes raised expectations enormously, and once the promised outcomes did not come true, the funds allocated for AI were withdrawn over time. Despite all the challenges AI faced in the late 1970s and early 1980s, new ideas were discovered in logical programming, reasoning, and many other areas [52].

- In 1974, Feigenbaum and others in Stanford launched the Heuristic Programming Project (HPP). Feigenbaum, Buchanan and Dr. Edward Shortliffe developed MYCIN, which can diagnose blood infections to study to what extent the new methodology of expert systems can be applied to other areas of human specialization. MYCIN was able to perform significantly better than young doctors as well as some specialists [52].
- In 1978, Simon won the Nobel Prize in Economics for his theory of bounded rationality known as "satisfactory", which is the cornerstone of AI (and human behavior); Moravec's car is the first computer-controlled autonomous vehicle [68].

3.1.4. The Rise of Expert Systems and The Knowledge Early 1980s

This period can be shown as the years when knowledge-based systems showed success in commercial activities and the use of AI studies in industry gained importance. During this period, efforts to use information to reason and solve cases that typically occur in narrow specialties have increased.

- In 1981, the Japanese Ministry of International Trade and Industry announced that it had allocated $850 million for the fifth-generation computer project. Their goal was to build machines that could sustain conversations, translate languages, interpret pictures, and reason like humans [54-68].
- In 1982, Rl, the first successful commercial expert system, came into operation at Digital Equipment Corporation. The system, which saved the company annually $40 million until 1986, represents a great success under then conditions [54-75].

- In 1982, Newell et al. created SOAR, an architecture for general intelligence, and the United States announced that it started a strategic information processing project to achieve AI goals [68].
- In 1985, R. Brooks and his team created Allen, the first of autonomous reactive robots, and Brooks' study began to focus on the engineering of intelligent robots to work in unstructured environments and the understanding of human intelligence by building humanoid robots [68].

3.1.5. The Second AI Winter

The late 1980s can be shown as the period in which AI studies experienced the most severe financial and security shocks. The fact that AI-based systems are being seen as a balloon in the business world and industry is shown as one of the biggest factors of this. The perception that AI systems that can only succeed in certain areas and contexts have high maintenance costs has shaken the trust of the business world in these systems. In the late 1980s, the Strategic Computing Initiative decided to stop funding AI. Thinking that AI has no future, DARPA (Defense Advanced Research Projects Agency) decided to invest the funds cut from AI studies in other fields of work [54-68].

3.1.6. Rising of AI 1993-Present

AI studies, which started to gain momentum again in parallel with the increasing computer power, came to the agenda again in this period. There may be a rebound in AI studies, with AI focusing on subsections that focus on specific problems or approaches. In the past part of this period, in addition to robotic studies, the studies that push the limits of human intelligence and can compete with human intelligence have come to the fore in AI studies. Besides, the focus of thoughts were concentrated on the ethical problems of AI studies and beneficial uses of AI.

- In 1997, Deep Blue defeated chess champion Garry Kasparov.
- In 2001, Berners-Lee et al. started working on the Semantic Web, an international effort to realize global commercial, scientific and cultural data exchange on the World Wide Web (WWW) using logic, inference and action AI techniques [68].

- In 2003, DARPA launched the "LifeLog" project, which will become an ontology-based system that captures, stores and makes accessible the flow of experience and interactions of a person in the world to support a wide spectrum. The purpose of the LifeLog concept was to track the subjects of an individual's life in terms of events, situations and relationships [76].
- In 2011, IBM's Watson computer defeated Rutter and Jennings, the champions of the television competition "Jeopardy!".
- In 2015, the Future of Life Institute organized the "AI Safety Conference" in Puerto Rico with the participation of many AI researchers to discuss the possible ethical problems of AI studies [77].
- In 2016, Google DeepMind's AlphaGO won the go match against Lee Sedol with a score 4-1.
- In 2017, the Future of Life Institute organized the Asilomar Conference on Beneficial AI at the Asimolar Conference Area in California.

3.2. PROBLEM SOLVING ABILITY OF AI

Before addressing the problem-solving capabilities of AI, it is necessary to address a number of anecdotes about why AI's problem-solving capabilities are needed today.

Max Tegmark (2017) points out that the origin of today's complexity is that the Earth began to transform into a planet with the occurrence of the Big Bang, where a planet began to take form from a dust cloud. New systems that took form during and after this process, which was already sufficiently complex alone, increased the level of complexity even more [77]. Nick Bostrom (2014) states that the emergence of Homo Sapiens led to a great improvement in brain size, neurological organization, and cognitive ability. As a result of such improvement, humans have gained the ability to think abstractly and complexly, to convey their thoughts, to collect and transfer information as a cultural heritage. The improved frame of mind and the partial technology helped humans to reach the Savannah from the rain forests; the sedentary life was adopted with the adoption of agriculture, and the increasing population accordingly caused more problems and more ideas to arise [78].

Constantly changing and evolving needs of human beings since the early ages make life, which is already complex and requires struggling with many problems, more complex. With the changes in the structure of intellectual and philosophical thinking and increasing scientific discoveries, the ever-developing scientific disciplines need to cope with the growing problem scale and the human being trying to adapt to the increasing problems and complexity, it is necessary to work to improve the approach to problems as well as to develop the method of approaching the problems. There is a strong need for an interdisciplinary approach to solve the world's problems that are becoming more complex and dynamic day by day.

Descartes mentions 4 principles he determined for himself in his work named “Discourse on The Method” in which he determined the methods of rationality. The first of these principles is to deny the facts that they do not know with certainty and to defend skepticism against these facts, the second principle is to reach a reasonable but imprecise, approximate conclusion by taking all the difficulties to be dealt with as smaller parts as possible; the third principle is to deal with problems that are more difficult in terms of shaping the structure of thinking, starting with the simplest of the problems that are fragmented, as much as possible, and finally, doing complete counts and overall checks on all aspects to make sure nothing is missed [79].

According to the World Complexity Science Academy, complexity can be expressed in terms of the number of individual components in a system, the amount of linkage or interconnection within a system, the ability of individual components to adapt over time, and the degree of diversity or variation between individual components of a system [80].

The principles introduced by Descartes are based on rationality, skepticism, reductionism and reasoning. These principles, which were adopted until the beginning of the 20th century, have been able to show the adoption of complex systems with changes in the intellectual thought structure and the history of scientific thought, and that problems can be handled as a whole or in parts. The complex systems approach is beneficial in developing technologies based on similar principles that emerged in the context of biological and physical sciences and then social sciences in order to understand the functioning of complex structures in nature.

Since complex systems are a detailed subject on their own, in this thesis the definitions and general features of complex systems only will be discussed. There are many different definitions of complex systems: Melanie Mitchell (2009) defines a complex system as follows [80]:

> …a system in which large networks of components with no central control and simple rules of operation give rise to complex collective behavior, sophisticated information processing, and adaptation via learning or evolution [78].

According to Melanie Mitchell (2009), complex systems distinguish from other systems by features [80]:

- Complex collective behavior:

A complex system consists of networks of individual components with a central control system that follow a simple set of rules without any manipulators.

- Signaling and information processing:

Complex systems generate a set of signals and information belonging to the internal and external environment and use these signals and information.

- Adaptation:

Complex systems adapt through learning or evolutionary processes, following an evolutionary route on which those with optimum fitness can survive.

Herbert Simon argues that a complex system is formed by the combination of a large number of different parts, conforming to a non-simple and certain set of rules. A complex system is formed by the combination of systems that have their own subsystems. Simon argues that the most important common feature of complex systems is hierarchy. According to Simon, complex systems have a certain hierarchical order and this order is network systems that operate with a top-down or bottom-up interaction [81].

The new structure formed by the combination of the components of many different structures in complex systems is called emergent properties. It is one of the prominent features of emergent properties that the certain features of the sub-parts of a certain structure constitute a whole and that this whole includes features that cannot be obtained separately from each part. Emergent properties do not always create an unexpected or highly specific structure. On the contrary, this new structure can come out within expectations [82]. The new structures formed by the combination of small system parts in complex structures such as ant colonies, the structure of the brain, and cities, and that are outside certain rules, can be shown as examples of emergent properties.

For understanding and analyzing a complex system, two methods are referred to. The first method is Descartes' principle of simplification, the second of the four principles addressed to above. The system is divided into simple parts as much as possible and mathematically abstracted. Simple mathematical parts are analyzed and universally interpreted. However, if a result cannot be reached using these mathematical abstractions, a simulation of the system is made, and theories of statistical information about the system can be produced. As a result of the analysis, a possible explanation of the subsystems of a complex system can be obtained.

While current AI researches aim imitating the human brain and intelligent creatures, existing AI algorithms can be considered successful in finding a solution to a specific problem. However, in order for an AI set to improve and become more intelligent, able to produce solutions to larger and more complex problems, it is desired to be used in the analysis of complex systems.

While the increasing population, developing technologies and developing social relations increase the amount of data and knowledge in the world, the newly emerging problems that need to be solved are parallel to this increase. Human brain, social networks, insect colonies, economy, and even the internet can be portrayed as a complex system [80]. Design, just like a complex system, consists of subsystems that follow certain set of rules, internal and external information is processed, and a final and most suitable product is aimed. The resulting product, on the other hand, is not a coincidence, but a product of a processual work.

Although solving problems is an important issue for AI, time and cost performance can be shown as the most significant factor in attaching importance to AI. The differences between the human brain and an artificial system will be discussed in the following chapters. Although the human brain is superior in these differences, AI is more capable than the human brain in computation and analysis. However, the main reason underlying this is the fact that the human brain can perform multiple functions simultaneously. The development of AI systems makes it possible to create low-cost and time-saving systems in computing and implementation.

3.2.1. Artificial Intelligence Agents

An AI agent is a software entity that can affect the environment by collecting various data with its own effectors (software, sensors, cameras, robotic components), just like humans perceive and collect data about their environment with various organs (Figure 3.1.) [52].

David L. Poole and Alan K. Mackworth argue that the inputs and outputs of an agent are as follows (Figure 3.2.): “prior knowledge” about the agent itself and the environment in which it is located, environmental interaction history including “observation” of the current environment and “past experiences” of past actions and observations, “goals” to be achieved and “ability” of the agent to fulfill actions [83].

David L. Poole and Alan K. Mackworth argue that the inputs and outputs of an agent are as follows (Figure 3.2.): “prior knowledge” about the agent itself and the environment in which it is located, environmental interaction history including “observation” of the current environment and “past experiences” of past actions and observations, “goals” to be achieved and “ability” of the agent to fulfill actions [83].

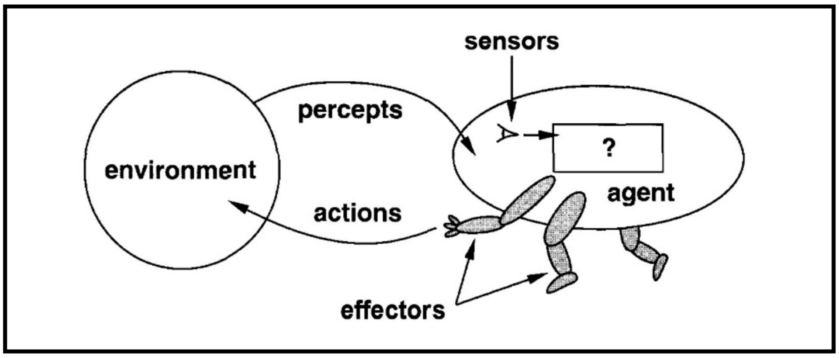


Figure 3.1. Intelligent agent [52].

Intelligent agents have two prominent features: rationality and autonomy. The autonomy of a intelligent agent includes the control of the wrong or incomplete information that the agent acquired through learning and the actions that may affect its success, with its own control mechanism [84].

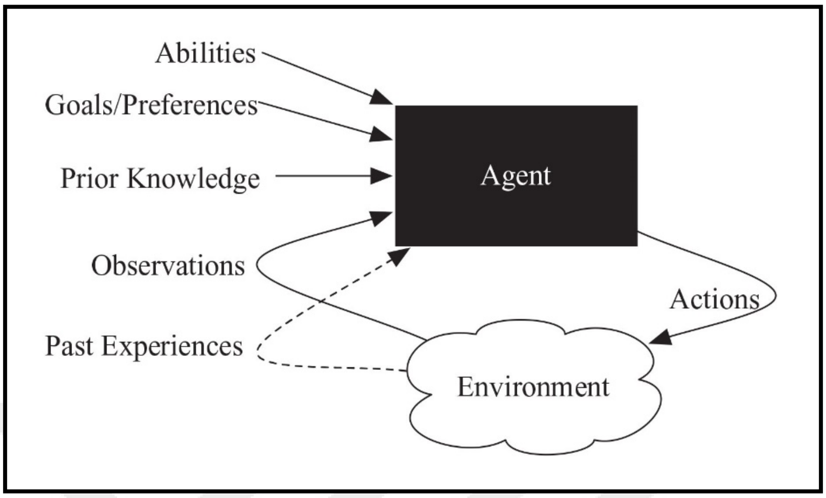


Figure 3.2. An agent interacting with an environment [83].

An agent is autonomous to the extent that its preference of action is based on its own experience but not on the environmental knowledge built by the designer [52]. A rational agent is the AI agent that does the right thing and is used to bring out optimum results in general acceptance. Hence, an AI agent is defined as a rational agent. The concept of rationality, which is defined among the "features that make a machine intelligent" in AI definitions and properties, represents a rational agent, and arguing that an unreasonable agent is intelligent contradicts these definitions.

David L. Poole and Alan K. Mackworth (2010) define rational agents as purposive agents. The formation formed by the combination of purposeless agents is called nature, and agents with purpose in nature follow a certain route to achieve their goals. The goal of a purposeful agent is actually representing the goal of its designer. Agents deal with the of the world (environment) as long as they have a goal, whereas, agents without purpose are part of current world situations [83].

The ability of an agent to act rationally depends on four factors [52]:

- A performance criterion that defines the agent's degree of success,
- Everything the agent can and might perceive,
- Everything the agent knows about its environment,
- Actions the agent can fulfill.

For measuring the achievement status of an agent, performance criterion is used. Performance criterion is established by external observers through determination of success as a standard. While it is possible for an agent to evaluate itself, the agent cannot make a subjective or a wrong evaluation [52]. The agent's self-evaluation may contradict rationality. Because the agent can consider the completion of its mission as its achievement. In this case, the criterion of achievement may be inconsistent with real world values, because real world values have an expectation-oriented structure. Certain criteria should be set in order to measure the performance of the agent. If the agent satisfies these criteria, it is considered successful, otherwise it is considered unsuccessful. However, an agent tries to meet all expectations within the framework of its perception capabilities, thus the agent may have difficulty responding to all elements outside of its perception field. If the agent does not have or cannot have enough information about the environment, this will affect the performance of the agent.

There are four different types of agent programs: simple reflex agents, agents that keep track of the world (model-based reflex), goal-based agents, utility-based agents, and learning [52-84].

- Simple reflex agents:

This represents agents who take a correct action based only on the current perception, in which all knowledge and perceptions obtained from the past are ignored. Simple reflex agents are used effectively only in environments where the agent can gather all information about the current environment, and this model is based on condition-action rules.

- Agents that keep track of the world:

The agent has environmental knowledge of the past and makes new decisions based on this information. In cases the agent cannot fully adapt to the current environment, it is required to produce action based on the knowledge of past observations and the information of current situation. Therefore, the agent has to maintain its inner state in order to choose an action. Such situations require the agent's program to encode two different situations: how the environment evolves independently of the agent's actions, and how the agent's actions affect its environment.

<u>Goal-based agents:</u> Goal-based agents take action to achieve a specific goal, depending on the current situation. A mere environmental knowledge may not be sufficient for the agent in all cases. In such cases, it tries to reach the main goal by dividing the desired goal into sub-goals. Sub-fields of AI such as search and planning in the research field are used for the actions that achieve the goals.

- Utility-based agents:

The methods followed by goal-based agents in achieving their goals may not be at the optimal level in terms of cost-benefit. In such cases, utility-based agents represent the utility functionality extension of a goal-based agent. The utility function plays an active role in the probability of success when there are conflicting goals that are partially achievable and a number of goals that are certainly non-achievable.

One of the most important data sources that a rational agent will use to achieve its goals is its environment. An agent collects environmental information through its sensors and performs its actions within the framework of this information. The data it can collect from its environment is directly proportional to the state of the environment and the perception of agent's sensors.

The different types of environments that affect the design of agents are as follows [52]:

- Accessible/Unaccessible:

If the sensors of an agent can have full information of an environment's state, that environment is accessible for the agent. If all aspects of the environment, including preferences of action, can be detected by sensors, that environment is accessible. An accessible environment is quite suitable for the agent because the agent does not need to maintain an internal state to follow the outside world.

- Deterministic/nondeterministic:

If the next state of an environment can be determined through the current state and actions can be preferred by the agent, that environment is deterministic. In principle, an agent does not need to worry about uncertain situations in an accessible and deterministic environment. If an environment is unaccessible, it is non-deterministic. If an environment is complex, it becomes difficult for the agent to follow directions of the unaccessible environment.

- Episodic/nonepisodic:

In a discontinuous environment, all the experiences of the agent are divided into episodes consisting of perceptions and actions. The quality of the action the agent will take depends only on the episode, since the next episodes do not depend on what actions were taken in the previous episode. Discontinuous environments are a simple structures for the agent because the agent does not need to think in detail about its future actions.

- Static/dynamic:

If the environment can change while an agent is operating, the environment can be said to be dynamic for that agent, otherwise it is static.

- Discrete/Continuous:

If there is a finite number of different, defined perceptions and actions, this environment can be said to be discrete. However, environments where such a situation is not in question are defined continuous.

Problem-solving agents are a type of goal-based agent that searches the necessary action sequences to reach the desired situations and perform their actions according to such search. The problems an agent will encounter in formulating its actions depend on whether the agent know the current situation of the agent and environment and the consequences of the actions it will take. An intelligent agent can be expected to behave in a way that can produce action to all other problems in the environment. However, in such a case, it may not be possible for an intelligent agent to perform optimally. A target formulation based on the current situation to be determined for an intelligent agent limits the agent’s interaction with the environment, leading the agent to perform its actions goal-based. The primary purpose of the problem solving agent is to decide, through the "search", what action it will take is most likely to approach its goal. Problem formulation is the process of deciding which actions and situations to consider and following the goal formulation. If the agent's next action has no knowledge-based basis, the agent can perform its next action by random decisions. However, in such a case, the agent uses its past knowledge to evaluate its next possible steps and prefer the action that best suits its goal. This process of problem evaluation and preference constitutes the "search" algorithm. The search algorithm evaluates a problem as input and aims to transform it into a solution in the form of a sequence of actions. Once the search algorithm finds and prefers the necessary solution, the decision mechanism is activated to implement it [52].

An agent has a state-based model of the world at the level of representation that is non-hierarchical or consists of a single hierarchy, in a probable uncertainty state [83]. The agent can find actions that will switch to the target state by evaluating the state space to move from its current state to a target state. Finding a path from a starting node to a target node can be abstracted mathematically.

3.2.2. Problem Solving

Problem solving strategies of the AI can be classified as search, use of information, and abstraction. Search is a strategy used in states where a direct and specific approach is not available. The use of information provides the solution of complex problems by using current state information. Abstraction, on the other hand, aims to produce solutions by distinguishing the important features from the unimportant ones.

Problem solving processes in AI are generally defined by search, using problem solving agents. The definition of search here represents all actions the agent will take to achieve the intended result. For taking the search action, the problem must be formulated. The formulation of the problem covers the process required for the implementation of these actions.

In order to find a way to move from its current state to a state that meets its goal, an agent can determine before taking any action how to achieve its goal in representing the world state space. The data provided for an agent to resolve the problem is limited in the search process that constitutes the foundation of AI. The agent can only identify the problem with the help of this data. All the ways to solve the problem should be found by the agent. Since humans do not have a general problem-solving ability in such situations and in solving difficult problems, they use a heuristic method. Unlike the search field, this extra information humans use in general is called heuristic information [83].

The primary task of an agent is to formulate the problem that consists of four parts: the initial state, a set of operators, a goal test function, and a path cost function [52]. The concepts representing the problem can be summarized as follows: Initial state, operator, goal test, path cost. Initial state represents the initial state the agent is in, that it has knowledge of. Operator refers to the set of actions that defines what consequences the action the agent will take in a particular state can lead to. The goal test checks if one of a number of possible goal states is simply reached, which can also, in some cases, be determined by an abstract feature rather than a clearly itemized set of states. Path cost refers to the sum of the individual costs of actions on paths used to achieve the goal and represents the function that assigns costs for a path used.

These concepts represent a simple single-state problem. Multiple-state problems, on the other hand, are similar to single-state problems.

> To deal with multiple-state problems, we need to make only minor modifications: a problem consists of an initial state set; a set of operators specifying for each action the set of states reached from any given state; and a goal test and path cost function as before. An operator is applied to a state set by unioning the results of applying the operator to each state in the set. A path now connects sets of states, and a solution is now a path that leads to a set of states all of which are goal states. The state space is replaced by the state set space [52].

The formulation of the problem often requires abstracting real-world details to describe a state space that can be actually explored.

3.2.2.1. State Space Representation

A problem in which intelligent action will take place can be represented by "state-space". A state space contains all the information necessary to predict the effects of an action and determine whether the state meets the goal. A search to be made in state space is a process used to find a goal state having a desired property.

The knowledge of an agent and the knowledge of its state play an important role in formulating problems. There are basically four different types of problems: single-state problems, multiple-state problems, contingency problems and exploration problems [52]:

- Single-State Problem:

Single-State Problem describes the situation in which an agent knows the reason for each action it will take, and in what state it might be after each sequence of actions in an accessible environment.

- Multiple-State Problem:

Multiple-State Problems represent the problems in which the agent has knowledge of all the effects and initial state of its actions, but has limited knowledge of the state of the environment, requiring the agent to reason about all possible states.

- Contingency Problem:

Contingency Problems represent the problems in which the agent is unable to fully detect its environment and have no knowledge of the effects of its actions. Problem solving in such problems requires perception during execution and the agent computes an entire action tree rather than a single sequence of actions. The reality can be defined as a contingency problem since many problems in the physical world are not precise and predictable.

- Exploration Problem:

Exploration Problem is the type of problem in which the agent has no knowledge of the effects of its actions, and can learn the effects of its actions and the state of the world by making self-trials. This type of problem requires a complete exploration for the agent. In the exploration problem, which represents a difficult process for the agent, the agent can create a map of the current world and perform its actions according to that map, if it can survive.

To solve a problem using AI, the state space must be determined. There are generally three basic components to determine the state space. These components are defining the state of the problem, the purpose and the operators. Defining the state of the problem is about naming the problem. Some states that are thought to cause the problem may be a solution to the problem. The purpose represents the goal to be achieved. Multiple explorations may be required to achieve the goal. Operators, on the other hand, are a function used to achieve other states from a state to the desired state.

3.2.2.2. The Problem-Reduction Problem Representation

The problem solving can be regarded as achieving sub-goals on the path from the first goal to the final goal. Using the problem reduction method, the problem is divided it into a set of sub-problems or sub-goals, thus restructured. This problem representation is commonly used in game-playing problems. When solving a game-level problem or a problem that can be represented as a game, it is often complicated to handle the entire problem space as a single problem, hence, instead of looking for the final solution at some point, it is necessary to continue the exploration by making a benefit-loss evaluation of the subgoals achieved.

These multiple methods of reducing a problem to smaller problems can be represented as a generalized tree called an AND/OR graph.

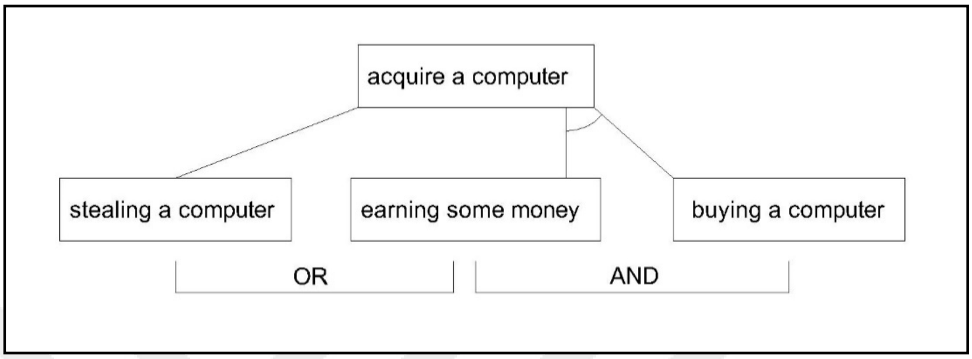


Figure 3.3. Example of an AND/OR graph.

Figure 3.3. shows three different sub-solutions of acquiring a computer. The first solution is stealing a computer, the second solution is earning some money, and the third solution is buying a computer. However, the second solution is not sufficient to acquire a computer alone, but also requires the third solution, and vice versa. Therefore, the third solution and the second solution are connected by the AND connector, and this connection is denoted by an arc. The first solution method is sufficient to acquire a computer alone, so it is separated with the OR connector from the solutions connected by AND connector. Thus, the number of solutions is reduced to two and these sub-solutions are sought first. The AND/OR graph represents a benefit-loss evaluation of a problem with different solutions or sub-solutions and the solution of the sub-problem with the least cost followed.

3.2.2.3. Search Strategies

The state space of a problem can be expressed in two ways: graph search and tree search. Graphs are the shapes used to represent structures encountered in real life. A graph used to search for an actual solution consists of nodes and arcs. Various mathematical and visual methods are used to express graphs. Arcs are aligned as node pairs that may have associated costs. Searching in graphs provides a suitable level of abstraction to study simple problem solving regardless of a particular space. In graph search, the goal is to find a path along these arcs from a starting node to a target node. In general, many problem solving tasks can be solved with the graph search method. Abstraction is necessary for a problem to be solved by graph search, because there can be multiple ways of handling a problem.

In the graph search process, a list called closed list (explored set) is used in order not to revisit and expand previously resolved nodes. In this way, a single expansion operation is performed for each node expanded. The advantage of graph search is that, if a node search is terminated, no searches occur between nodes again. However, the disadvantage of graph search is that it has a cyclic structure and requires a lot of memory area.

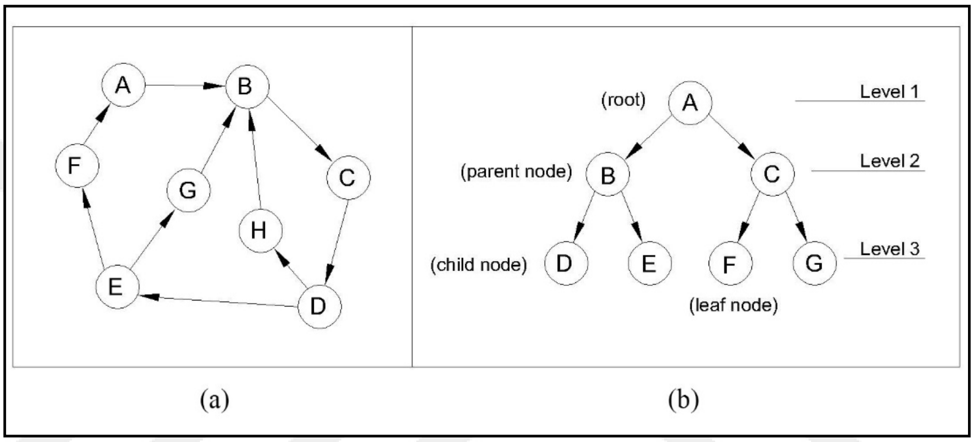


Figure 3.4. (a) Graph, (b) Tree.

Tree is a graph that contains no loops and is not in a loop. The data is kept similar to a tree structure. Trees consist of sections that represent almost a real tree such as node, leaf, depth, level and root. The data that the tree has is called the node. Each new data produced with the initial data called root and the new child data generated from that data constitute a new level. Each new node line generated with root is named depth. The nodes that do not constitute any child/sub-generation are named leaf. Each node in a tree is defined as a problem state. All paths from root to other nodes define the problem state space. In the case of a tree search, no closed list is kept and the same node can be visited an infinite number of times, ie the generated tree can contain the same node multiple times.

The problem solving method through search requires choosing the right strategy for problem solving. These strategies are evaluated under four main criteria as completeness, time complexity, space complexity and optimality [52].

Completeness determines the criteria whether the strategy determined for the solution of the problem can achieve the final solution. Time complexity is the criterion that determines how long the determined strategy can achieve a solution. Space complexity is the criterion that determines how much memory the specified strategy needs to perform the search. Optimality is the criterion that enables the determination of the best-quality solution among different solution alternatives. Search strategies are divided into two search strategies: uninformed (blind) and informed (heuristic).

Uninformed search is a search strategy with no information about the number of steps to be followed from the current state to the goal state or the cost of the path. An uninformed search is sometimes called a blind search. These algorithms ignore the paths followed until a goal is found and success is achieved. Uninformed strategies use only the information available in the problem definition.

Uninformed search strategies are actively used because of the limitations of the information available, and many real world problems do not contain precise information [52].

Informed search is a search strategy that has limited information about the number of steps to be followed in problem solving from the current state to the target state or the cost of the path. Informed search strategies are also called heuristic search strategies and are the algorithms in which an heuristic approach is applied to problem solving. The accuracy of the method used in Informed Search does not need to be proven, all that is required is to simplify a complex problem or that the algorithm can find a satisfactory result. Heuristic algorithms either produce a quick solution to the problem, but it cannot be guaranteed that it will always solve the problem, or it solves the problem in a reasonable time, but not always at the same speed. Heuristic algorithms are a type of search that humans frequently use, especially in real life problems.

3.2.2.4. Knowledge Representation

To solve complex problems using AI, large amounts of information and some mechanisms are needed to manipulate that information to use it to produce solutions. An intelligent agent needs a knowledge base (KB) of the real world to make effective decisions and reasoning. Knowledge-based agents are capable of maintaining an intrinsic state of information, reasoning on that information, updating information after observations, and taking action. These agents can represent the world with some formal representation and act wisely.

The concept of logic refers to the ability of an agent to establish a link between some information involving its environment and its own actions. Developing computer programs with AI programming technology requires organizable knowledge. Knowledge may be specific to a problem, generalized in a field, deep, superficial, precise, doubtful, ambiguous, incomplete, etc. The purpose of knowledge representation is to quickly access the necessary knowledge during the realization of intelligence-related functions such as decision-making, planning, presentation of objects and states related to AI programs [85].

Knowledge-based agents consist of two main parts: knowledge base and inference engine. The knowledge base is the central component of the agent, which stores the facts regarding the world as "sentences" encoded in a knowledge representation language. A knowledge base is necessary for an agent to learn from experience, act according to knowledge, and update information. Inference engine is the unit that enables the agent to make recommendations about the world. The knowledge-based agent applies a set of logical rules to add new sentences to sentences that it has learned about the world and stored in its knowledge base. The inference system produces new facts so that an agent can update its knowledge base. An inference system basically works based on two rules: forward chaining and backward chaining [86].

Forward chaining is a form of reasoning that begins with atomic sentences in the knowledge base and applies inference rules in forward direction to extract more data until a goal is achieved.

Backward chaining starts with a list of goals (or a hypothesis) and works backwards from the leading one to see if any data supports any of these results, and applies backward inference rules. In forward chaining, the propositions, of which accuracy is known, have priority, while in back chaining, the propositions, of which accuracy is sought, are prioritized.

A knowledge-based agent needs to know many things: the current state of the world; how to infer the unseen properties of the world from perceptions; how the world developed over time; its goal to be achieved; and cause and effect relationships of its actions in various states. There are some procedures for adding new sentences to the knowledge base, questioning the known, and demonstrating intelligent behavior. These are TELL and ASK.

TELL function refers to the process of adding knowledge obtained from the environment to the knowledge base. ASK is the function that asks the knowledge base what action it should take. The important point here is that these functions are continuously successive. When asked about a knowledge base question (ASK), the answer must come from a discourse (TELL) that was previously stored in the knowledge base. The ASK function uses the inference engine that determines what is tracked from what is told (TELLed) to the knowledge base. Knowledge-based agents that take perception as input TELL the knowledge base what knowledge is perceived and then ASKs what action to take. With logical reasoning, the agent decides which action is the most appropriate, within its goals and knowledge, and fulfills the decided action [52].

At any point, we can define a knowledge-based tool at three levels:

- The knowledge level:

This is the level at which the agent is defined within its knowledge. At this level, it is stated what the agent knows and what the agent's goals are. At the knowledge level, it is assumed that the agent knows the path to achieve the goal.

- The logical level:

This level shows how knowledge is stored in the representation of knowledge, that is, the level where knowledge is encoded into sentences. At the logical level, an agent is expected to be able to achieve its goal.

- The implementation level:

This is the physical representation of logic and knowledge. This is the level where the physical representations of sentences are available at the logical level. At the implementation level, the agent takes actions according to logical and knowledge level. At this level, the agent applies knowledge and logic to achieve the goal.

For a system to be intelligent, it must have knowledge of its world as well as the ability to make inference from that knowledge and take action. AI researchers suggest two approaches in which knowledge is represented: Procedural approach and Declarative approach. In AI researches, the knowledge is represented by the procedural approach directly in the programs that use such knowledge. In the procedural approach, the desired behavior is encoded directly as a program code. The declarative approach, on the other hand, is represented in symbolic structures different from many different programs that can use the knowledge in these structures. In this approach, a knowledge-based agent is created by telling (TELLed) the desired "sentences" to an empty knowledge base [87]. Thus, in the real world, a successful agent can be created by combining both declarative and procedural approaches.

- Logical representation of knowledge:

The concept of logic is a means for an agent to establish a link between its environment, its actions, and its knowledge. Therefore, rational agents need a logical order to represent knowledge.

Agents keep the data in the knowledge base as sentences in the representation language used and define them by associating these sentences according to their semantics, syntax, and pragmatic relationships. Syntax refers to which group of symbols is arranged in what way; semantics shows what well-formed expressions mean, and pragmatic refers to how to use meaningful expressions in language, how to use meaningful sentences of a representation language for knowledge representation as part of a knowledge base from which inferences can be made. In cases where the actions of the associated sentences coincide with real world values, then the correct decisions can be mentioned [88]. In AI systems, information can be represented by classical logic, as well as by logic languages such as First-Order logic that handle multiple environments and multiple objects with different methods.

- Inference:

A knowledge base (KB) is a technology used by a computer system to store complexly structured and unstructured information. The process of creating the knowledge base is named knowledge engineering. A knowledge-based system consists of a knowledge base that represents facts about the world and an inference engine that can reason about these facts and use rules and other forms of logic to reveal new facts or highlight inconsistencies [86].

Making the right decisions by the agent is possible by making an inference from the knowledge about previous states and the current state. Inference can be briefly explained as producing conclusions from evidences and facts revealed by suggestions. However, different logical approaches that the agent has may produce different conclusions. It is important for accuracy that an inference is applicable to all sub-worlds. The inference process is completed when the agent processes the data available to it, creates a conclusion set, and infers all the real conclusions from this set.

While the terms "reasoning" and "inference" include all processes in which conclusions are reached, logical inference is a process applied to establish a healthy relationship between sentences. It is necessarily true sentence for the world/worlds that are required for a healthy logical inference. In order for a sentence to be considered correct, it must be applicable and satisfactory. A sentence is true if it can be considered true in all possible situations in all possible worlds. A sentence can be qualified as satisfactory only if there are some interpretations in some worlds where it is true [52].

- Resolution:

Resolution is a theorem proofing technique that produces proofs with the help of contradictions. This technique is mainly used to prove the satisfactory nature of a sentence. Since the knowledge base has a consistent structure within itself, contradictory sentence structures that create inconsistencies must be proven within logic and rejected. This makes achievement of consistent goals possible.

- Reasoning:

Reasoning in AI refers to the process of a rational agent to reach logical truths by making predictions from existing world knowledge, facts and beliefs, which expresses a process just like the human brain works and thinks. Reasoning is necessary to create a rational system.

Intelligent agents are modular systems with a control mechanism that can represent knowledge and use it in a reasoning process that can independently or jointly use each represented knowledge. While all these properties ensure that the agent has a rational structure, it facilitates the system's self-learning. Automated reasoning systems are designed to address different problems and can be divided into the following categories [52-89]:

- Deductive reasoning
- Inductive reasoning
- Abductive reasoning
- Common sense reasoning

- Monotonic and non-monotonic reasoning

From the very beginning, AI researchers realized that the rules of inference of formal logic provided one of the most powerful tools in the information processing toolkit. Inference is a method of reasoning from given pure knowledge to the new knowledge. The AI system should include the rules of inference in order to eliminate or apply the present problem. Since logic rules limit the problems to a specific method of reduction and association, the search function can be well maintained by a system. Another very important role of logic is to represent the knowledge itself and to provide formalism to derive inferences from this knowledge. The goal of AI research is to incorporate inference rules into computer systems, thereby automating the reasoning process [89].

- Deductive reasoning:

Inductive reasoning is the most basic example of reasoning based on the logic of cause and effect, and it simply reaches a general conclusion or statement by using the reduction method from a finite number of facts by following a vertical path. As the conclusion or statement reached is expressed with the knowledge in the knowledge base, its accuracy is not certain. The data in the knowledge base are only the basis for inference and cannot guarantee achieving an exactly correct conclusion. However, if all the propositions are true and the deductive rules are followed with a certain logic, the conclusion may be true, while this accuracy may not be valid for other worlds.

- Inductive reasoning:

Inductive reasoning is defined as a method in which simple generalizations are made about all other phenomena after examining only a few specific phenomena, and experiences and observations are synthesized to reveal a general truth. Inductive reasoning is a type of logic of suggestion, also known as cause and effect reasoning or bottom-up reasoning [89]. It represents the process of producing a general conclusion starting from a particular phenomenon. The factuality of the facts does not always guarantee the accuracy of the conclusions, but contributes to a certain level of generalization.

- Abductive reasoning:

Abductive reasoning, which is an extension of deductive reasoning, is the approach used to reach the most probable conclusion with the help of single or multiple observations. While abductive reasoning can reach a reasonable conclusion as opposed to being an extension of deductive reasoning, it cannot guarantee an exact truth like other approaches.

In abductive reasoning, an agent assumes what might be true about an observed event. An agent determines what may be true for its observations to be correct. Observations are trivially implied by contradictions. Abductive reasoning is a form of reasoning in which assumptions are made to explain observations [83].

Deductive, inductive, and abductive reasoning are at the basis of AI studies beside being the forms of classical logic. However, in order for AI to have a rational structure, it needs more than a relatively narrow frame of classical logic. A rational behavior can be achieved by the agreement of truth and accuracy with general world realities. For this reason, the hypothetical reasoning techniques of the propositional logic cannot present a complete rationality. The reasoning and conclusions provided by classical logic can be mentioned as recommendations and general phenomenon ideas. AI systems need more rational methods of reasoning. The reasoning methods used in AI can be listed as Common Sense Reasoning, Monotonic Reasoning and Non-monotonic Reasoning.

- Common Sense Reasoning:

Common Sense can be shown as a less developed area than other studies during the development of AI studies. All phenomena that require general world view and knowledge such as size-smallness, length-shortness, freshness-staleness, etc. require a technique different from the reasoning abilities of the propositional logic. Common reasoning is one of the branches of AI that deals with simulating the ability of humans to make assumptions about the type and essence of ordinary situations they face every day [90].

One of the most important reasons underlying the slow development of Common Sense Reasoning can be said that the skills that Common sense reasoning researches are acquired through experience. These experiences are in a structure that a system with general world knowledge can evaluate and infer conclusions from, and they are in a heuristic structure just like in daily human life. Common sense reasoning is used in many fields such as natural language processing, computer vision, and robotic manipulation.

- Monotonic and Non-Monotonic Reasoning:

In monotonic reasoning, when a conclusion is inferred, existing knowledge remains current even if other knowledge is added to existing knowledge in the knowledge base. The valid conclusion to solve monotonic problems can only be derived from existing facts. Monotonic reasoning is not useful for real-time systems as world realities change real-timely. Ordinary deductive reasoning is monotonic, that is, new facts can only generate additional beliefs [88]. If some conclusions that the system can reach with new knowledge added to the knowledge base may be invalid, this is called non-monotonic reasoning. Non-monotonic reasoning is a rule that can be used unless it is overridden by an exception [83].

After addressing to definitions, historical process, and problem solving abilities of AI, the approaches, in which AI is involved, will be examined in the following chapters of this thesis.

3.3. MAIN APPROACHES AND APPLICATION AREAS OF AI

It can be said that the field has gained more functional and powerful capabilities thanks to the various AI approaches that have emerged with the development of AI technologies. With these approaches, it can be said that AI is divided into sub-fields and AI technologies have developed in specific subjects with studies on these fields. While these approaches gained importance in some periods of the historical development stages of AI technologies, it can be mentioned that they lost their significance in some periods with the developing needs and technologies.

Apart from AI approaches, there are also some application fields where AI technologies are directly or indirectly related to a method. Such application fields cover the fields of study that are used in fields other than AI technologies, but where it can interact due to the multidisciplinary and interdisciplinary nature of computer sciences. In this section of this thesis, the application fields and approaches of AI technologies are examined.

3.3.1. Machine Learning

Machine learning (ML) is a field that includes the creation of computer programs that develop automatically with experience. ML benefits from the concepts and results from many fields such as statistics, AI, philosophy, knowledge theory, biology, cognitive sciences, computational complexity, and control theory [91].

ML is an AI system that can follow constant and variable resources and thus make predictions and learn with the help of constantly changing data instead of programming. ML enables data to be analyzed, defined and predicted in the light of these data with the help of algorithms learned recursively. Each data collected by algorithms is an exercise for ML. ML is trained with these data, and when a model containing inputs is provided to the system after the training, the system produces an output in line with its training. ML is also a suitable system for creating analytical models.

Some ML models have an online structure and maintain their own training with the new data gained from the online environment, thus strengthening the relationship between the obtained data items. Since the growing data size will increase the level of complexity, human observation and human learning model may cause many errors, while ML learning algorithms minimize the error risk. ML can automatically adjust to rapid changes in data variables and generate an output by considering the probability of variables. Online ML algorithms continuously improve models by continuously processing new data in real time and training the system to adapt to changing patterns and associations in the data [92].

Deep Learning (DL), considered as a sub-branch of ML, is an ML method that is based on artificial neural networks and is used to train systems in much shorter times and using much less data center infrastructure.While DL has Supervised Learning and Unsupervised Learning concepts like ML, as it is a machine learning method, whereas DL method also includes Semi-supervised Learning concept. Semi-supervised Learning is an ML approach that combines small amounts of defined data with large amounts of undefined data during the training of the program. Deep learning uses many layers of nonlinear processing units for feature extraction and conversion. Each successive layer uses the output from the previous layer as input. Today, the most intensive studies in the field of ML providing the best outcomes are carried out on artificial neural networks.

Artificial neural networks are a class of ML algorithms that are inspired by the working structure of the human brain, learn from concrete data and specialize in pattern recognition. Deep learning is a member of the family of neural networks algorithms and the two terms can be used interchangeably [93]. Deep Learning has become possible with neural networks with wide layer structure, thanks to developing hardware technologies and back propagation algorithms.

Artificial neural networks are one of the artificial intelligence methods that mimic biological nerve cells (neurons) by creating an artificial model of the human brain. Artificial neural networks began in 1943 with the first AI study, which is the basic physiology of neurons in the brain, one of the topics McCuloth and Pitts focused on and studied. In 1957, Frank Rosenblatt, a psychologist at the Cornell Aviation Laboratory in New York, started to conduct studies on neural networks in the PARA (Perceiving And Recognizing Automaton) project in the light of the studies of McCulloch and Pitts. Frank Rosenblatt aimed to create a representation of the human learning, cognition and memory model with these networks, which he called "perceptron". Rosenblatt's perceptrons consisted of neural elements formed by McCulloch and Pitts. [87].

When neural networks are viewed from a biological perspective, artificial neural network studies can be defined as a mathematical model of brain operations. In addition to useful computational features, neural networks can provide a chance to understand many psychological phenomena that result from the specific structure and functioning of the brain. The most important resource that will help in identifying artificial neural networks is the biological structure and learning mechanism of the brain subject to simulation.

Although the data processing process of computers is faster, error-free and stronger than humans, artificial neural networks of humans that are fed by the senses are more capable in learning than computer systems. Computers have gained their ability to identify and predict data with their own reasoning method gained by algorithms based on artificial neural networks. This made computers have a learning method that is very close to human thinking.

In general, human brain has neurons more than the “bytes” that computers may have, and these neurons require the brain to handle different tasks differently. While the human brain needs to perform more operations in a shorter time than computers in daily life, the slow development of the human brain and the high time it spends to process, synthesize and output data turns the table on. Nevertheless, the human brain is considered superior to computer systems because of the ability of the human brain to activate all neurons and synapses at the same time, and the fact that artificial neurons in computer systems must go through many cycles to activate them. The basic idea behind creating a model of the human brain with artificial neural networks is to create a device that combines the ability of the human brain to activate neurons and synapses with the switching speed of the computer, as opposed to creating a benchmarking environment between human and computer [52].

Artificial nerve processing element consists of nerve cells (neurons). Neurons have separate memory structures. Neurons are connected to each other through connections that have values called weights. In addition, ANNs are capable of making complex data processing and computations. Hence, the ANN can be said to imitate biological neural networks.

A neural network consists of several nodes connected by links. Each link has a numerical weight associated with it. Weights store data in neural networks for a long time and learning usually occurs by updating numerical values of such data. Some of the units are dependent on external environment and can be defined as input or output units. The weights are changed to try to make the input/output behavior of the network more compatible with that of the medium that supplies the inputs [52].

Different types of network structures can be used due to their different computational features. The most basic network structure differences are seen in forward feed and recurrent networks. In a feed forward network, the links are directly unidirectional and there is no loop connection. While there is no link between networks in the same layer in a layered feed forward network, these links cannot be backward either. The absence of loop capability in a feed forward network ensures that the computation can proceed equally from input units to output units. In recurrent networks, connections can form arbitrary topologies. Recurrent networks, on the other hand, can implement more complex agent designs and model systems depending on the state. Recurrent networks require highly sophisticated mathematical methods [52].The main application areas of artificial neural networks can be considered as classification, prediction, and modeling [94].

The neural network of McCulluch and Pitts, who introduced the first artificial neural network, had no set of rules required for learning to occur. Published his book “The Organization of Behavior” in 1949, Donald Hebb showed a simple updating rule to change the connecting forces between neurons, enabling learning become possible. Hebb argued that the force of interaction between the two nerves is dependent on that the input transferred from one nerve to the other makes both nerves highly active. Hebb hypothesized that long-term memory in humans and animals is accompanied by persistent changes in synapses. Hebb thought that the phenomenon of "shooting together" tends to persist in the brain, and that this is the way the brain represents the perceptual event that led to the formation of its cell assembly. Hebb suggests that "thinking" is the sequential activation of cell assembly sequences [87-95].

In general, ML has three main concepts:

- Supervised Learning:

Every case in which both inputs and outputs of a component can be perceived in ML is called supervised learning [52]. In supervised learning, it is known what the data sets are and it can be predicted what the data to be created with these data sets will be. A supervised learning algorithm analyzes training data and produces an inference function that can be used to match new samples from this training data. In this way, the learning algorithm creates a function from the inputs to the corresponding targets.

- Unsupervised Learning:

Unsupervised learning is learning where there is no clue about the correct outputs the system can produce. In unsupervised learning, the system can learn to predict its future perceptions using supervised learning methods. However, learning new data is impossible unless it has an auxiliary program [52].

Unsupervised ML is better suited to AI studies as the program does not require automatic data input and can learn complex processes with its own methods.

- Reinforcement Learning:

Reinforcement Learning consists of a system that can perceive its environment and make decisions on its own. In this way, Reinforcement Learning is a system that perceives the data with its own mechanism and learns how to make the most appropriate decisions to produce outputs. The system starts without any model of the environment and utility functions. In reinforcement learning there is a trainer but does not give too much detail to the system as in supervised learning. Instead, when the learning system makes a decision, it rewards the system if that decision is right and punishes if wrong. The aim is to check whether the possible states that the student is trying to control are targeted and remember all the states that have been tried.

3.3.2. Expert Systems

In the early years of AI studies, a general-purpose search mechanism that tried to bring together basic reasoning methods was emphasized, and these general-purpose search mechanisms, which were created with the weak information method it used, and displayed low performance for complex problems, were called "weak methods". For this reason, researchers recommended to approach narrower spaces with a broader reasoning method [52]. Since the second half of the 1960s, researchers have emphasized that knowledge is as important as reasoning in an intelligent behavior method, and this situation may be a scientific turning point [68].

Developed by Feigenbaum et al. at Stanford University in 1967, the DENDRAL program is the first AI assistant designed and put into use. DENDRAL is the first program to bring the knowledge of an expert to the computer environment by means of AI. The main purpose underlying the creation of the DENDRAL program was to analyze the data to be obtained from a mass spectrograph with a chemist's logic, by developing a chemistry program that would accept physical data and generate sufficient hypotheses to explain the data. DENDRAL is the pioneer of next generation AI programs called knowledge-based systems. Feigenbaum et al. made observations by meeting with chemists to analyze specific situations such as how chemists think, how they answer questions, and how they approach problems. As a result of these observations, they developed the method of representing specific behaviors in a large database with DENDRAL. The significance of DENDRAL is inarguably that it is the first successful knowledge-based system [52-68].

In AI, an expert system is a computer system imitating the decision-making ability of a human expert [96]. Expert systems are created on the basis of the experiences of experts in their field, their approach to problem solving models and their inferences. The difference of expert systems from any computer program is, unlike conventional computer programs, their "knowledge-based" structure and their ability to control their knowledge-based structure with their "inference engine". Thanks to the inference mechanism, expert systems can retrospectively question the reasons of the result.

In an expert system, the knowledge base represents facts and rules. The inference engine applies the rules to known phenomena to deduce new facts. Inference engines may also include explanation and debugging capabilities.

The expertise possessed in a subject represents a specific domain and can be defined as the ability to solve problems in the domain. Regardless of the domain, knowledge is of two types, public and private. Public knowledge includes generally accepted definitions and theories predicted by sources on the subject. However, public knowledge consists of the heuristic scanning methods that an expert should have and a set of subjective rules created by the expert within the framework of gained experience, and these rules enable the expert to deal effectively with incorrect or incomplete data.

Researchers suggest that the emphasis should be placed on the knowledge itself rather than formal reasoning methods because there is no specific path that can be followed to solve problems that can be interpreted as linguistically complex and unusual, the adaptability of the knowledge of human experts to concrete facts with a pragmatic point of view, and since the knowledge is considered a lofty resource in all time periods, and every new knowledge produced is perceived as a new wealth. Knowledge is a key component of expert performance as in all fields. Expert systems include symbolic inference and heuristic search features of AI, unlike conventional data processing systems. The ability to perform difficult and complex operations at the level of expert performance, to develop solutions and strategies to problems in certain areas that are considered weak AI, to use self-knowledge to analyze their own inference processes, and to be used in solving problems in areas such as interpretation, prediction, diagnosis, debugging, design, etc. are the headings that distinguish expert systems from general AI systems [86].

Knowledge consists of symbolic definitions that characterize descriptive and empirical relationships in a field, and procedures for manipulating these definitions. The purpose of knowledge-based systems is to reveal the critical knowledge required for the system to operate explicitly rather than implicitly. Unlike conventional computer programs, the purpose of expert systems is to state the rules in a heuristic and easy-to-understand, revised, or even edited format by an expert in that field rather than a computer programmer. The maintenance costs of expert systems are low, as they eliminate the need for coding, minimize the errors that may be caused by software problems, have a rapid prototyping process, and the system can develop itself [86].

Expert systems use the historical value of knowledge to make logical inferences. The existence of knowledge does not purely serve a purpose without seeking benefit and is symbolic. However, while making logical inferences, an expert system performs the process of using and analyzing the knowledge by a human expert, and transforms the knowledge into benefit.

3.3.3. Fuzzy Logic

Propositional logic, or classical logic, requires a clear distinction when defining the elements of a set between members of that set and those that are not. In daily life, definite classifications are not used most of the time. Fuzzy logic pioneers the creation of an intelligent system that is convenient and useful to human thinking by modeling everyday linguistic elements, decision-making mechanisms, and heuristic states [97].

When some mathematically modeled systems that contain a general world view are created according to the precision values of mathematical systems expressed by "1 and 0", their use in supervisory design might cause high costs or inaccurate results. Because in the real world, precise results are out of question. Relative concepts of the real world require a heuristic and experiential, and cultural perspective. In addition, problems may be encountered in the controller performance of control algorithms that can change constantly, remain uncertain, and are not well-defined.

Systems based on fuzzy logic may have heuristic degrees of belief and may also allow truth degrees: a phenomenon does not have to be true or false in the real world, but can be true to a certain extent [52]. In cases with a certain degree of truth, if the mathematical expression is insufficient, an expert opinion may be required and the expert can respond to this situation with imprecise qualitative linguistic elements. However, the knowledge of experts, the results that experts put forward heuristically, and the problems that may arise due to the expression of linguistic qualifiers can be achieved with a mathematical system. For these reasons, fuzzy logic can be regarded as a mathematical representation of the real world [94].

Classical logic has a two-level representation mathematically represented by "0" or "1". In classical logic, a phenomenon can be expressed with certain reasoning such as yes/no, and a phenomenon in classical logic can only be a member of a single set. Fuzzy logic is expressed in a multi-level representation between "0" and "1". According to fuzzy logic, a phenomenon can be described with more linguistic and real-world expressions such as less, more, sufficient, insufficient, and a phenomenon in fuzzy logic can be a member of several sets. Fuzzy set theory allows an element to have partial membership in a set.

The foundations of fuzzy logic were laid by Jan Lukasiewicz with studies that can be summarized as non-classical and single or multivalued logic in the 1920s [98]. In the 1930s, Lukasiewicz introduced a new type of logic that could be described as fuzzy or multivalued logic developed as a result of his studies. This logic has been introduced to the literature as the Lukasiewicz Notation. Lukasiewicz showed that, contrary to classical logic, the truth value of a phenomenon can be described by all real numbers between “1 and 0”. The philosopher Max Black published his article "Vagueness. An Exercise in Logical Analysis", published in 1937, in which he argued that there are degrees of continuity and that uncertainty is a probabilistic problem. Black described the first simple fuzzy set in the appendix to his article and summarized the basic ideas of fuzzy set operations. In 1965, the term "Fuzzy Logic" was introduced by Professor Lotfi Zadeh, Head of the Department of Electrical Engineering at the University of California, with the suggestion of fuzzy set theory. Aiming to represent and manipulate fuzzy terms, this theory extended probability theory to the mathematical logic system and studied on the natural language processing mechanism [65-97].

Fuzzy set theory provides a framework for the cases where there are uncertain and imprecise conceptual states, fuzzy relationships, criteria and phenomena, and can be considered as a very suitable modeling language for modeling non-conceptual fuzzy states. In classical normative decision theory, the components of the basic decision-making model can be explained as definite sets or functions. The action set is defined as precisely as the set of possible states, and the utility function is assumed to be precise. Uncertainty is included in the theory only in the case of evaluating decisions under uncertainty or in the risk assessment process and is modeled within the framework of probabilities. In descriptive decision theory, on the other hand, uncertainty is usually modeled with linguistic elements and generally does not allow the use of powerful mathematical methods for analysis and computation [99].

> The notion of a fuzzy set provides a convenient point of departure for the construction of a conceptual framework which parallels in many respects the framework used in the case of ordinary sets, but is more general than the latter and, potentially, may prove to have a much wider scope of applicability, particularly in the fields of pattern classification and information processing. Essentially, such a framework provides a natural way of dealing with problems in which the source of imprecision is the absence of sharply defined criteria of class membership rather than the presence of random variables [65].

Zadeh (1990) explains the properties of fuzzy logic as follows [100]:

- In fuzzy logic, real values are allowed to be fuzzy sets labeled as true, fairly true, very true, more or less true, mostly true, etc. For instance, the real value of a suggestion, unlike classical logic, can be expressed as "very true" and represents a fuzzy subset of the very true unit range.
- Classical logic systems, including multivalued logic, allow the use of only two quantizers. In fuzzy logic, quantifiers such as most, many, few, often, etc. are interpreted as fuzzy numbers that serve to describe the relative qualities of fuzzy sets. This interpretation is important in terms of representing the meaning of suggestions in a natural language.
- Fuzzy quantizers have fuzzy probabilities underlying much of the reasoning used in decision making in everyday life and provide mechanisms for qualitative decision analysis through the link between fuzzy quantifiers and fuzzy probabilities.
- Conventional approaches to those meaning representation in natural languages are mostly based on suggestion logic and variables. Such approaches do not address answers such as, for instance, more, more or less, quite, a little, a lot, etc. Fuzzy logic is interpreted as an operator for linguistic items and provides a method for dealing with these mechanisms.

Fuzzy logic depends on a set of rules just like any other type of logic. While expert systems created with a rule-based structure represent a kind of expert system written with IF-THEN connectors, whereas, in an expert system created with fuzzy logic, all and/or some of the variable values have imprecise reasoning.

Fuzzy logic theory can perform effectively as an expert opinion in cases where an instance under investigation is described as very complex, uncertain and where there is a lack of knowledge. Fuzzy expert systems, developed based on fuzzy logic, make reasoning by using numerical operations instead of symbolic reasoning, unlike conventional expert systems. The use of fuzzy logic in expert systems increases the efficiency and decreases the response time. Fuzzy checker rules are often formulated with linguistic terms. Hence, the use of linguistic variables and fuzzy sets implies matching input variables to appropriate linguistic values. Bringing linguistic skills to an expert system provides the system with the ability to analyze and make decisions just like a human expert.

3.3.4. Genetic Algorithm

Evolutionary computational methods create a simulation of evolution, modeling the evolutionary process, and this simulation usually involves a set of optimization algorithms based on a set of simple rules. They are able to learn to predict changes in their environment by conducting a search within the framework of the rules set of simulation, by transferring the genetic characteristics of each solution set to a subgeneration until an optimum solution is reached or by terminating the generations. Evolutionary computation simulates evolution using selection, mutation, and reproduction processes. Although the process that natural evolution goes through is long, the solution-based evolutionary process in the computer environment does not take as long as the natural evolution process does [97].

Nils J. Nilsson (2009) argues that the evolution of living creatures in natural life gives humans two clues to create intelligent machines: the idea that every new object can be intelligenter, because evolution can be simulated in computers to create imagined intelligent machines, and evolution creates intelligenter organisms with each new generation [87]. While such two clues can be shown as the main sources of interest in evolutionary computation methods, the fact that the nature, alleged to be the best designer and computer in the historical process, is imitable has enabled the studies in this field to be followed with various methods.

Holland argues that the increasing complexity creates long and perhaps impossible problems to solve, and the adaptation will make it possible to overcome the complexity problems arisen. Adaptation involves the gradual change of a number of structures. A system subjected to adaptation is characterized by a mixture of operators substantially influencing structures at each stage [101].

The principle of survival of individuals with optimum conditions caused by evolutionary processes may seem like cruel at first glance. However, evolution takes into account the "fitness" of living creatures for the continuity of life. In cases where these fitness conditions cannot adapt to the developing/changing fitness conditions of nature, it is ensured that new and adaptable individuals emerge with the strong genetic characteristics of their parent generations. Hence, the main goal of evolution is to create a population of individuals that fit its increasing fitness value.

The process that started with Charles Darwin, who introduced the theory of evolution in London on July 1, 1858, has established Neo-Darwinian thinking based on reproduction, mutation, competition and selection processes with Darwin's classical theory of evolution, Weismann's theory of natural selection, and Mendel's concept of genetics. Reproduction is regarded as the basis for the survival, mutation as the core element that enables adaptation to changing dynamics, and competition and selection as the factors determining the continuity of species expanding within the boundaries of life [97].

Genetic Algorithms (GA) is the most widely known and used evolution-based search algorithm. Taking the evolutionary processes that living creatures undergo in nature as an example, GAs simulate the process in which the new generations created by parents and individuals can survive if they adapt to conditions, based on the principle that the living creatures who are suitable for living conditions continue their generations, but those who cannot adapt extinct or evolve.

GAs are heuristic search techniques aiming to reach the result with random search techniques that are based on natural genetics and natural selection, created by simulating the evolutionary genetic structure as a search algorithm in computer environment, where those who provide optimum conditions between generations can survive. In genetic algorithms, each new generation receives the strong characteristics of the parent generation from which it is a descendant through gene transfer. Genetic algorithms, as random and heuristic search techniques, subjects the knowledge of the past to an effectively evaluation for the necessary speculation. Similar to random and heuristic methods in nature, GAs also use a random and heuristic evaluation method alongside coding to solve the problem. GAs use the fitness values of individual chromosomes to perform reproduction. As reproduction occurs, the crossover operator changes parts of the two single chromosomes, and the mutation operator changes the gene value at a randomly selected location on the chromosome. As a result, after several consecutive reproductions, the less-suitable chromosomes disappear, while those that can survive the best gradually dominate the general population [97]. Although the algorithms use a random method, GAs efficiently use their past knowledge to perform various redirects at new search points whose performance is questioned [102].

John Holland, as one of the founders of evolutionary computing, introduced Gas as natural selection and modeling of genetic populations in 1975. Holland observed that the processes observed in nature represent an artificial evolution of the algorithms he was working on. Accelerated his studies on ML, Holland emphasized that the system based on the learning and development of a single structure can evolve into new individuals and generations to be formed by processes such as reproduction, proliferation, and mutation. Holland's GA can be represented by a series of procedural steps for the transition from an artificial chromosome population to a new population. Natural selection uses techniques inspired by genetics known as crossover and mutation. Each chromosome consists of a series of ‘genes’, and each gene is represented in the computer environment as 0 or 1 [94-97-101].

Golberg argued that John Holland and his team supporting GA studies had two motivations for the creation of GAs. The first of these motivations is to simulate the adaptive processes of natural systems by simulating them in an artificial environment and try to explain such processes. The second motivation is to design an artificial software system that maintains the important mechanisms of the natural system. The main theme of researches on genetic algorithms is robustness, efficiency and effectiveness. Establishing the balance between such 3 factors that ensure the creation of the main theme is important in terms of using GAs to solve many different types of problems. If an artificial system has a robust, efficient and effective use, the cost increase caused by redesign and short life systems can be prevented. GAs can offer a low cost, robust and effective search method with all these features they have [102].

Although it is not possible for GAs to reach precise results as they use a heuristic method, they can reach values very close to precise results. In solving a problem, knowing the difficulty level of the problem can help define the paths to the solution and gain information about the result. While solving problems that can be defined with polynomials is simple, solving problems that cannot be represented by polynomials require a long time. Although GAs cannot reach a final solution in the solution of such problems, they use an approximate solution method and approximate solution algorithms that can produce an approximate result. Approximate solution algorithms fail in solving everyday life problems. Although GAs also are included in the class of approximate solution algorithms, solution time is effective in solving problems that increase exponentially with the importance of the problem [94].

The fact that GAs do not need any auxiliary information compared to many other search techniques, and their ability to search with objective function values associated only with single strings makes them stand out among the generally accepted search techniques [102].

> Unlike many methods, GAs use probabilistic transition rules to guide their search. To persons familiar with deterministic methods this seems odd, but the use of probability does not suggest that the method is some simple random search; this is not decision making at the toss of a coin. Genetic algorithms use random choice as a tool to guide a search toward regions of the search space with likely improvement. Taken together, these four differences—direct use of a coding, search from a population, blindness to auxiliary information, and randomized operators—contribute to a genetic algorithm's robustness and resulting advantage over other more commonly used techniques [102].

These features of GAs, which differ from other search and optimization techniques, can ensure that the system created by the algorithms of GAs is a technique with a longer life and a less cost increase. The fact that GAs search all populations of that point instead of a single point and use probabilistic transition rules play an active role in achieving the result even if it is not precise/final.

3.3.5. Computer Vision

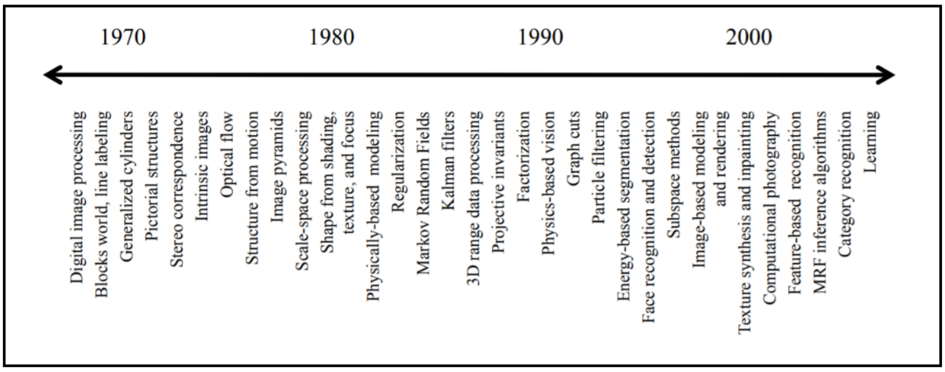
In order for a machine to achieve the abilities that humans have, it must also have the ability to see. Giving a computer or robot the ability to see is provided by processing the image coming from the camera.

The adaptation of human vision, which occurs through the reflection of light, to a machine takes place by mathematical calculations of the three-dimensional shapes of the objects. With these computations, the physical properties (color, shape, reflectivity) of the object obtained from one or more image-frames are re-described in the computer environment. This description causes the vision function, which is easily performed by humans, to become fallible in the computer environment [103].

The camera, which is the main sensor of vision function, can be on the robot or in a suitable place in the environment. The objects in the environment are tried to be recognized through the images coming from the camera. The recognition process can be simple or complex depending on the application. The quality and competence of image processing is determined in accordance with the purpose of the study. Therefore, its contribution to the artificial intelligence system is limited to this determination.

When computer vision studies started in the late 1960s, it first represented the aim of bringing humanoid features to machines. The studies that first started with digital image processing studies have made progress in subjects such as quantitative image and scene analysis with the use of mathematical techniques in the 1980s. In this period, the discovery of many new inventions in the field of photogrammetry as a result of the optimization of camera calibrations, represents the prominent aspect of this period. The 1990s cover the period in which the studies of the previous decade continued and some fields became more important. By the late 1990s, the interaction between computer graphics and computer vision fields increased and studies have been conducted on the subjects such as image-based rendering, image transformation, view interpolation, panoramic image merging. After the 1990s, computer vision studies were revived with the recent ML and deep learning studies. Especially with deep learning algorithms, errors in computer vision have decreased considerably compared to previous techniques [103].

Table 3.1. A timeline of some of the topics of research in computer vision [103].



Computer vision is generally based on the ML system. It is expected that the physical properties of the objects to be seen/perceived by the machine are defined by a series of learning data and at the end of the learning, the machine is expected to detect objects that have the desired properties. It may not be right to mention that it is possible for the machine to obtain all the data belonging to the real world by computer vision. Seeing all data in the real world is generally limited due to the high level of complexity. Computer vision is ensured to perceive only specific areas by means of this limitation.

Computer vision is used today in different fields such as optical character recognition, detection, photogrammetry, medical imaging, motion matching and detection, and biometric matching.

3.3.6. Natural Language Processing

Natural Language Processing (NLP) is an artificial intelligence function that makes sense of the relationship between humans and computers that analyze natural language. Natural language processing techniques make it practical to develop programs that query a database, extract information from texts, taking relevant documents from a collection, translate from one language to another, or recognize the words spoken [52]. NLP is an interdisciplinary field that combines computational linguistics, computational science, cognitive science, and AI [104].

The data sets used in NLP studies differ according to the spoken languages. In this sense, natural language processing technology has a different place from other AI fields in terms of examining modern languages. While natural language affects people's thinking abilities, NLP studies, on the other hand, started with the idea that language can be an important resource for human-computer interaction. However, just like the fact that it is difficult for people with different languages to understand each other, in order for the interaction between computer and human to be achieved with language, both computer and human must be able to hear and even comprehend natural spoken languages. Therefore, in order to use natural languages in computer interaction, this interaction has turned into computerized linguistics studies over time.

Due to the wide variety of syntactic forms of natural languages, NLP studies need to be fed with many sources, including syntactic, lexical and semantics, in order to choose the correct interpretation of a sentence when it is spoken [52]. For a computer to be able to understand human speech and interact with people by speaking in a human-understandable language, the computer must know all the features of language. The NLP technique generally uses ML in the interpretation process.

NLP includes the steps of applying algorithms to define and analyze unstructured language data according to natural language rules so that it is converted into a form by the machine. When a text input is given to a software, the computer uses algorithms to deduce the meaning associated with each sentence and to collect the necessary data from them. In ML, it is often necessary to create a pipeline to deduce a complex meaning. For this, it is necessary to divide the problem into very small parts and then to make sense of each small part separately, and this process requires machine learning.

Typical applications in NLP can be listed as speech recognition, understanding a spoken language, dialogue systems, word analysis, parsing, machine translation, info graphics, information retrieval, answering questions, sentiment analysis, social computing, natural language rendering, and natural language summarization [104]. Instead of perceiving the entire language, NLP is successful at limited issues such as machine translation, database access, information retrieval, text categorization, and extraction data from the text [52]. Humans who can communicate and understand each other through language, which has an important place in human history, can play an important role in the interaction between computer and human with the development of NPL systems.

3.3.7. Robotics

The American Robot Institute (1979) defines a robot as a programmable, multifunctional manager designed to move materials, parts, tools, or specific devices with variably programmed movements to perform various tasks [105]. Based on this definition, the concept of "robot" may be quite far from high-tech creatures with human-like or special abilities, which are reflected in science-fiction films, described in literary and theatrical works from past to present. The definition of the American Robot Institute shows how simple the robot concept can be envisioned.

The word robot originated many years ago, in 1921, before this simple definition of the concept was introduced, from the Czech word "robota", which means "forced labor" or "slavery", that the Czech playwright Karel Čapek used in his play “Rossumovi Univerzální Roboti” (Rossum's Universal Robots, R.U.R.). When the play of Čapek was staged in London in 1923, the audience was affected by the struggle between machine and human and the word robot was transferred to English [68].

The word “robotis” was first used by ythe Russian-born American science fiction writer Isaac Asimov in his short story "Runabout" in 1942. Disagreed with Capek's view of the robot's role in human society, Asimov had more positive opinions for the future of robots and humans. In contemplation of that robots are destructive and that the struggle between humans and robots should be put aside, "I, Robot", which was written as a collection of stories by the science fiction writer Isaac Asimov in 1950 and also included the story "Runabout”, has conveyed the Three Laws of Robotics to a wider audience. These laws were [106];

First law, a robot may not injure a human being or, through inaction, allow a human being to come to harm. Second law, a robot must obey the orders given it by human beings except where such orders would conflict with the first law. Third law, a robot must protect its own existence as long as such protection does not conflict with the first or second laws.

Even though the influence of recent literary and theatrical works on the concept of robots and the imposition of robotic thinking on humans cannot be ignored, first emergence of robots in written, verbal, and physical forms dates back to many years ago.

Around 3000 BC, Egyptians used human-like robots to activate the hour bells of their water clocks. In 400 BC, Archytas, the ancient Greek philosopher, mathematician, astronomer, statesman, and strategist, known as the inventor of the hoop and screw, invented a wooden pigeon that can fly. Hydraulically operated sculptures that can speak, make mimics and gestures were widely used in Greece and Egypt in the 2nd century BC. In the 1st century AD, Petronius Arbiter made a doll that can act like a human. In 1557, Giovanni Torriani created a wooden robot that could bring the Emperor's daily bread from the bakehouse. Robotic inventions reached a relative peak in the 1700s, and during this time numerous impractical automata were created. [107].

The concept of robots dating back to prehistoric times and robotics, which started taking part in literature in the 1950s, appeals to a general use from the simplest electronic devices to industrial devices at the factory level today.

The aspect of robots that involves the AI field is an autonomous robotic system rather than systems working with an assistant support in ordinary daily operations. An autonomous robotic system has a structure that can make its own decisions, just like an autonomous agent, with the help of physical sensors.

Robotic AI systems have some problems they may encounter in real world [52];

- As the sensors react to near elements, it cannot access to the real world.
- In the real world, a robot has to deal with uncertainty.
- In the real world, the effects of actions are dynamic and can change constantly. Therefore, the robot needs to evaluate the timing of its actions.
- The real world is continuous because states and actions are deduced from a range of physical configurations and movements.

Robots can be considered as the mechanical interface of intelligent computers. The concepts of robot and robotics do not directly affect AI technologies and are not considered as an AI field, but can be regarded as an important field of study in terms of revealing a concrete/physical aspect of AI. Intelligent robots have an important place within the scope of artificial intelligence studies. It is possible to attribute this to the idea of making a machine that thinks and acts like a human. Robots respond to many questions in studies trying to achieve such a goal.

In this section of this thesis, a brief history of AI technologies and its problem solving abilities, functioning structures, main approaches and application areas are examined. AI technologies can be used in solving new problems arising from complex systems as an auxiliary tool in reaching optimal solutions. Developing main approaches and application fields provide solutions in solving problems encountered in daily life and in fields such as industry and production that require more advanced technologies. In the following chapter of this thesis, the use of AI technologies as a problem-solving tool in architectural design processes, as in every sphere of daily life, will be examined.

4. AI IN THE ARCHITECTURAL DESIGN PROCESS

The architectural design process, under the influence of the change movements that emerged in all spheres of life with the effect of the modernism movement, has been tried to be evaluated in a systematic structure like all other processes of life. While these systematic structures create the maps of the design process, these maps are far from keeping pace with the dynamic nature of life and from technology getting more involved in life every passing day. Therefore, while design process models, which are currently far from the power of computing and technology, are trying to find answers to design problems with conventional methods, the effective role of computer technologies in design processes and the effects of AI studies, which have been developing since the 1950s, on design process models require different perspectives to design.

In this chapter of this thesis, the possible relationships between AI technologies and architectural design processes are addressed and the SEEK project and the CityMatrix project that involve AI in the architectural design process are examined.

4.1. RELATIONS BETWEEN ARCHITECTURAL DESIGN PROCESS AND AI

The tools used by architects aiming to create a built environment have been constantly evolving throughout history, albeit slowly. The first architectural works, which were created unplanned and instinctively, taking into account the conditions and needs of the period, evolved as "design elements", which were replaced by sketches, two-dimensional drawings, and models, in which a more planned, functional and aesthetic concern comes into prominence. Changes in architectural understanding concentrate on visual elements where the form comes to the fore rather than function, especially in the 21st century. This situation has led to the consideration of more complex forms by architects, and the need for computer-aided software to design and create such forms. However, most of the digital design tools available are limited to predefined commands. The narrow scope of tools put forward for the complex architectural vision encourages designers to create their own algorithms by bringing together the branches of science, which were previously devoted to different scientific fields, with architecture. The hierarchical and complex structure of the design is similar to the evolutionary process and follows a linear and inductive path. In addition, new design techniques created with a heuristic approach allow a non-hierarchical form and architectural vision to be created with a significantly different approach to solving design problems, and unique outputs to be achieved.

The heuristic approach is generally based on assumptions, and a heuristically created form can be created by AI's computational capability of various inputs using a computational method, unlike conventional design. Both algorithmic and data-driven methods have led to the form creation process that is fully adaptable at all stages of design. AI's ability to "achieve complex goals in complex problems" can be used as a design method for architecture and can provide valuable tools for designers.

> The complexity is the result of a developmental process. The complexity acquires its peculiar characteristic in front of the temporality of a developing path, using the stratification of multiple choices in front of the manifold different moments and opportunities. If we want to build the complexity, we must use a multiplicity of extemporaneous and subjective keys of approach, applied on different and, maybe, apparently contradictory fields [108].

The computation in architectural design process is based on hierarchy and automation. The design softwares frequently used by designers support the creative process, nurture the visual senses, but have minimal contributions to design. These softwares are insufficient to provide designers with the opportunity to create their own design tools that fit their needs and habits. Architectural design should be considered as a data processing to achieve the best possible computational results in the complexity of the architectural design process.

When it comes to the beginning of architectural design and/or any design process, it is a complex process that requires a lot of time and effort for the designer to create a product. The factors that are important in creating the optimal design are to determine the general requirements, goals, problem space of the design, to analyze, to direct the general situation information, to collect data and to conduct researches continuously, to conduct case studies, to make predictions about the design, and to prepare alternative solution proposals. When human capabilities are limited, AI's ability to receive unlimited data and solve problems quickly can help the designer to produce more effective design products in shorter times in such long processes.

It was discussed in the previous sections of this thesis that computer programs that have built-in intelligence and can have a large knowledge capacity on a particular subject are called expert systems or knowledge-based systems. Expert systems perform a task done by experts in a particular field and use blind or heuristic knowledge in doing so. The design process is an iterative process in which the same set of computations is performed over and over again. Data from each iteration can be recorded to provide information about the problem that helps find the optimal design.

Another reason AI is used in design is that in fact no algorithm can solve all classes of problems efficiently. However, AI systems that can work efficiently on certain issues can play an important role in problem solving and computation. In addition, designers with limited senses and skills are expected to overcome all challenges of design alone.

Another explanation for the relationship between design and AI can be explained by the enhancement of designers' abilities and intelligence capabilities. The first original idea of augmenting human intelligence was described by Douglas Engelbart in 1962. Douglas Engelbart aimed to develop tools to augment the human mind. The term augmenting the human intelligence can be explained as increasing the ability of people to produce solutions to problems, to have more ideas about complex problem situations and to gain a new understanding by taking into account the needs. Douglas Engelbart defines this as understanding faster, understanding better, the possibility of gaining a useful understanding in a previously very complex situation, producing faster solutions, better solutions and possibilities, finding solutions to problems that seem to have not been solved before. The use of modern technology provides important tools for understanding complex situations, abstracting important factors and providing direct assistance in problem solving. Humans make all their influence on the world through limited motor channels, and these limited sensory channels are based on knowledge received from the outside world. These limited motor resources are not always sufficient to make sense of problems and produce solutions with the use of basic cognitive abilities [109].

The domain of knowledge representation and reasoning within AI has been the cornerstone of most formal AI internal pathways when it comes to problem solving for design [110]. Nigel Cross argues that one of the goals of design research in the field of AI is to provide an understanding of the natural intelligence of design talent. Design is a natural and widespread act among the human population that distinguishes humans from all other living and non-living beings. AI research in design aims to imitate design through interactive systems that aid creativity of the designer. In cases the goal is to develop interactive systems that support designers, knowledge of the human designer's cognitive behavior is clearly essential, since users of the interactive system should be able to use them in a cognitive manner. Hence, systems should be designed based on the cognitive behavioral models of the system users. Another purpose of using AI is to improve human understanding of cognitive behavior by trying to model by computation or imitate the ability of human design. It is possible to learn some facts about the nature of the human design relationship by looking at design from a computational perspective through AI research in design. Instead of imitating human abilities, machines can do the things that designers cannot [111].

Using AI algorithms as a tool to increase human abilities can contribute to the development of design to the same extent. Here, increasing human intelligence with AI is just an abstract definition. The main purpose here is to use the capabilities of AI algorithms in coordination with human design capabilities. Being able to respond to problems in design, which has an interdisciplinary, complex, unpredictable and hierarchical order, can be shown as the most important criteria for optimal design. Although it is possible for the designer to respond to the problems with conventional computation and search methods, the conventional computation and search methods may be inadequate due to the increasing needs, periodic differences and differences in aesthetic understanding. AI systems can be functionalized as autonomous design machines that can draw a project, see the drawing and make decisions according to what they see, bring together the computation vision and the computed project, by means of intelligent agents.

Ill-defined problems are characterized by the lack of goal, problem solving operators, current state and the data required to reach a solution. As addressed to in the second part of this thesis, architectural design is generally referred to as a poorly structured problem, and architectural design is based on search, decision making, and inference paradigms. These three paradigms represent a holistic process that should be applied at every stage of architectural design. Search is a process applied to obtain conditions that meet the target criteria at every step of the design.

In the architectural design process, the search action is not only used to obtain the consequential data of the design product, but also to find the problem itself. The criteria to be investigated in the design process are the reason why the considered design product is needed and the possible human, economic and environmental impacts of this design product.

Decision making is the process of selecting optimal states that meet the specified goals from among the different alternatives encountered. This process covers the entire evolutionary steps required to obtain an optimal product that meets the objectives of the design. In the decision-making process, whether the goals set by the human designer are met is answered with the limited data available to the designer.

The decision-making process in AI is a process of reasoning that represents cognitive processes and logic. The decision-making mechanisms of a software programmed by a human are also programmed by a human. Although this creates a contradictory prediction, the actions of AI programs have a heuristic process, just like the human decision-making process. Therefore, human decision-making mechanism is important for understanding AI decision-making mechanisms. Although AI is generally considered a science that imitates humans, it is actually considered a science of knowledge representation and reasoning. AI can be explained as a system that represents a specific individual only, as decision-making produces subjective consequences. Before making a decision, only the present state is recognized, some of what happened before the present is known, and the system only has its own perception of what is happening now. Then it tries to describe it with reference to subject experience, considering how it perceives the current state. This means that the subject recorded many states that it has met or learned before [112].

AI agents are imprecise to achieve results and are based on probability theory, which is an uncertain system. Agents need utility theory to check the accuracy of these goals while achieving optimal goals. Effective agents can be produced for uncertain situations by combining probability theory and utility theory. Agents can generally access data about their environment through their perceptions, the data they obtained from their current and previous states, but they cannot always have all the necessary data. Probability theory provides an agent's belief basis for all general states. It consists of the evidences obtained from the environment of basic agent and its own knowledge. Utility theory represents the states in which an agent wishes to be, and the decision theory defines the actions of the agent by composing the utility theory with the probability theory. A system having a decision theory represents a rational agent. Rational agents aim to reach the optimal result by considering all possibilities. Decision networks that enable an AI system to make decisions, express and resolve decision problems need the assistance of belief networks in uncertain states. Belief networks are the tools used to represent and reasoning uncertain knowledge.

Unlike simple inference systems, systems with utility theory have decision-making capabilities based on probabilities and data values [52].

It may be taken in stride to consider AI and the design process as two quite different disciplines. However, the structures of architecture and AI that enable interdisciplinary studies ensure that AI and architectural design process are combined. In the following sections of this thesis, the projects that employ AI as a design tool and that exhibit the role it plays in human-machine-design interaction will be addressed.

4.2. RELATED WORKS

In this section of this thesis, the SEEK project and the CityMatrix project, which include AI technologies in the design process, are examined. These projects include AI approaches such as machine learning, robotics, computer vision, and natural language processing.

SEEK is a project created by AMG at MIT in 1970 and demonstrates the machine's ability to manipulate a built environment with the help of computer vision and robotics approaches.

CityMatrix, on the other hand, is a project created within the scope of a thesis study at MIT in 2017 and aimed to use AI technologies as an assistant of the designer in the design of a city section with expert systems, machine learning, natural language processing, and computer vision approaches, supporting a participatory process.

These two projects were chosen to observe the development of artificial intelligence technologies over time, to examine the effects of different artificial intelligence approaches and different intended purposes of AI on architectural design.

4.2.1. SEEK Project

Founded by Nicholas Negroponte and Leon Groisser, pioneers of the MIT Media Lab, AMG has combined architecture with different fields such as AI and computer science and pioneered a number of interdisciplinary studies on how AI can interact with people in architecture. Interacting with MIT's Artificial Intelligence Laboratory, AMG has conducted studies that bring together many different disciplines such as cognitive psychology, AI, computer science, and art. While conducting studies on the common points of AI and architecture, AMG has also been engaged in other fields where "building" and "environment" issues come together. Producing ideas on the dimensional structure of classical architectural design and claiming that the interaction aspect of classical architectural design is weak, AMG worked on some interfaces and projects on the fact that AI could be a tool that can be used for this interaction. Negroponte developed a theory and practice of interaction between humans, computers, and the built environment, published books and articles on the idea of "architectural machines", emphasizing the importance of interaction for architectural design, and that machines can be a part of human life. Going beyond the conventional boundaries of architecture, and combining a computational architectural process with AI, AMG conducted studies on the interaction of machine-living creatures-architecture [6].

SEEK is a project that demonstrates the possible interaction with architecture of the machine that is fed from the real-time actions of the users and was exhibited in the New York Jewish Museum in 1970 by AMG under the direction of Nicholas Negroponte, aiming to make architectural design interactive with machine support. SEEK can be expressed as a blocksworld simulation since it is a closed and limited project.

Figure 4.1. A gerbil in computerized environment [6].

Blocksworld is represented by a plexiglass box, the structures formed with blocksworld are represented by 500 pieces of 5.08 cm (2 inch) metal-coated mirrored cubes placed individually, on top of each other, or side by side, and blocksworld residents are gerbils (desert rats). Blocksworld is equipped with a computer-aided robotic arm with the ability to move and manipulate cubes. Benefiting from the robotics and computer vision fields of AI, the project tries to provide the layouts of the structures in a dynamic environment.

One of the key points SEEK has to deal with is expressing the mismatch between the real world and a world model using a blockworld and to organize this world model [113]. In the SEEK project, which was exhibited in the New York Jewish Museum in 1970, the robotic arm was capable of manipulating the blocks that formed the structures, according to the results of the actions of the gerbils representing the residents of blocksworld. SEEK represents a system that perceives the physical environment, aims to produce solutions for unknown and unexpected events and can make certain manipulations on the environment [114].

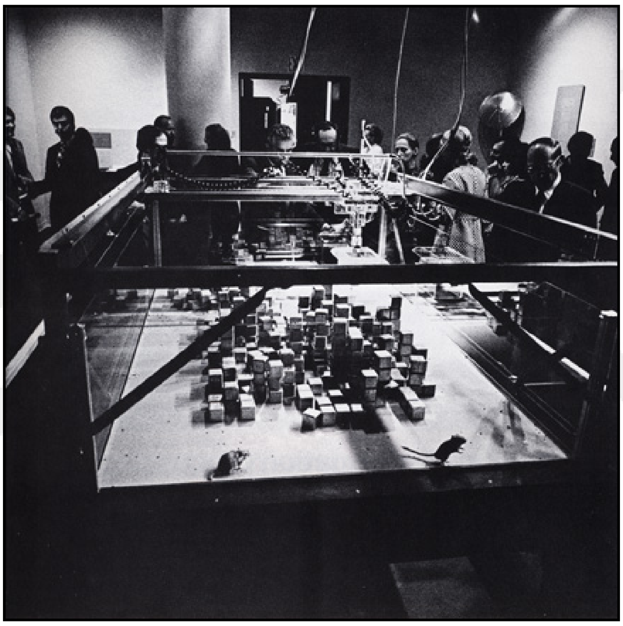


Figure 4.2. The Architecture Machine Group: SEEK, 1970, exhibition "Software", The Jewish Museum, New York 1970 [115].

Exhibiting the interaction of architecture and machinery, SEEK was also demonstrating the effort of an environmental prototype to cope with unexpected events, just like in everyday life, with the help of a machine. In an environment where the actions of the gerbils are almost impossible to predict and in a system having a computer vision, the computer-aided robotic arm must produce solutions for all the actions.

> … Seek is a mechanism that senses the physical environment, affects that environment, and in turn, attempts to handle local unexpected events within the environment [114].

SEEK has a superstructure with a conveyor system with 154.2x243.84 cm (5x8 feet) dimensions, capable of moving in X, Y and Z axes. The robotic arm forming the limb of this structure is equipped with an electromagnet, several microswitches, and pressure sensing devices. This robotic arm is driven by an Interdata Model 3 autonomous computer with 65536 single (yes/no) memory bytes shared with instructions and data to change the positions of cubes and manipulate the movements of gerbils [114].

Negroponte explains the creatures chosen for SEEK as the adaptation of the dynamic movements of curious gerbils to the constantly changing environmental dynamics [115]. Gerbils' constantly changing actions create a dynamic environment, increasing the interaction between machine, living creatures, and architecture. This interaction gains dynamism only by the heuristic actions of the gerbils, independent of a certain order.

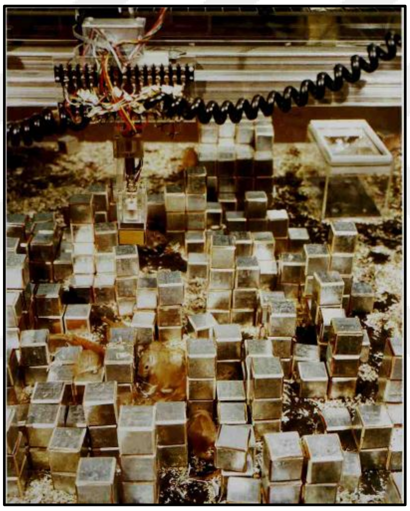


Figure 4.3. The SEEK project consists of gerbils, a robotic arm and blocks manipulated by this robotic arm [116].

> Even in its triviality and simplicity, Seek metaphorically goes beyond the real-world situation, where machines cannot respond to the unpredictable nature of people (gerbils) [114].

Regardless of the presence of gerbils, SEEK keeps the order of the blocks in its memory and when it detects a mismatch between the actual block order and the block order in its memory as a result of the gerbils distorting the block order, it tries to eliminate these mismatches [6]. The interest of SEEK with real world mismatches comes to light at that point. The actions of the Gerbils highlighted the shortcomings of the model in which they lived. SEEK, which does not have a full ML system, can take action to restore the order that changed with the actions of gerbils. However, it lacks the ability to detect, for instance, that the order of the blocks on a route that gerbils constantly use is disrupted by gerbils frequently and to arrange the blocks according to these actions with an ML system.

At the time of its exhibition at the Software exhibition, SEEK reveals that the machines cannot keep up with the dynamic changes taking place in the environment. In a dynamic environment, a machine that can meet human needs should understand human metaphors with AI support and interact in triple interaction against problems that may arise in the relationship between human and environment, which is an unpredictable complex context [114].

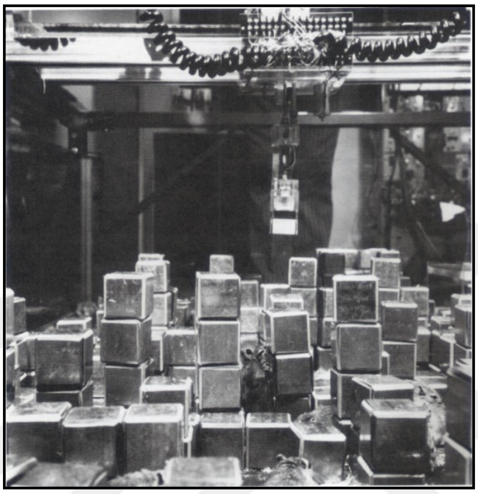


Figure 4.4. The robotic arm, a gerbil and blocks [117].

SEEK shows the failure of the machine that cannot fully adapt to the context of human environment, and machine. Along with the failure accompanied with some other reasons, independent from SEEK, such as the worsening of Software's budget and the failure of the time-sharing computer that supports most of the projects due to software-based problems, the Jewish Museum coming to the brink of bankruptcy, and the censorship of the exhibition catalog, the studies on the project, which was experimental yet, was terminated. SEEK has pioneered other block world initiatives and collaborations at the MIT AI Lab on computer vision and dynamic adaptation, which were seen as timeless phenomena at the Software exhibition [6].

Continuing its computer vision and AI studies later on, AMG has continued its relationship with the MIT AI Lab. AI Lab has begun working on "a practical real world scene analysis system" that tries to make sense of everyday chaos. Marvin Minsky and Seymour Papert reported to ARPA that they work on systems on visual-controlled automated manipulation and physical-world problem solving [118]. Later, under the leadership of Papert, architecture students Anthony Platt and Mark Drazen developed the "Minsky-Papert Eye", a computer-connected video camera that can read certain areas of a block stack for drawing on a CRT (Cathode Ray Tube) terminal [119].

Although it is considered an unsuccessful work due to internal and external reasons and was launched as a work ahead of its time [114], SEEK can be said to be a source of inspiration for future projects. In addition to reflecting the chaos in daily life and the effort to sustain living creatures, which have heuristic actions, in a harmony within the physical environment, it can be said that SEEK also pioneered the studies conducted in computer vision, a branch of AI.

The design process, which ends up with the creation of the design in the architectural design process maps, may not be able to adapt to the daily conditions of life that has a constant and dynamic structure. The solutions of the problems that arise in such cases require a new design process. However, as seen in the case of SEEK, the development of AI systems that can put the design constantly in an order and produce solutions to design problems with its own dynamic systems ensures the completion of the design as a single process.

4.2.2. CityMatrix Project

The CityMatrix project is an urban design support system that includes a natural language guide with real-time environmental data of expert and non-expert participants to design an urban plan with a participatory method and can provide optimized recommendations.

The need for new design and decision-making tools in urban planning is increasing due to the inability of existing design tools used for urban planning to adequately respond to the unforeseen possibilities in complex and dynamic urban life and the compatibility problems between living and non-living urban elements. Following a participatory policy instead of planning existing urban plans and the elements such as traffic caused by these plans, green space needs, building typologies that can affect the climate with a rational method based solely on the city planner's predictions can help urban plans to have a healthy and satisfying structure. When an urban designer’s ability to process many data simultaneously and to achieve an appropriate urban plan based on this data is compared to a machine’s capability of data processing, significant differences come out.

The goals of CityMatrix are as follows: To design a heuristic Tangible User Interface (TUI) to improve the accessibility of decision making for non-experts, to produce real-time feedback of multi-purpose urban performances to help users evaluate their decisions and thus enable rapid, collaborative decision making, to establish a suggestion system that frees stakeholders from excessive, quantitative considerations and allows them to focus on qualitative aspects of the city, thus helping them define and achieve their goals more efficiently [7].

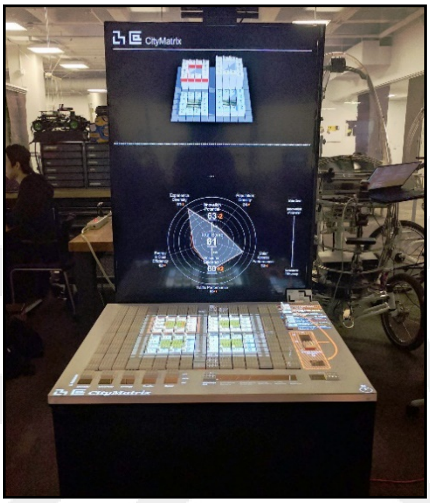


Figure 4.5. Configuration of the City Matrix project [7].

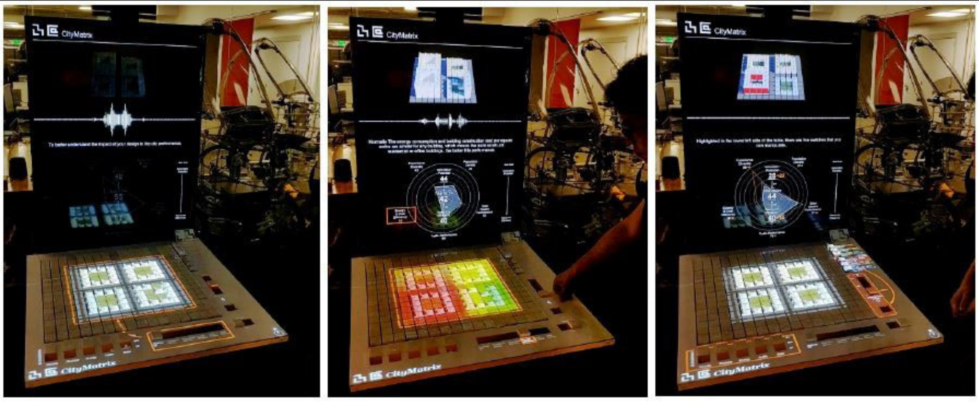
CityMatrix is an urban-plan building prototype with real-time feedback capability. CityMatrix is designed to support a democratic decision-making process that can quickly process versatile data with the support of expert and non-expert participants, find suggestions thanks to AI support, with a low-cost structure. CityMatrix can guide the participant using optimization search algorithms to provide AI suggestions. The changes in the urban performance score due to guidance support the creation of the ideal urban plan (Figure 4.5.).

CityMatrix is a 1:762 scaled model that allows the participants to design an urban area with all inputs with the help of TUI, providing possibilities such as removing, replacing or adding new bricks on a table where building modules represented by optically labeled Lego bricks are available. In the model, each Lego block represents 26.7 x 26.7 meters, and the 5 x 5 Lego block area of the CityMatrix represents 133.3 x 133.3 meters. City dimensions are not realistic and in an abstract structure. The purpose of this abstract city is to isolate the participants from physical conditions and to obtain a raw typological information. In CityMatrix, where the participants can manipulate the optically labeled Lego bricks representing the buildings in the city, enabling each cell to transform into six types of buildings (Small Housing Unit, Medium Settlement Unit, Large Housing Unit, Small Office Unit - OS, Middle Office Unit - OM, and Large Office Unit -OL), roads and courtyards, users can create their own building typologies or use the bricks that are already in a library. The system has a computer vision system to read optical labels. The slider and selection slot on the side of the prototype allows users to change the urban density, while the slider in the middle-upper part helps them to change the building heights. Urban density statistics in the upper part of the table show the buildings of different types and their effects on the urban population in real time by bar graphs [7].

Figure 4.6. Users can change the land use pattern by adding, reducing and modifying optically labeled Lego bricks [7].

With the graphics created by the representation of real-time data, the user has the ability to synthesize possible design solutions at the beginning of the design process and to intervene in the early stages of the design with new problems that may occur in the new design product to be achieved at the end of the design. Thus, all steps of the design process can be fulfilled with optimal decisions.

In the planar and physical part of the prototype, the buildings placed on the urban area by the participants are represented in three dimensions on the vertical screen of the prototype, assisting to determine the building heights. Using the heat map slider at the bottom right of the TUI table, users can select five different heat maps, namely population density, experience diversity, energy and cost efficiency, traffic performance, and solar access performance, corresponding to the determined urban performance. When users rearrange the layout of the urban area, the heat map of the CityMatrix is updated in real-time accordingly. A radar chart is used to demonstrate the urban performance heat maps shown in a vertical viewing screen. Also, a graph showing innovation potential and resource efficiency is provided. While population density and experience diversity contribute to the innovation potential index, they affect resource efficiency, energy and cost efficiency, traffic performance and solar access performance [7]. CityMatrix, which is not only concerned with the structural integrity of the design, but also aims to approach issues such as resource use and experience statistically and in design terms throughout the design process, can therefore address the interdisciplinary aspects of design with the help of the computing power of the machine.



(a) (b) (c)

Figure 4.7. User interface of the CityMatrix: (a) voice assistant, (b) heat map, (c) radar graphic [7].

CityMatrix uses a natural language voice output to convey all the necessary information to participants who do not have sufficient knowledge of urban planning and the system. Besides, the natural language voice output helps the participant to reach all the necessary information at that time. All information conveyed by the CityMatrix Guide are also reflected in writing [7]. CityMatrix Guide guides the participants and enables them to adapt to the system.

AI Suggestion guides CityMatrix users at an optimum level to enable them to make choices for urban texture and building typologies. The suggestions made by the system to improve the urban performance score reflect all the possibilities that may arise as a result of the guidance made via AI support, in real time. Participants can consult AI suggestions for the problems they encounter because they have difficulty making decisions or do not have the necessary cognitive skills. AI Suggestions, while guiding the participants, can sometimes become a factor that complicates the design, rather than an auxiliary system, due to the effort of adding new knowledge to the insufficient knowledge of the participants. In addition to all these, the participant who cannot fully comprehend the current states may be vulnerable to the suggestions made by AI. For users who cannot see the distinction between the current states and the suggestions, the current states and the suggested states are expressed in different colors [7].

AI suggestions also include a series of design learning that is created as a result of statistical data obtained via ML about the environment and the models made by the users involved in the design. For this reason, while developing AI suggestions for each new design they make, users can update their opinions about the design.

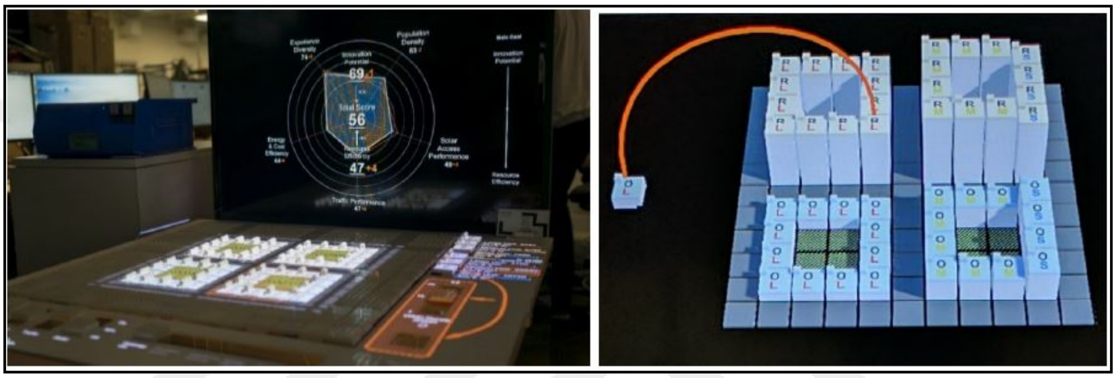


Figure 4.8. A participant can always get support from AI suggestions. AI suggestion is shown in real time with orange color [7].

AI Suggestions are arranged in the system in a way that can be set by the user gradually, and AI Suggestions update themselves according to the selected level. For instance, the user can set AI Suggestions on, off, or with 50% support. This is called Artificial Intelligence Support Weights. The user can change the Artificial Intelligence Support Weights at any time of the design [7]. This ensures that a help mechanism is always available to which the user can refer when the design is paused.

While making an urban planning in CityMatrix, which is a part of a Python server and consists of a strategy and evaluation component, the strategy component plans the possible movements, and the evaluation component ensures that the urban design being created is evaluated. After the strategy and evaluation components define the optimum movement, a suggestion output is provided to the visual interface. While the evaluation component scores the urban design, the AI considers the support weights. The search algorithm applied in the strategy component aims to find the most suitable one among the possible 1966 possibilities in CityMatrix with search algorithms. In this narrow search area, a search can be concluded by evaluating 150 to 500 possible movements. AlphaGo forms the basis of the search in CityMatrix due to its general learning strategy. Currently, the AI suggestion searches only the next movement [7]. CityMatrix, where search strategies are applied limitedly, searches for the next step instead of searching all possible steps and thus can only help a specific and limited design process instead of guiding the user completely. Besides, since searching the entire design process of the system may cause high memory and time consumption, the search can be applied in a limited search area.

Provided as a versatile, fast, accurate and low-cost approach, CityMatrix benefits an ML algorithm to learn the design from an urban performance simulation and to predict the result real-timely. ML was preferred because of its versatile structure, speed, accuracy and low development cost. ML algorithms trained for the CityMatrix project are versatile because they can process different parameters such as solar radiation, traffic density, and population density, and develop with the help of different parameters, are fast because they constitute a system that can process data quickly after the training process is completed, have a high level of accuracy since the data generated can always be crosschecked and this can be performed internally, and have low development costs as the algorithms can be trained autonomously [7].

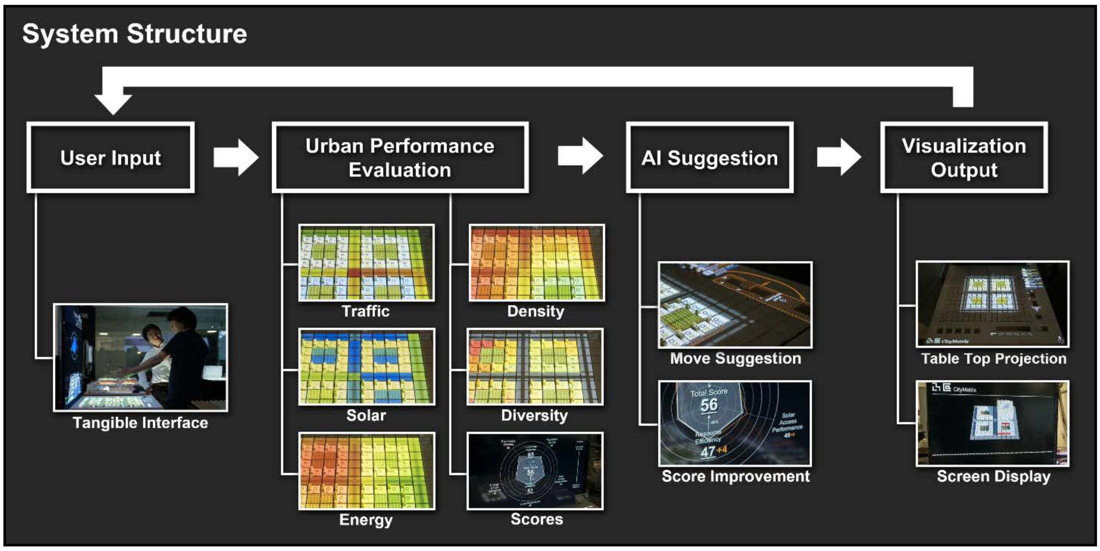


Figure 4.9. System structure of the CityMatrix [7].

CityMatrix has acquired some data beyond the created urban plans. These data are mostly about the problems caused by machine-human interaction. The use of a machine in the design and the existence of a machine that can make suggestions establish a ground for the suggestions offered to the user to be accepted and approved. While the accuracy of the suggestions made by AI that moves the urban score upward is not certain, they also have an impact on the user's own heuristic ideas. Since CityMatrix represents an abstract city on a scale, it becomes very difficult for the user to grasp the urban context, causing the user to try to create a city with only a high score. Besides, the prototype's lack of suggestion mechanism creates a negative effect for the user. In addition, AI suggestions, which are far from specific ideas, can score high on creating an ordinary city, while the system has difficulty responding to unusual design experiments [7].

CityMatrix can help users who do not have experience and knowledge on urban planning to gain awareness of urban context. The system also aims to raise awareness about understanding the effects of small actions on cities. Thanks to its AI systems, CityMatrix can assist the designer in reaching a new and optimal result in every design step, especially those made with the help of ML. The project, which can achieve a new result from each new design, learn and thus bring the design process to an effective and real-time structure, can show the importance of the role of the participatory system and computation support in the future of design processes.

4.3. EVALUATION OF DEVELOPMENT OF AI TECHNOLOGIES IN ARCHITECTURE

Human thought is dynamically formed with experiences and interactions with concepts of different fields. The computation of the thought formed is based on a set of complex computations that resulted at the end of evolutionary processes. Although such computation is relatively successful in struggling many different cases encountered in life, it has a consistently evolving structure such as dynamics baffling with the effect of different concepts that are completely dependent on external factors. Complex systems consisting of dynamic structures are formed by aggregation of multiple sub-factors. For solving new and sometimes unprecedented problems formed by a complex system, different computation methods may be required. A dynamic computation method may have a rapid decision-making capability for each new and unprecedented case. However, in cases not experienced before and have no common concept with past experiences in baffling dynamics, an intuitive approach is taken.

Humans encountering a dynamic and unpredictable problem primarily envisage an abstract representation of the problem to solve it and then try to estimate possible solutions to that problem over such abstract thought. This situation is also expressed by a similar representation process for AI as discussed in the third chapter of this thesis. This representation process consequentially abstracts the problem, thus ensures making an inference, producing solutions, and making a reasoning.

The machine approach, on the other hand, is dependent on training data provided especially with the machine-learning method, and the cases not included in the training set are indefinite for the machine. However, imitating the skill of learning from human experiences, machines can process new data and produce solutions to the problems in the next encounter based on such data. The dependence of current machine intelligence mainly on the training data can be argued to be one of the most significant barriers before new and innovative processes, as it restrains the provision of necessary flexibility. Humans are skilled in subjective value judgments such as culture, aesthetics, and sense, unlike machines. Where humans remain incapable, the machines can efficiently and sensitively make complex numerical computations with multiple targets simultaneously.

The use of AI technologies in architecture can be considered as an effort to include architecture in a statistical field. With the reduction of architecture to an abstract and mathematical basis, the computability aspect of every action the designer performs heuristically can be revealed. Founded by McCarthy in 1956 and started with the idea of using the human mind as a model for a machine, AI studies allow the computer to use the data it collects from the environment or through user inputs as a parameter. Beyond the data acquired by the machine, the occurrence of machine learning in the light of such data and the occurrence of the decision making, reasoning and solution producing processes with the help of a process similar to experience shows that machines can use the human mind as a model.

Table 4.2. shows the technologies used by SEEK project and CityMatrix project as well as their design methodologies and space limitations Using this table, in this section, the features and capabilities of the projects are re-evaluated and the approaches of the two projects are compared.

AI studies, which were at a certain momentum in the 1970s, focused on what a machine can achieve autonomously, and it is possible to suggest that the ideas about AI were considered promising in such years. SEEK is a project that aims to demonstrate what a machine can do alone, and perhaps examines that a machine can be superior to human intelligence. The capabilities of SEEK and its relative success have also pioneered further projects.

AI studies, which were at a certain momentum in the 1970s, focused on what a machine can achieve autonomously, and it is possible to suggest that the ideas about AI were considered promising in such years. SEEK is a project that aims to demonstrate what a machine can do alone, and perhaps examines that a machine can be superior to human intelligence. The capabilities of SEEK and its relative success have also pioneered further projects.

Table 4.1. Comparison chart of SEEK and CityMatrix projects, created by Author.

		SEEK	CityMatrix
Design Process Methodology	Descriptive		✓
	Prescriptive	✓	✓
Approaches and Application Areas of AI	Machine Learning		✓
	Expert System		✓
	Computer Vision	✓	✓
	Natural Language Processing		✓
	Robotics	✓	
Space Limitation	Blocks world	✓	✓
	Real world	✓ (Limited)	

Negroponte mentions that the computers can perform successfully when the desired result can be clearly communicated and when they have a system that can learn and infer conclusions from the data. Negroponte further argues that the use of machines in architecture was needed in order to improve performance because architects could not solve very large and complex problems and ignore small problems [120].

Here, there are two issues regarding SEEK. The first of these is that, with the idea of exciting intelligent machines, which emerged as a result of the early years when AI studies gained importance, studies were conducted based on the fact that the machines could be qualified as a subject rather than a tool. The second is that AI technologies did not develop enough at the time of their creation. SEEK represents a structure dependent on the machine's absolute truths and was designed with a system based on the unquestioned implementation of all actions of the machine. However, considering that the machine ignored the gerbils in the prototype and took actions based on the block layout only, it can be argued that living creatures, which are the most important element of the dynamic problems of daily life, are far from interacting with machines. The fact that the machine manipulates living creatures with its effects only on inanimate objects does not comply with the heuristic decisions of daily life and creates a prescriptive structure. That the system is autonomous, has limited monitoring capacity, and does not have a control panel or interface for humans causes all decisions of the machine to be considered as mandatory truths. SEEK has conducted a research on the potential of such systems rather than creating intelligent algorithms. At this point, it would not be correct to mention about an effective use of the computer vision of SEEK. Computer vision is used only to restore the deteriorated order. However, the system, which is incapable of following the actions of the gerbils, begins to repeat the same actions over time. For achieving this, an ML algorithm integrated with computer vision is required. Only through this can the actions of the gerbils be observed and the design can be reorganized based on these actions.

When evaluated in terms of the design process, SEEK adopts a prescriptive method as it is an engineering project as a design methodology. Hence, the truths of the design are precise and unique. The occurrence of another block layout that cannot be evaluated as true will not be possible for the project, and this situation can make the interaction of a spatial project with living creatures into a prescriptive structure and make it fail.

SEEK also brought the interaction between machine and living creature to a limited real world domain. The project, which can be described as a blocksworld, can interact with living creatures in a limited area in real time.

In cases where the design process models cannot predict, it can be mentioned that, as in SEEK, its structure can be said to be far from the actions of living creatures with which a certain design layout interacts. The design model created with a systematic structure is likely to predict the living conditions in a systematic structure. However, constantly changing and developing vital activities may over time exceed the limits of architectural design and this situation may render architectural design dysfunctional.

Created about half a century after SEEK, the CityMatrix project brought machine-human interaction to the fore, unlike SEEK. It also adopted a participation process in order to make the design process that takes place between designer and design more democratic.

The CityMatrix project, on the other hand, considers machine and human as two separate forces and aims to combine these forces. While the AI system focuses on quantitative criteria, it aims to make humans focus more on values and qualitative criteria [7]. CityMatrix represents an expert urban-planning system with AI approaches such as machine learning, natural language processing, and computer vision. These approaches can be regarded as sufficient features for CityMatrix alone to create an urban plan. However, the democratic and participatory process adopted in the project enables designers and non-designers to use the machine as a design partner. While ML algorithms can become experts on certain urban plans after a certain training period, they also encourage the user to create an urban plan with high score parameters. Thanks to its NLP capability, it can audibly guide the user and follow the changes on the platform in real time with a computer vision.

While the CityMatrix project has a prescriptive design methodology as it is an engineering project, it has a descriptive structure since the created design products have a structure that is far from precise accuracy. While it is not possible to mention that the urban plans produced with CityMatrix create only truths, the design products created with high scores aim to create an optimal truth. Besides, for instance, in cases such as the creation of a non-standard urban plan, some problems may arise in the system's design product scoring.

CityMatrix is a prototype created as blocksworld outside of real world scales. Therefore, it cannot be expected to adapt to the real world and to struggle with the major problem spaces of the real world. In addition, the users may not have any opportunity to experience results in the real world and in real time, even in very confined spaces.

Another problem faced by CityMatrix was argued to be "the user's acceptance of AI suggestions without query". This case is based on the assumption that when a person does not have sufficient knowledge, he/she will be heuristically indifferent to directions and can reach a conclusion with machine interaction, as a new experience.

To mention the differences between the two projects, the SEEK project represents a project that is prepared for users beforehand and that forces users to live according to the determined layout, but at the same time, it is believed to be able to make some arrangements that can respond to the actions of living creatures, and the users of the space (gerbils) can experience the system in real time.

In CityMatrix project, on the other hand, the users create the physical environment themselves and use AI as a guide while creating this physical environment. At the same time, it can comprehend the physical environment created on the simulation with the ML algorithms of artificial intelligence with the help of analysis in virtual environment, but there is no real-time experience. CityMatrix allows each individual to intervene in their own living environment like a designer. When creating living spaces, the data that a designer will obtain about the area is often lacking in experience. For this reason, previously designed plan will not satisfy the user and, therefore, the user's experiences should be reflected in the designs. CityMatrix provides users with the ability to decide on the shape, regulations, types of settlements and even population density for a city segment. While providing these features, the CityMatrix offers a fast system with low development costs for non-professionals.

Limited problem spaces to focus on specific problems and produce answers to those problems by reducing variables are called “microworlds” or “blockworlds”. A micro world is an artificial domain in which all possible objects, properties, and events are clearly predefined within a narrow-scope [44].

The basis of AI studies is to be able to respond to problems in a generally constrained environment, because researching a problem is much simpler than reaching the right conclusion or any conclusion in large search domains such as the real world with an infinite number of variables. The trend in the evolutionary development of all systems follows a direction from simplicity to complexity, and the method of operation of such systems tends from the specific towards the more abstract. When more complex systems are examined, whether biological or electronic, understanding and dealing with such complexity require more abstract concepts [89].

Studies in the field of architecture generally include a prototype of limited domains. The SEEK project and the CityMatrix project can be defined as blocks world. Since the Blocks world generally represents a limited domain where all variables are known, the expected results can be evaluated hypothetically different from the real world. On the other hand, unlike blocks world, it contains a large number of unpredictable dynamic variables and the problem space is quite wide.

> The fact that a program can find a solution in principle does not mean that the program contains any of the mechanisms needed to find it in practice [52].

Most AI studies are applicable because only micro-worlds contain very few objects and subjects. They need to combine different combinations until the truth is found by representing the basic facts about a problem and trying a series of steps to solve it, and indeed, the number of different combinations in the real world can be infinite. It may not be a right approach to talk about the success or applicability to the real world of a system tested with blocks world, which reflects an abstracted prototype of a hypothetical world.

It was mentioned in the second chapter of this thesis that one of the reasons for the systematic nature of the design process systems is the effort to transform design into an easier process by subjecting the design process to certain limitations. At this point, the efforts to incorporate design process maps and AI technologies into the design process lead to a similar problem. A systematic process may require the designers to limit their intellectual, creative and artistic abilities fed by experience, knowledge and culture. A design that is handled by adhering to certain limitations and rules can be expected to be unique at a certain level and to be able to respond to problems at a certain level.

Also, with the incorporation of AI technologies into the design processes, it should be known that systems have limits on their computing and learning capabilities. These limits can be determined by considering the costs of memory and time. While it is not certain that every designer can achieve a conclusion at the end of the design process, it is not certain either that these systems can always achieve the optimal solution. The methods of design process maps and the capabilities of intelligent machines incorporated into the design process can respond to limited problems in limited domains. As in the design process maps, the incorporation of AI technologies into the architectural design process can make the systems, which yet did not reach the level that can produce optimal solutions to real-world problems with today's technology and knowledge, cause problems in producing solutions in the domains that cannot be limited. This can ultimately result in an obstacle rather than an advantage before unlimited intellectual, artistic, and creative abilities of the designer.

5. EPILOGUE

This epilogue focuses on the possible effects of AI technologies on the architectural design process. These effects are thought to have various impacts on the design process as well as on the architects.

This epilogue is a prediction rather than an outcome due to the ongoing development of AI technologies in architectural design. For this reason, the aim of the epilogue is to make the purpose of the research far from defending a dogmatic reality. It is thought that the use of AI techniques in architectural design processes will help obtain new perspectives and evaluate a possible new design hypothesis.

5.1. POSSIBLE FUTURE EFFECTS OF AI TECHNOLOGIES ON ARCHITECTURE

In this section, the use of AI technologies in design is evaluated from two main aspects:

- Possible changes on architects and in the professional role of architects
- Possible effects on the architectural design process.

The classical tools used by architects to create their design products cannot show sufficient performance in solving the design processes that are a parallel outcome of the age and technological developments and which are constantly evolving into a more complex and dynamic structure. Design, the architect's most important and powerful ability, has always been an internal problem-solving tool. However, the continuous evolution of the problems that need to be solved creates the necessity of new tools that can adapt to the dynamic nature of the problems in creating a design product. The use of AI technologies in architecture can have various effects on the role of the architect. However, with today's technology it does not seem possible for these effects to be so powerful that architects or designers are not needed anymore. Negroponte points out machine and architecture as two separate forces. Human represents design while machine represents computation [121].

Design can be regarded as a simple act in the rapidly growing universe of AI technologies. However, the fact that design combines universal and ethnic structures is the biggest obstacle before AI technologies. AI is able to perform the quantitative processing of various subjective and objective factors comprehensively and produce a large number of "creative" solutions in a shorter time than architects. It may not be correct to define about the creativity of a design product created with the help of an intelligent machine as artistic and intellectual. The creative solution here reflects only the results of a statistical decision. However, these results must have an evaluation process. The evaluation of a machine-made design product involves a statistical and mathematical logic and evaluation process based on mechanical laws. In this evaluation process, abstract concepts such as aesthetic judgments and cultural values cannot be evaluated with intellectual knowledge. The process to be intervened by the designer is a subjective evaluation, and this subjective evaluation expresses the designer's personal values and his/her representation of the values of a community. It may not be a correct approach to mention that a machine-produced design product represents an unconditional truth. A design product lacking a cultural and aesthetic value, but statistically and logically true may not be sufficient to satisfy the user.

Another change that will take place in the role of the architect with the use of AI technologies in architecture is the provision of the necessary data to AI systems. It would not be a correct approach to mention that AI technologies can be designers. A designer collects a lot of data while solving a specific problem using classical design methods and must make a statistical and heuristic decision about which of these data will help him/her achieve the optimum result. Such heuristic decisions are the decisions that contain general world-view/knowledge and are fed by the culture of life. Providing the necessary and optimum data to a design product to be created with AI systems may also be possible with the support of a designer. Otherwise, the design products can be said to be far from satisfactory results. However, at that point, it should be mentioned that machines can produce unique and exceptional results. The structure of machines that can focus on even small details compared to human designers and suggest different products from the general phenomena of the human designer should not be ignored [121].

The changed nature of architectural design tools has led to various changes in the role of the architect. In the 20th century and before, architects had to conduct more studies on design and construction than today's architects. However, in the following periods, the idea of digitizing architecture with computer technology was introduced, as conventional design tools could not support the desire to produce more with speed and less-cost, which are the necessities of the era. This idea primarily led to the emergence of CAD software. Design products created with CAD software are a gateway to the next era of architecture with both cost-reducing and speed features. The architects who cannot adapt to CAD systems had opportunities to work in different areas of the profession. However, on the other hand, it is possible to mention that AI systems will reduce employment. Architects, who have gained long-term experience in conventional design and construction systems, which are the basis of architecture and design, but cannot adapt to new design systems, have turned to areas that require experience and cognitive skills such as project and construction consultancy. The same also applies to the development of BIM systems. Learning any CAD or BIM software, anyone who is not an architect or designer can create a project. How, in fact, are architects and non-architects distinguished?

Investigated user satisfaction problems that may arise from differences in the cognitive and physical foundations of ideas between professional architects and non-professionals, Ghomeshi et al. (2013) examined the aesthetic evaluation of building facades by architects and non-architects. Although the research revealed similar ideas about facade surface design, they concluded that there are significant differences between designers and non-designers in the preferences of physical cues in ideal designs. They also stated that the architect could act as a bridge between design and the opinions of non-architects [122].

Yazdanfar et al. (2014) compared the space perception of architects and non-architects in the evaluation and interpretation of architectural spaces. During the conduct of their study, they evaluated differences of opinion in different aspects such as cognitive, emotional, interpretive, and appreciative aspects. At the end of the study, while the architects made evaluations in cognitive and interpretative aspects, whereas the non-architects came into prominence with their emotional and appreciative evaluations. It was also found out that the visual literacy degree of architects is higher than non-architects [123].

According to Negroponte, what distinguishes a talented, competent designer is the ability to provide his/her incomplete knowledge. While any environmental design process can be characterized by unavailable or uncertain knowledge, the design process can be summarized as the provision of such knowledge. Some part of this knowledge can be acquired by researches during the preliminary design stages, while some part can be acquired by a combination of experiences. Other knowledge can be acquired through induction, prediction, and even ironic attitudes that appear at unexpected times [119].

Studies and statements made to reveal the differences between architects and non-architects can also help in revealing the differences between human and machine. Looking at the differences between architects and non-architects, it is seen that architects focus more on physical and cognitive clues and consider design with an interpretative perspective. In a design to be created with machine intelligence, the machine intelligence can create a design product that is far from an aesthetic, artistic, and intellectual interpretation process based on statistical data. However, this situation may take the design away from real world values. An architect can adapt a design process to a general world-view with all experiences of her/him.

It does not seem possible today for AI technologies to have the power to eliminate the need for architects. Experienced architects may always be needed due to their construction experiences and their skills to bring their education to an aesthetic structure blended with cultural and vital values. The actual point to be mentioned here is that machines use a skill-enhancing subject, that is, the designer and the machine can jointly manage a process.

> By virtue of ascribing intelligence to an artifact or the artificial, the partnership is not one of master (intelligent, leader) and slave (dumb, follower), but rather of two associates which each have the potential for self-improvement [121].

According to Negroponte's statement above, the joint work of the machine and the architect will represent a beneficial process for both sides. It can be mentioned here that the emergence of the machine's own power rather than the machine being a structure under the command of the human may produce more functional results. Besides, through mutual learning and comprehension methods i, the architects can become more skilled in seeing the clues hidden in smaller details or ignored by people.

Negroponte considers the architect as an unnecessary, cumbersome, and even harmful intermediary in an order where needs are constantly incorporated into the built environment, and states that it is correct that it undertakes the work that the machine and human are good at [115]. Carrying out a joint design process with a machine requires more than the computing power of the machine. When a machine supported by architects encounters a problem with a set of exceptional rules, it seeks solutions by comparing that problem with similar situations in its knowledge base. It can also follow a heuristic method with inferences from similar situations just like architects do. When a situation is encountered again and again, it can preserve the current situation to use these solutions when similar situations are encountered later using the solutions of such situations stored in its knowledge base [121]. The machine also needs to evaluate or at least observe the goals and results. Knowledge may become less important over time and eventually eliminate exponential forgetting. Or past procedures may not meet environmental conditions that change over time, thus overriding a heuristic, rote response or conditioned reflex [120]. Because of these similarities, the support of a fully intelligent machine in the design process as an architectural partner can help solve design problems and produce more satisfactory results.

The use of machines in architecture was needed to increase performance. These shortcomings can be eliminated with adaptive and learning machines that liberate the designers and give them more time to do what they really love [120]. The fact that architects can devote time to working in different fields other than their own may lead to the emergence of new professional roles in architecture. This professional role may appear as future "architectural AI system designers" of architects who have mastered computer skills in addition to conventional architectural skills.

The second chapter of this thesis includes a research on design and design process approaches, and as mentioned in the later chapters of this thesis, conventional design process approaches are far from providing satisfactory answers to today's design problems.

The process, which started with the emergence of new searches and standards in almost every field that is a part of life and which is a result of the effect of modernism movement, has also caused new searches and models in production and industry. While the process that started with Taylorism at the beginning of the 20th century and then continued with Fordism, which emerged partially based on Taylor's principles, revitalized systematic production, it also played a role in mass production gaining importance. In parallel with this acceleration experienced in industrialization with mass production in the 20th century, it is also possible to mention about a tendency towards systematization and mechanization in architectural and construction technologies. The introduction of industrial and machine-aided production in almost all spheres of life has also paved the way for the ideas of using machines in architecture. Particularly, the World War II, which took place in the middle of the 20th century, can be said to have had effects on material technologies and design [1-2-13].

The new needs that emerged in the post-war period increased the need for new building types and technologies in the architectural field, and parallel to this, the level of complexity architects had to deal with also increased. Most of the design process maps examined in the second chapter of this thesis are from the post-war period.

In addition to the effects of industrialization, with the understanding of the importance of intelligent machines that played an active role especially during war, systematization of knowledge has started to gain importance. In these times, contrary to their conventional structures, the architectural design researches have been in an evolutionary process in which knowledge can be processed in a virtual environment and even sketches, which are the most basic design elements, can be created in a virtual environment. While this virtual environment is represented by computer technologies, these technologies can be considered as a tool that relieves architects of increased complexity. Negroponte explains the benefits of the machine in the design process as follows:

> There are three possible ways of having machines assist in the design process:
>
> - Current procedures can be automated, thus speeding up and reducing the cost of existing practices.
> - Existing methods can be altered to fit within the specifications and constitution of a machine, where only those issues are considered that are supposedly machine compatible.
> - The process, considered as being evolutionary, can be introduced to a mechanism (also considered as evolutionary), and mutual training, resilience, and growth can be developed [121].

With the first generation Computer-aided design (CAD) software used in architectural practice, computer environments have been created which can be described as attractive with their production speed advantage, as a need of that era, and through which the hand drawings of the architects can be represented. Later on, CAD systems were further developed as 3D and 4D software and are still actively used today. Building Information Modeling (BIM) softwares created after CAD systems represent the second generation of design computation in the architectural field. BIM can be defined as "knowledge management and complex relationships between social and technical resources that represent the complexity, cooperation, and interrelationships of today's organizations and the environment thereof" [124]. Like the increase in the complexity of building systems after World War II, it has led to an evolution in architectural modes of production, the building systems information digitized through extensive and easily-connected information networks has created a need for systematization and organization of this new information.

By the end of the 1960s, with the beginning of the creation of computer-based design systems, the computer started being considered as a design companion that can be a partner to designs, where ideas can be conveyed with natural language, and can provide systematic support to design, rather than being a design tool [115].

The computation tools can be said to have transformed into intelligent machines with developed AI technologies. It can be said that the assumption of the creation of systems that know how to access all the information about the design task at hand, can distinguish between states and phenomena, and can produce complete design solutions without the need for an architect to control the process using AI systems, is a new era for architectures. AI technologies will make it possible to undertake projects in a shorter time frame. The ability to integrate advanced analysis and optimization techniques into the design process will enable complex and subtle design constraints to be maintained.

Many researches have been conducted on classical design processes in order to achieve a systematic structure of the design, to get it evaluated in various classes, to predict the design problem, and to create a more reliable process. Nigel Cross et al. point out that conventional design is not methodologically suitable for systematization [125]. The fact that systematizing the design process itself is not the right method can be shown as evidence that there will be a faster transformation towards the use of the computer as a design companion.

Classical design process research involves several different steps supervised by the designer. While the controller was solely the architect in classical design systems, especially with the development of BIM softwares, the architect started to get support from the power of a computer system. The systems created in the following periods, on the other hand, made an effort to create systems that can create designs and produce solutions to problems without the need for an architect.

As discussed in the fourth chapter of this thesis, it would not be correct to mention about the success of AI systems. However AI studies can mechanize some of the difficult problems currently encountered in the design of more powerful problem-solving procedures [126]. With the development of AI systems, they can be used as a support in design processes that can increase the abilities of the designer. This situation requires classical design processes to be updated.

Negroponte argues that as architectural technology and intelligent computer systems develop in parallel witj each other, there is more development possibility for a "sensitive architecture" in which any adaptable environment can arouse with machine intelligence [115]. Classical design processes basically involve a cyclical process consisting of analysis, synthesis, and evaluation steps. Here, the only person who has a say in analysis, synthesis and evaluation is the designer/design team. The participant-supported system used in CityMatrix allows a design currently evaluated by a computer to be finally evaluated by an architect as well. Currently, the architect tends to play a central role in coordinating the activities and design inputs of all individuals in the other design team. Automation will be able to continuously provide coordination as a tool that can make this collaboration of intelligent systems more effective. This participatory-based and coordinated process can make living in and designing a building an almost equivalent process. With a new architectural design process, where certain experiences can be gained at the design stage and the results of these experiences can be simulated in real time, design can play an important role in analyzing new problems that the product may cause. Experience and design can have a common structure with real-time evaluation of the design process.

The constant cross-questioning of ideas by both human and machine with the use of AI technologies will encourage creative thinking, which will disappear due to the absence of an opposing environment. Computer aided design is associated with mutual design completion and development. [121]. At that point, the biggest change that will take place in the design process can occur with the involvement of the machine in the process.

The design process, with the use of AI technologies, can be systematized as in Figure 5.1., created with the information researched and obtained throughout this thesis. The model created in Figure 5.1., shows the inputs conveyed to a problem space from the designer and AI system.

Out of the inputs conveyed to the problem space, the rectangle drawn by dashed lines represents the architect, while the rectangle drawn by a continuous lines represents the AI system. The reason underlying this is the assumption that the architect can always have a real-time interaction with the culture, experience, and qualities of the environment due to his/her openness to the outside world. It is also assumed that the AI system may have limited and statistical data. Another input to the problem space is provided by the participant. The participant is shown with a dashed circle in this model.

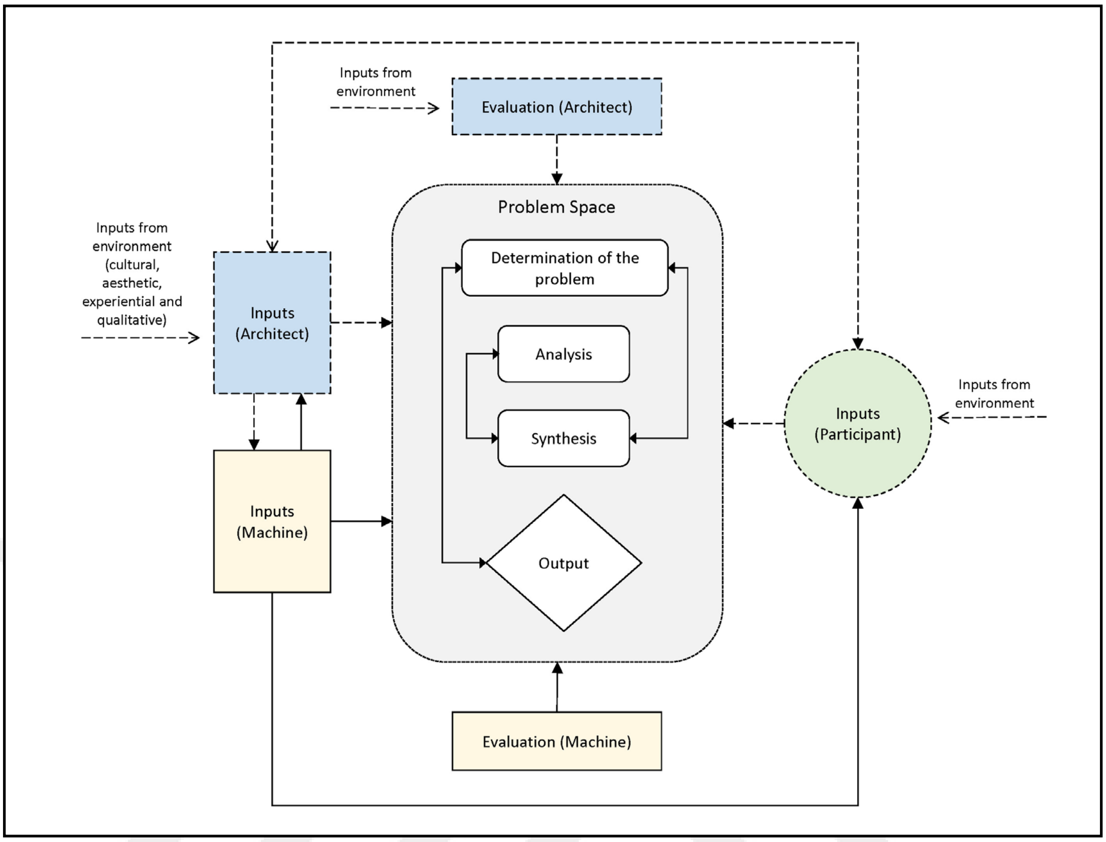


Figure 5.1. Machine-architect-participant interactive design process model, created by Author.

The reason for this is that the participant does not have certain knowledge with defined boundaries. Besides, the participant is in interaction with the architect and the machine as the participant can have a certain level of unilateral trust in the architect's and the machine's suggestions. It can be mentioned that a similar interaction exists between the architect and the machine. However, this interaction between the architect and the machine includes the method of transferring and comparing information and updating existing information. With this interaction, the architect can expand his architectural vision with the information he obtains from the machine, while also updating the existing data of the machine.

The inputs gained from the machine, architect, and participant stelled in the model have similar values at the beginning of the design process and can cause major and minor effects on the design. While statistical data provided by the machine complements an architect's shortcomings, it can be the participant's support in identifying areas that the architect has ignored. With the participatory process, the design product can achieve a satisfactory structure for the end-user.

Determination of the problem in the problem space is made through analysis and synthesis processes. Analysis and synthesis can be repeated constantly, and there can also be a relationship between synthesis and problem definition. The output at the end of these processes represents the design product. There is a constant interaction between the design product and the defined problem. The output should satisfy the problem definition and be able to respond to the problem space.

The inputs processed into the problem space require the auditing of the entire process of defining problem, obtaining analysis, synthesis, and product, that is, the whole problem space. While this control mechanism only represents the designer/architect in the previous design process maps, whereas, in this newly created map the design process can be controlled by the machine and the designer/architect. An AI system can collect environmental data with the help of artificial neural networks with the help of ML and computer vision, and partially manipulate the design by learning some design elements with the inputs of the architects. With a cross-learning ability, ML systems can grasp the design ideas of the architect and the participant and thus provide the necessary data for new and even original design solutions. It can also provide optimal solutions with GAs. It can respond to limited and defined problems in limited areas with today's technology by creating an expert system. ANN's support can provide new and unexpected approaches to design by imitating the role of a designer in contrast to these limited and defined solutions. These new approaches can create a new architectural understanding that the designer could not predict. This situation may cause positive developments on the aesthetic and artistic perspective of the designer. A system that can process natural language and interact with the architect or participant using natural language can play an important role in machine-human interaction. With this interaction, machine and human can create a new approach as two separate elements of a joint work rather than the elements that control each other. A robotic system fed from AI systems in a prototype study or an architectural design study close to real scales can autonomously guide the design in line with the needs. All these capabilities can transform AI systems into a structure that can play a role in design and be included in evaluation processes. The architects can also play an important role in the supervision process with their experiential, artistic, intellectual and cultural knowledge.

As with the inputs, the evaluation phase is also bidirectional, and the evaluation step shown with continuous lines represents the machine, while the designer/architect who is in constant interaction with the outside world is expressed in dashed/permeable lines. The prominent point here is that the participant is included in the input processes but not included in the evaluation processes.While the data obtained from the participant can play an important role in creating the design, the evaluation of the final design product includes a number of statistical, intellectual and experiential skills. Hence, the evaluation process can be evaluated by the machine and the architect, thus an optimal result can be expected.

This model was not created to systematize the design or to prove that previous design process maps were insufficient. The important point here is to establish a link between the design process and the machine-human interaction and to show how the participant can be involved in an architectural design. With the methods intended to be shown in this model, the design problem and the solution methods may not be different from classical design processes. However, the architect may no longer have to deal with the problems of collecting data, processing that data, and making inferences alone.

The model created in Figure 5.1. represents a model that can be created with current technology in which the architect is always involved in a design process. However, with the developing technology and new approaches to AI systems, the architectural design process can develop as an independent field from the architect. Here, the ability of new models created with ANN studies to produce independent and unexpected design products may come to the fore. This situation may require questioning the validity of the design rules accepted as correct with the conventional design approach. The design process, which presently represents an unpredictable process, may result in the emergence of "striking" products. This can make the statistical, and abstract aspects of the design process come to the fore. With the developing technology, it can be predicted that the artistic, intellectual and creative aspects of design can be modeled systematically with a statistical method.

6. CONCLUSIONS

The idea of considering design as a process rather than focusing on the final design product for the creation of design products that satisfy the needs brings along the necessity of the design process to keep up with technological developments in order to develop more effective design products.

Officially named in 1956 and defending the purpose of creating humanoid machines, AI studies started to be used effectively especially on automation systems and the idea of using these technologies in architecture and design emerged similarly with all other fields. The idea underlying the use of AI technologies in architecture is to finalize the computable aspects of architectural design with AI technologies that have advanced computational capabilities in a shorter time and with optimal results.

The SEEK project and the CityMatrix project that employed AI technologies in architecture are examined in the fourth chapter of this thesis. These experimental projects, which were created in different periods and technologies, can provide new insights about the use of AI technologies in architecture. The necessity of the design process to keep up with technologies causes AI technologies to have certain effects on architectural design processes. The combination of the skills of architects and machines in different fields can play an active role in solving the design problems caused by the dynamic variables of daily life, and this may have the effects that are stated in this thesis on design processes and architects. Based on all these researches, the AI-supported design systems that are created have two main problems that make it difficult to investigate the effect of architectural design tools. The first problem is that intelligent computers are likely to alter the design process elements and the roles of architects partially. The second problem is that it is not completely possible for everyday life to set up experimental systems that try to provide the practical experience and feedback required to test new systems. New methodologies to be developed as a result of the introduction of AI technology and the concept of design automation in general can profoundly change today's style and assist in creating a new understanding of architecture in the future. Technological developments and changes that may occur in design methodologies may result in the autonomous systems to produce aesthetic, intellectual and artistic architectural design products. These studies, which can highlight the computable aspects of architecture, can show that intellectual, artistic and aesthetic concepts can be computed by passing through certain processes.

REFERENCES

1. Schröter HG. *Americanization of the European economy: a compact survey of American economic influence in Europe since the 1880s*. Dordrecht: Springer; 2005.

2. Boyer R, Durand J-P. *After Fordism*. London: Macmilan Press LTD; 1997.

3. Harvey D. *Postmodernliğin durumu*. Istanbul: Metis Yayınları; 1997.

4. Cross N. *Engineering design methods: strategies for product design*. Milton Keynes: John Wiley & Sons LTD.; 2000.

5. Lawson B. *How designers think*. Oxford: Elsevier Architectural Press; 2005.

6. Steenson MW. *Architectural intelligence: how designers and architects created the digital landscape*. Cambridge: The MIT Press; 2017.

7. Zhang Y. CityMatrix – an urban decision support system augmented by artificial intelligence. Master Thesis, Massachusetts Institute Of Technology, Cambridge; 2017.

8. Arcan EF, Evci F. *Mimari tasarıma yaklaşım*. Istanbul: Tasarım Yayın Grubu; 1999.

9. Foster N. Yaşamak için tasarım. In: *Çağdaş dünya mimarları dizisi 7 Norman Foster*. Istanbul: Boyut Yayınları; 2000:119–127.

10. Weijnen M, Herder P, Bouwmans I. Delft science in design 2. *Designing complex systems: a contradiction in terms*. Amsterdam: Ios Press; 2008:235–254.

11. Bayazıt N. *Endüstri ürünlerinde ve mimarlıkta tasarlama metodlarına giriş*. Istanbul: Literatür Yayınları; 1994.

12. Taylor FW. *The principles of scientific management*. New York: Harper and Brothers; 1911.

13. Yentürk N. Post-Fordist gelişmeler ve dünya iktisadi işbölümünün geleceği. *Toplum ve Bilim*. 1994;56(61):43.

14. Bayazıt N. Tasarım araştırmaları. In: Er HA, Er Ö, Eds. *Endüstriyel tasarım eğitimi:*

İTÜ endüstriyel tasarım toplantıları. Istanbul: Endüstri Ürünleri Tasarımı Bölümü; 1999:22–28.

15. Bayazit N. Investigating design: a review of forty years of design research. *Design Issues*. 2004;20(1):16–29.

16. Jerrard B, Newport R, Trueman M. *Managing new product innovation*. London, Philadelphia: Taylor & Francis; 1999.

17. Jones JC, Thornley D, Eds. *Proceedings of the conference on design methods*. Oxford: Pergamon Press; 1963.

18. Jones JC. A method of systematic design. In: Jones JC, Thornley D, Eds. *Proceedings of the conference on design methods*. Oxford: Pergamon Press; 1963:10–31.

19. Hillier W, Musgrove J, O'sullivan P. Knowledge and design. In: Cross N, Ed. *Developments in design methodology*. Chichester: Wiley; 1984:245–265.

20. Cross N, Roozenburg N. Modelling the design process in engineering and in architecture. *Journal of Engineering Design*. 1992;3(4):325–337.

21. Cross N. The design process. *Engineering design methods: strategies for product design*. Milton Keynes: John Wiley & Sons LTD.; 2000:29–42.

22. Markus TA. The role of building performance measurement and appraisal in design methods. In: Broadbent G, Ward A, Eds. *Design methods in architecture*. London: Lund Humphries; 1969:109,117.

23. Lawson B. Route maps of the design process. *How designers think*. Oxford: Elsevier Architectural Press; 2005:38–39.

24. French MJ. The anatomy of design. In: *Conceptual design for engineers*. Springer Berlin Heidelberg; 1985:1–4.

25. Archer LB. Systematic method for designers. In: Cross N, Ed. *Developments in design methodology*. Chichester: Wiley; 1984.

26. Rowe PG. *Design thinking*. Cambridge: MIT Press; 1987.

27. Pahl G, Beitz W. *Engineering design*. London: Springer; 1984.

28. VDI-Fachbereich Produktentwicklung und Mechatronik. *Systematic approach to the design of technical systems and products*. Düsseldorf: Verein Deutscher Ingenieure. Report number: VDI 2221; 1984.

29. March L. The logic of design and question of value. In: March L, Ed. *Architecture of form*. Cambridge: Cambridge University Press; 1976.

30. March L. The logic of design. In: Cross N, Ed. *Developments in design methodology*. Chichester: Wiley; 1984.

31. Archer LB. The structure of the design process. In: Broadbent G, Ward A, Eds. *Design methods in architecture.* New York: G. Wittenborn; 1967:76–77.

32. Newell A, Simon HA. *Human problem solving*. New Jersey: Prentice-Hall, Inc.; 1972.

33. Jonassen DH. Toward a design theory of problem solving. *Educational Technology Research and Development*. 2000;48(4):63–85.

34. Kalay YE. *Architecture's new media - principles, theories, and methods of computeraided design*. Cambridge, Massachusetts: The MIT Press; 2004.

35. Wagter H, Vries M de. A CAAD model for use in early design phases. P*roceedings CAAD futures digital*. Boston, USA; 1989:215–228.

36. Lattuca L. In the eye of the beholder: defining and studying interdisciplinarity in engineering education. *Proceedings of the 117th Annual Conference & Exposition of the Association for the Study of Engineering Education*. Louisville, USA; 2010:1-24.

37. Alexander C. *Notes on the synthesis of form*. Cambridge: Harvard University Press; 2002.

38. Jones JC. *Design methods: seeds of human futures*. London, New York: Wiley-Interscience; 1970.

39. Lawson B. *What designers know*. Oxford: Elsevier Architectural Press; 2004.

40. Lawson B, Dorst K. *Design expertise*. Oxford: Architectural Press Oxford; 2009.

41. Gil N, Tommelein I, Kirkendall R. Modeling design development in unpredictable environments. *Proceedings of the 2001 Winter Simulation Conference*. Arlington:

IEEE Press; 2001:515–522.

42. Simon HA. *The sciences of the artificial*. Cambridge, Mass.: MIT Press; 1969.

43. Bellman RE. *An introduction to artificial intelligence: can computers think?* San Francisco: Boyd & Fraser Publishing Company; 1978.

44. Haugeland J. *Artificial intelligence: the very idea*. Cambridge: Massachusetts Institute of Technology; 1985.

45. Kurzweil R. *The age of intelligent machines*. Cambridge, MA: MIT Press; 1990.

46. Rich E, Knight K. *Artificial intelligence*. New York: McGraw-Hill; 1991.

47. Charniak E, McDermott D. *Introduction to artificial intelligence*. Massachusetts: Addison-Wesley; 1985.

48. Winston PH. *Artificial intelligence*. Massachusetts: Addison-Wesley; 1992.

49. Schalkoff RI. *Artificial intelligence: an engineering approach*. New York: McGraw-Hill; 1990.

50. Luger GF, Stubblefield WA. *Artificial intelligence: structures and strategies for complex problem solving*. Redwood City: Benjamin/Cummings; 1993.

51. Schank RC. What is ai, anyway? *AI Magazine*. 1987;8(4):59–65.

52. Russell S, Norvig P. *Artificial intelligence: a modern approach*. New Jersey: Prentice Hall Press; 2009.

53. Turing AM. Computing machinery and intelligence. *Mind*. 1950;59(236):433–460.

54. Crevier D. *AI: the tumultuous history of the search for artificial intelligence*. New York: Basic Books, Inc.; 1993.

55. Moor J. The dartmouth college artificial intelligence conference: the next fifty years. *AI Magazine*. 2006;27(4):87.

56. Rosenblatt F. *The perceptron: a perceiving and recognizing automaton*. Cornell Aeronautical Laboratory, Inc. Report number: 85-460-1; 1957.

57. McCarthy J. History of LIPS. In: *History of programming languages*. New York: Association for Computing Machinery; 1978:173–185.

58. McCarthy J. Programs with common sense. In: Minsky M, Ed. *Semantic information processing*. London: MIT Press; 1968:403–418.

59. Slagle JR. A heuristic program that solves symbolic integration problems in freshman calculus. In: Sharples M, Hogg D, Torrance S, Young D, Hutchinson C, Eds. *Computers and thought*. Cambridge: MIT Press; 1995:191–203.

60. Evans TG. A heuristic program to solve geometric-analogy problems. In: Fischler MA, Firschein O, Eds. *Readings in computer vision*. San Francisco: Morgan Kaufmann; 1987:444–455.

61. Minsky M, Papert S. Artificial intelligence progress report. [cited 2020 26 August]. Available from: https://dspace.mit.edu/bitstream/handle/1721.1/6087/AIM-252.pdf?sequence=2.

62. Waldrop M. *The dream machine: J.C.R. Licklider and the revolution that made computing personal*. New York: Penguin Books; 2001.

63. Raphael B. *SIR: a computer program for semantic ınformation retrieval*. Massachusetts Institute of Technology. Report number: AITR-220; 1964.

64. Bobrow D. *Natural language input for a computer problem solving system*. Massachusetts Institute of Technology. Report number: AIM-066; 1964.

65. Zadeh LA. Fuzzy sets. *Information and Control*. 1965;8(3):338–353.

66. Zadeh LA. Fuzzy logic: computing with words. *IEEE Transactions on Fuzzy Systems*. 1996;4(2):103–111.

67. ON-line system. [cited 2020 26 August]. Available from: https://www.darpa.mil/about-us/timeline/nls.

68. McCorduck P. *Machines who think: a personal inquiry into the history and prospects of artificial intelligence*. Natick: AK Peters Ltd; 2004.

69. Weizenbaum J. *Computer power and human reason: from judgment to calculation*.

San Francisco: W. H. Freeman & Co.; 1976.

70. Norvig P. *Paradigms of artificial intelligence programming: case studies in common lisp*. San Francisco: Morgan Kaufmann Publishers Inc.; 1992.

71. Moses J. Macsyma: a personal history. *Journal of Symbolic Computation*. 2012;47(2):123–130.

72. Greenblatt RD, Eastlake DE, Crocker SD. The Greenblatt chess program. *Proceedings of the November 14-16, 1967, Fall Joint Computer Conference*. New York: Association for Computing Machinery; 1967:801–810.

73. Levy DNL, Ed. *Computer games I*. New York: Springer-Verlag; 1988.

74. Cohen H. The further exploits of AARON, painter. *Stanford Humanities Review*. 1995;4(2):141–158.

75. McDermott J. R1: an expert in the computer systems domain. In: *Proceedings of the first AAAI conference on artificial intelligence*. Menlo Park, CA: AAAI Press; 1980:269–271.

76. Kavallieratou E, Del-Blanco CR, Cuevas C, García N. Retrieving events in life logging. [cited 2020 18 April] Available from: http://ceur-ws.org/Vol-2125/paper_86.pdf.

77. Tegmark M. *Life 3.0: being human in the age of artificial intelligence*. London: Penguin Books; 2017.

78. Bostrom N. *Superintelligence: paths, dangers, strategies*. Oxford: Oxford University Press, Inc.; 2014.

79. Descartes R. *Discourse on method*. New York: Macmillan; 1986.

80. Mitchell M. *Complexity: a guided tour*. Oxford: Oxford University Press, Inc.; 2009.

81. Simon HA. The architecture of complexity. *Proceedings of the American Philosophical Society*. 1962;106(6):467–482.

82. O'Connor T. Emergent properties, [cited 2020 22 December]. Available from: https://plato.stanford.edu/archives/fall2020/entries/properties-emergent/.

83. Poole L, Mackworth K. *Artificial intelligence: foundations of computational agents*. Cambridge: Cambridge University Press; 2010.

84. Garro A, Mühlhäuser M, Tundis A, Mariani S, Omicini A, Vizzari G. Intelligent agents and environment. In: Ranganathan S, Gribskov M, Nakai K, Schönbach C, Eds. *Encyclopedia of bioinformatics and computational biology*. Oxford: Academic Press; 2019:309–314.

85. Frost RA. *Introduction to knowledge base systems*. London: Macmillan Publishing Co.; 1986.

86. Hayes RF, Waterman D, Lenat D. *Building expert systems*. Reading, Massachusetts: Addison-Wesley; 1983.

87. Nilsson NJ. *The quest for artificial intelligence*. Cambridge: Cambridge University Press; 2009.

88. Brachman R, Levesque H. *Knowledge representation and reasoning*. San Francisco: Morgan Kaufmann Publishers Inc.; 2004.

89. Firebaugh MW. *Artificial intelligence: a knowledge-based approach*. Danvers: Boyd & Fraser Publishing Co.; 1988.

90. Davis E, Marcus G. Commonsense reasoning and commonsense knowledge in artificial intelligence. *Communications of the ACM*. 2015;58(9):92–103.

91. Mitchell TM. *Machine learning*. New York: McGraw Hill; 1997.

92. Hurwitz J, Kirsch D. *Machine learning for dummies*. Hoboken, NJ: John Wiley & Sons, Inc.; 2018.

93. Rosebrock A. *Deep learning for computer vision with python - starter bundle*. Philadelphia, PA: PyImageSearch; 2017.

94. Elmas Ç. *Yapay zeka uygulamaları*. Ankara: Seçkin Yayıncılık; 2011.

95. Hebb DO. *The organization of behavior*. New York: Wiley; 1949.

96. Jackson P. *Introduction to expert systems*. Boston, MA: Addison Wesley; 1998.

97. Negnevitsky M. *Artificial intelligence : a guide to intelligent systems*. Edinburgh Gate, Harlow: Addison-Wesley; 2005.

98. Pelletier FJ. Review of metamathematics of fuzzy logics. *The Bulletin of Symbolic Logic*. 2000;6(3):342–346.

99. Zimmermann HJ. *Fuzzy sets, decision making and expert systems*. Dordrecht: Kluwer Academic Publishers; 1987.

100. Zadeh LA. The birth and evolution of fuzzy logic (FL), soft computing (SC) and computing with words (CW): a personal perspective. In: *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A. Zadeh*. Singapore: World Scientific Publishing Co., Inc.; 1996:811–819.

101. Holland JH. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control and artificial intelligence*. Cambridge: MIT Press; 1992.

102. Goldberg D. *Genetic algorithms in search, optimization, and machine learning*. Addison: Wesley Publishing Company; 1989.

103. Szeliski R. *Computer vision: algorithms and applications*. Berlin: Springer-Verlag; 2010.

104. Deng L, Liu Y. *Deep learning in natural language processing*. Singapore: Springer Publishing Company; 2018.

105. Sciavicco L, Siciliano B, Sciavicco B. *Modelling and control of robot manipulators*. Berlin: Springer-Verlag; 2000.

106. Asimov I. *I, robot*. Garden City, New York: Doubleday; 1950.

107. Fried J, Davydov E, Pa W. Robotics: a brief history. Robotics and Motion Planning, [cited 2020 7 February]. Available from: https://cs.stanford.edu/people/eroberts/courses/soco/projects/1998-99/robotics/index.html.

108. Soddu C, Colabella E. Artificial intelligence and architectural design, [cited 2019 14 April]. Available from: http://www.soddu.it/papers/1995_Artificial intelligence and Architectural Design.pdf.

109. Engelbart DC. *Augmenting human intellect: a conceptual framework*. Washington D.C.: Stanford Research Institute; 1962.

110. Bhatt M, Freksa C. Spatial computing for design: an artificial intelligence perspective. In: Gero JS, Ed. *Studying visual and spatial reasoning for design creativity*. Dordrecht: Springer; 2015:109–127.

111. Cross N. Natural intelligence in design. *Design Studies*. 1999;20(1):25–39.

112. Pomerol J-C. Artificial intelligence and human decision making. *European Journal of Operational Research*. 1997;99(1):3–25.

113. Negroponte N. The return of the Sunday painter. In: Dertouzos ML, Moses J, Eds. *The computer age: a twenty-year view*. Cambridge: MIT Press; 1979:21–37.

114. Jack Burnham, Ed. Gerbils in a computerized environment: SEEK. *Software: information technology: its new meaning for art*. New York; 1970:20–23.

115. Negroponte N. *Soft architecture machines*. Cambridge: The MIT Press; 1975.

116. Goodman C. *Digital visions, computers and arts*. New York: H. N. Abrams, Inc.; 1987.

117. 1969-70 – seek – nicholas negroponte (american), [cited 2020 14 August]. Available from: http://cyberneticzoo.com/robots-in-art/1969-70-seek-nicholas-negroponte-american/.

118. Minsky M, Papert S. *Proposal to ARPA for research on artificial intelligence at MIT, 1970-1971*. Cambridge: Massachusetts Institute of Technology; 2004.

119. Negroponte N. *The architecture machine*. Cambridge: The MIT Press; 1970.

120. Negroponte N. Toward a theory of architecture machines. *Journal of Architectural Education*. 1969;23(2):9–12.

121. Negroponte N. Towards a humanism through machines. *Computational design thinking: computation design thinking*. Chichester: John Wiley & Sons; 2011:78–85.

122. Ghomeishi M, Jusan M. Investigating different aesthetic preferences between architects and non-architects in residential facade designs. indoor and built

environment. *Indoor and Built Environment*. 2013;22(1):925–964.

123. Yazdanfar SA, Heidari AA, Aghajari N. Comparison of architects' and non-architects' perception of place. *Procedia - Social and Behavioral Sciences*. 2015;170(1):690–699.

124. Jernigan FE. *Big BIM, little BIM : the practical approach to building information modeling : integrated practice done the right way!* Salisbury: 4Site Press; 2008.

125. Cross N, Christiaans H, Dorst K, Eds. *Analysing design activity*. New York: John Wiley and Sons; 1997.

126. Amarel S. On the mechanization of creative processes. *IEEE Spectrum*. 1966;3(4):112–114.