REPUBLIC OF TÜRKİYE

ALTINBAŞ UNIVERSITY

Institute of Graduate Studies

Architecture

A CYBORG'S PERCEPTION OF THE CITY

Mahad Mohamed Elhadi IMHEMED

Master's Thesis

Supervisor

Asst. Prof. Dr. Can UZUN

İstanbul, 2024

A CYBORG'S PERCEPTION OF THE CITY

Mahad Mohamed Elhadi IMHEMED

Architecture

Master's Thesis

ALTINBAŞ UNIVERSITY

2024

The thesis titled A CYBORG'S PERCEPTION OF THE CITY prepared by MAHAD MOHAMED ELHADI IMHEMED and submitted on 11/06/2024 has been **accepted unanimously** for the degree of Master of Science in Architecture.

Asst. Prof. Dr. Can UZUN
Supervisor

Thesis Defense Committee Members:

Asst. Prof. Dr. Can UZUN	Department of Architecture, Altınbaş University	____________
Asst. Prof. Dr. Erinç ONBAY	Department of Interior Architecture and Environmental Design, Altınbaş University	____________
Assoc. Prof. Dr. Orkan Zeynel GÜZELCİ	Department of Interior Architecture, İstanbul Technical University	____________

I hereby declare that this thesis meets all format and submission requirements for a Master's thesis.

I hereby declare that all information/data presented in this graduation project has been obtained in full accordance with academic rules and ethical conduct. I also declare all unoriginal materials and conclusions have been cited in the text and all references mentioned in the Reference List have been cited in the text, and vice versa as required by the abovementioned rules and conduct.

Mahad Mohamed Elhadi IMHEMED

ABSTRACT

A CYBORG'S PERCEPTION OF THE CITY

IMHEMED, Mahad Mohamed Elhadi

M.Sc., Architecture, Altınbaş University,

Supervisor: Asst. Prof. Dr. Can UZUN

Date: June/2024

Pages: 91

When the algorithms of reinforcement learning (RL) blend with architecture and urban design, a 'cyborg' construct that entails cybernetic principles of feedback, can be used to find adaptable and responsive designs that are better suited for users, as user preference, user habitual behaviours, and social and environmental conditions are continuously changing. With methodologies motivated by RL algorithms, architects can use them in Unity's ML-Agents toolkit, containing agents that have the capability to learn from their positive and negative actions in environments, in an alike manner to how human beings would, through simulation studies that imitate real-world set-ups that may be too intricate to study using conventional techniques, by examining curiosity-driven behavioural patterns and preferences in agents to discover design solutions that apply to real-life. Using RL algorithms in ML-Agents mimics real-world settings where cyborg explorers, agents with a flâneur attitude on exploration, can use their curiosity about environments to be motivated in navigating and learning about it, offering architects the opportunity to learn about real-world behavioural patterns through the perceptions of a cyborg explorer. This thesis' simulation study examines the influence of curiosity as a motivational drive on a cyborg explorer's behavioural patterns in one of Istanbul's urban settings, with a focus on the Sultan Ahmet Camii neighbourhood.

Keywords: Behavioural Exploration, Cyborg, ML-Agents, Motivational Curiosity, Reinforcement Learning.

ÖZET

BİR CYBORG'UN ŞEHİR ALGISI

IMHEMED, Mahad Mohamed Elhadi

Yüksek Lisans, Mimarlık Ana Bilim Dalı, Altınbaş Üniversitesi,

Danışman: Dr. Öğr. Üyesi Can UZUN

Tarih: Haziran/2024

Sayfa: 91

Takviyeli öğrenme (RL) algoritmaları mimari ve kentsel tasarımla harmanlandığında, sibernetik geri bildirim ilkelerini içeren bir 'cyborg' yapısı, kullanıcı tercihleri, alışkanlık davranışları, sosyal ve çevresel koşullar sürekli değiştiğinden, kullanıcılar için daha uygun, uyarlanabilir ve duyarlı tasarımlar bulmak için kullanılabilir. RL algoritmalarıyla motive edilen metodolojilerle, mimarlar Unity'nin ML-Agents araç setinde, geleneksel tekniklerle incelenemeyecek kadar karmaşık olabilecek gerçek dünya kurulumlarını taklit eden simülasyon çalışmaları yapabilirler. Bu çalışmalarla ajanların merak odaklı davranış kalıplarını ve tercihlerini inceleyerek, insanların yaptığına benzer şekilde ortamlardaki olumlu ve olumsuz eylemlerden öğrenme yeteneğine sahip ajanlar oluşturabilirler. ML-Agents'da RL algoritmalarının kullanılması, keşif konusunda flâneur bir tutuma sahip olan cyborg kaşiflerin, ortamlar hakkındaki meraklarını gezinme ve öğrenme konusunda motive olmak için kullanabilecekleri gerçek dünya ortamlarını taklit eder. Bu sayede, mimarlar bir cyborg kâşifin algıları aracılığıyla gerçek dünyadaki davranış kalıpları hakkında bilgi edinme fırsatı bulurlar. Bu simülasyon çalışması, Sultan Ahmet Camii mahallesine odaklanarak, İstanbul'un kentsel ortamlarından birinde bir cyborg kâşifin davranış kalıpları üzerinde, merakın motivasyon bir dürtü olarak etkisini incelemektedir.

Anahtar Kelimeler: Cyborg, Davranışsal Keşif, ML-Agents, Motivasyon Merakı, Takviyeli Öğrenme.

TABLE OF CONTENTS

LIST OF FIGURES

Pages

ABBREVIATIONS

ABM	:	Agent-Based Modelling
AI	:	Artificial Intelligence
AMG	:	Architecture Machine Group
ML	:	Machine Learning
NN	:	Neural Network
RL	:	Reinforcement Learning

1. INTRODUCTION

As advancements in different fields are significantly affecting the realm of architectural design, this work focuses on the future of architecture as a topic of interest. Architectural design has always been human-centric with its priorities siding with human-experiences above all else. Designing for user preferences and needs while being inclusive and socially interactive is what architects aim to prioritise. Designs associated with technology-enabled solutions have the potential to respond dynamically to users' needs as well as accommodate urban changes. In order to acknowledge that, we must first acknowledge the potential of the computer as a utopian design tool that aims to democratise architectural design.

Technology has been present since the early 1960s, as it was used for numerous purposes including as a drawing and calculating tool in the 60's and 70's, a designing tool in the 80's and 90's, later used by designers as a storytelling tool, and lastly as a flexible interface for different types of interactions and communication uses. Architects have continuously recognized the potentials that technological machines offer as they primarily used them for simple visualization tasks and drafts (Negroponte, 1969).

Reinforcement learning (RL), a sub construct of machine learning (ML), is the focal algorithm of this thesis, the term being used to explore intelligent behaviours in architectural approaches within an inception of artificial intelligence (AI) (Sutton & Barto, 1999). As a field that emerged in the 1950s, it explored to mirror all dimensions of learning in machines and presenting the field of cybernetics with challenges. And despite its negative contemporary connotations as a comprehensive term for intelligence related aspects, the term was initially motivated by aspirations for machines to imitate human abilities and behaviours, aspirations possibly associated with the representations of the modern cyborg. A cyborg is an acronym for a "cybernetic organism", which is an entity that consists of both biological and technological components, the concept originated from a 1960's paper written by Clynes and Kline titled "Cyborgs and Space" (Clynes & Kline, 1960).

The possibilities of technology go far beyond that, as Negroponte states that architecture machines "won't help us to design; instead, we will live in them" (Negroponte, 1975: 5). Nicholas Negroponte coined the term 'architecture machine' as a 'thinking aid' for architects to come to more adaptive and responsive design outcomes (Negroponte, 1970). Negroponte proposed that they can be used to model and analyse user behaviour and ultimately better design through machine intelligence. He considered that “A computer is the only machine we can use to model human behaviour” (Negroponte, 1975: 194).

The idea of ‘architecture machines’ emerged during the 60’s, ultimately as a response to the abrupt advancements in technology and the belief that cities should be able to respond dynamically to social, environmental, and economic changes. They represent vessels that realize the principles of a multi-interactive platform, which is abundantly adaptive to fluctuating situations of associated space and time (Negroponte, 1970).

The projects discussed in this thesis, such as Gordon Pask’s Colloquy of Mobiles (1968), Nicholas Negroponte’s Architecture Machine Group experiments (1967-1985), Cedric Price’s Fun Palace (1961), and Peter Cook’s Plug-In City (1964), were thought to be forward-thinking and ahead of contemporary trends. These conceptual structures were designed as a vision for flexible environments that had the potentials to adapt to future changing conditions with the help of technology, and as a result would encourage responsive architecture. Price’s Fun Palace theorized the technique of a temporary structure that can be modified and taken apart according to the user demands it continuously receives. Gordon Pask states that the Fun Palace is a model system for a cybernetic design prototype, as it references architecture as an interactive and continuously progressing system ordered by the user’s feedback, instead of simply viewing it as a building (Pask, 1969). Archigram (1961-1974), an architectural group that was formed in the 60’s, imagined and abstracted interchangeable and transformative structures that could essentially be plugged into existing urban layouts where they propose short-term solutions that did not require any permanent constructions. Peter Cook’s Plug-In City (1964) theorized the concept of a mesh-like arrangement with the spaces designated for plugging in flexible modular pods as a solution for the constantly fluctuating population proportions, urban conditions, or user’s lifestyle choices. Another Archigram project that proposed similar ideas is Walking City (1964), which projected a self-motivated type of urban structure composed of mobile and self-sufficient pods with the ability to walk on legs, mirroring the concept of an independent walking machine housing inhabitants instantaneously adjusting to urban and societal changes. Dennis Crompton’s Computer City (1964) is a city designed as a complex system of intersected links (traffic, good, people, etc.), and was actually abstracted for Plug-In City to work. And although these projects, and many more, were visionary and ‘ahead of their time’, because of the lack of technological resources and application availability, they were merely perceptions for the future and to say that these plans are not possible now, or in the near future, would be a misconception.

With chief technological innovations such as progresses in computational power and the readiness of massive datasets, fuelling the evolvement of algorithms like ML, and within that RL, perpetually advancing, remarkably in the subject of architectural design. Despite the promising advancements in ML and within that, the growing paradigm of RL, their applied integration of these technologies for generating innovative digital architecture tools have yet to materialize and happen. Architects and urban designers can use these RL-based algorithms to come up with environmental designs that transcend mere virtual simulations of urban environments in isolation. Essentially, with the aid and involvement of the cyborg concept, alongside feedback mechanisms of cybernetics and RL, there is potential for interactive and responsive architecture design.

Of the many potentials of interactive and responsive architecture design includes introducing adaptability into buildings and spaces, allowing them to react to changing circumstances, whether in environmental conditions or user preferences. In addition, they can improve the overall impact and application of user experiences as well as be a platform for more personalized and comfortable architecturally induced urban layouts. When designing for adaptable and responsive architectural environments, because of paying close attention to the details of the changing urban features and conditions, the energy usage as well as the sustainability of the designs are naturally encouraged. And as a result of this feature in design, it may open doors to adaptive spatial layouts that enable for transformations associated with the usage patterns displayed by individuals.

Architects can make use of RL algorithms in coming up with solutions for architectural and urban designs because it acts as a framework that mimics human behaviour in the way that its core concept being learning through consequences, but in digitalized form, easily implemented in Unity’s ML-Agents toolkit (Juliani et al., 2018). An example of this technique is used in Anni Dai’s “Co-Creation: Space Reconfiguration by Architect and Agent Simulation Based Machine Learning” (2022) simulation study, which represented a combined process in architectural design integrating “machine learning based on agent simulation" along with an architect's instinctive knowledge (Dai, 2022). In her research, she chose to experiment with residential spaces where she trained agents in a way where they reshaped the spaces according to how near they were to the walls to be configured. The four elements of the research were "Space", "Agent", "Machine Learning", and "Co-creation", as Dai's approach fused an architect's expertise and experiences with RL as she targeted for fresh collaborations between these study subjects (Dai, 2022). She accentuates that the agents' behaviours produced a sense of uncertainty in the simulation which mimicked the complexities of the real world. Instead of the usual sense of adding the NN model brain to the agent in the task of generating new wall arrangements, her idea was to add the brain to the walls to be reconfigured in this way. Basically, she used her instincts and understanding as an architect to make choices in regard to whether the RL design suggestions that the agents' gave were architecturally applicable. She established that while there are restrictions that offered an inadequate understanding of outcomes, the agents' movements illuminated that their common usage of spaces essentially makes them bigger in size and shrink when they are used less. Also, her findings signified that RL processes in ML-Agents neglect detailed aspects like how long the agents spend in the spaces and other details of design.

Chien-Hua Huang's "Opportunity of Machine Learning in a Material-informed Circular Design Strategy" (2021) study also uses RL in Unity's ML-Agents through the tool "Reform Standard" (Huang, 2021). In the first stage, architects input recycled shapes captured with the program RealityCapture, into the Rhinoceros and Grasshopper applications for analysis. Each shape is arranged for assembly in order of parts in ML Agents, where a boundary (2D or 3D) makes an outline to be able to successfully put it together. This outline, similar to a site perimeter, shapes overall forms while facilitating design control and reducing learning time. The study's technique showed that despite the computational limitations encountered along with RL's true effects remaining incomprehensible, this tool was successful at forming types of complex geometries. These studies provide evidence of adaptive and responsive designs, insinuating that incorporating RL into architectural workflows offers potentials in creating adaptable and user-centric spaces that cater to diverse needs and respond to changing environmental conditions.

With RL's uses in Unity's ML-Agents, agents, technological entities able to learn, can be taught to find a target with motivational curiosity as the catalyst for exploration tasks, where they will go on their navigational journeys to make sense of the digital environment they inhabit (Juliani et al., 2018). Encouraging individuals to question things with a sense of exploration and learning, curiosity's role is deeply rooted in behavioural psychology, because it comes from a motivational sense of purpose and fulfilment seeking (Silvia, 2012). Ultimately, when an individual's explorations are motivated by curiosity, their sense of purpose in fulfilling a task or acquiring information is heightened to lead them in the direction of experiences that align with their objectives (Silvia, 2012).

Abstracted as an explorer of its own city, by French literature writer Charles Baudelarie, the expression "flâneur" is a specific type of approach and outlook to urban exploration that is classified by relaxed explorations together with deep reflections and motivational curiosity (Boutin, 2012). Their voyages throughout the streets of a city are steered by motivated curiosity with the mission of finding out new understandings and perspectives with exploring neighbourhoods that are both recognizable and unrecognizable (Boutin, 2012). From the flâneur's viewpoint, Lynch's ideas from "The Image of the City" act as a basis for understanding the independent perspectives of urban exploration as the flâneur's outlooks are in harmony with the main elements in encouraging the way that they view their surroundings and become motivated to obtain as much knowledge as they can about it (Lynch, 1964). In the situation of an urban explorer like the flâneur, an agent can too suit these depictions as it has the flexibility to be a flâneur explorer of its own environment, a 'cyborg explorer', where just like the flâneur, it curiously perceives its digital urban setting without a pre-defined purpose and lets these senses guide the way that it performs. For architects that are targeting design solutions better fit for individuals, this basis of this framework has the option to be useful just by studying and making sense of the way that agents are taught to behave with RL.

As we discuss technology's multi-layered connection with architectural design, in terms of its cybernetics associations and technology motivated design operations that span conventional as well as digital domains, this thesis' chapters will look into how the continuous occurrence of technology is redefining human-machine relationships in contemporary architecture. It will also examine the way that this relationship reconfigures the concepts of the modern 'cyborg' as a flexible explorer of realms and taking the form of the flâneur as curiosity motivated navigator of its own environment, defined as the cyborg explorer.

The second chapter will review and consider the existing literature that include the central discourse that this framework is built on while mapping the interaction throughout history between technology, cybernetics and architecture. This includes the exploration of the cybernetics of responsive machines and how Pask's conversation theory sits as a foundation for interactive machines, the timeline of architecture machines not only as a theory by Negroponte, Price, and Archigram but also as projects that express the idea.

The third chapter of the thesis is associated with RL as a learning algorithm, into the fusion of fields like architectural and urban design, rooted in cybernetic principles of feedback and embodied through the concept of the cyborg, as a symbol for interactive machines, in offering vast perspectives for creating intelligent and flexible urban environments. This chapter explores the link between AI and cybernetics in their associations with architecture, in discussing the progression of responsive machines through the control mechanisms of feedback loops. In addition, the growing paradigm of the RL algorithm, in ML, is considered as a design tool to coming up with solutions for architectural and urban designs because of its ability to mimic human behaviour using game engine platforms like Unity's ML-Agents. The final part assesses the power that motivational curiosity has as the catalyst for exploration tasks where the cyborg explorer in ML-Agents goes on their navigational journeys to make sense of the digital environment it inhibits, strengthening the link between curiosity and behavioural urban patterns, and views the cyborg as the explorer of a city.

Detailed in Chapter 4 is a simulation study exploring how the RL algorithm within Unity's ML-Agents toolkit, in potentially providing urban design solutions based on human-like motivations that is curiosity. Using Unity's ML-Agents, a game engine and "open-source" toolkit (Juliani et al., 2018), this study endeavours to examine an agent's motivational curiosity in exploring the digital environment of one of Istanbul's urban settings. Using the reward and punishment behavioural learning that comes with RL (Sutton, 1992), the agent in this study will attempt to learn to explore and navigate the urban environment, in search of its target, displaying varying behavioural outcomes and interpretations linked to curiosity, through the trial-and-error mechanism. The fifth chapter will display and interpret the findings and discoveries and concluding this thesis with a conclusion chapter that summarizes the research outcomes and future potentials of such cyborg explorer applications.

The primary aim of this thesis is to delve into the capabilities and constraints of using RL in simulated curiosity-based explorations, through the perspectives of the ‘cyborg explorer’, in bridging the gap between virtual and tangible design environments. In its approach, this work will explore the central theories and concepts tracing the inception of cybernetics’ interaction with architecture, the historical trajectory of responsive architecture, the evolution of architecture machines, and the modern representations of the cyborg while unveiling its purposes as well as methodologies across architectural and urban design contexts. By doing this, it focuses on a series of endeavours at conceptualizing and constructing a cyborg explorer, that is influenced by advancements in technology and a background in cybernetics as well as being motivationally curious in its attitude and outlook as the flâneur, exploring and suggesting behavioural patterns in its environment.

Taking these concepts into consideration, this thesis aims to answer the questions: What is curiosity’s motivational influence on the behaviour of a cyborg explorer? Are there any chances for architects to actively engage and interact with the cyborg explorer throughout the RL cycle? Alternatively, what can an architect gain from such experience in relations to their design solutions?

2. LITERATURE REVIEW

The subject of adaptive and responsive architecture began at a point in time where architects and designers needed newer design explorations, and with the ideas that technology brought, a future with promises of urban environments that are personally customized for users was more easily imagined. Centring on the diversity of evolving technologies and the ideas about the promise of responsive architecture with a background of the effects of cybernetics, this chapter continuously recognizes the capabilities of machine demands thorough grasping the narrative behind interactive ‘architecture machines’, where it includes their journeys through time, the essence of creating such machines, and the influences that they continue to have on architecture presently.

The first section of this chapter delves into the historical timeline of the development of machines with their associations with cybernetics principles, particularly on the influence that Gordon Pask’s conversation theory has on interactive machines, from the 1960s onwards. The second part discusses Nicholas Negroponte’s ideologies and theories (1969) regarding architecture machines as a manifestation of man-machine coexistence, with a feature of some of the Architecture Machine Group (1967) projects, Cedric Price’s Fun Palace (1960), and Archigram’s Plug-In City (1964). A few ‘architecture machines’ are represented in this chapter to illustrate the fundamental principles of these ideologies.

2.1 MACHINE EVOLUTION: A LINK TO CYBERNETICS

To really understand why architecture machines have such a large influence on architectural design today, and why they cybernetically came to be, we must first break down their sequential appearances through history by tracing the link between technology and architecture. The first segment of this literature review picks apart some of the pivotal instances where the areas of architecture and technology intersected. The subsequent part will specifically investigate the applications of cybernetics and its influences on architecture discourse, referencing key theories and ideologies by Gordon Pask, including his Conversation Theory (1976), as well as his projects “Musicolour” (1958) and “Colloquy of Mobiles” (1968).

2.1.1 Origins of Cybernetics

In “Cybernetics: Or Control and Communication in the Animal and the Machine” (1948), Norbert Wiener first introduced cybernetics, where he discussed the human being as a vital part of a system (Wiener, 1948). Wiener’s “The Human Use of Human Beings” (1950) viewed the field of cybernetics as a viewpoint on technology itself while presenting the idea of ‘feedback’ in 1942 (Wiener, 1950). He analysed that cybernetics offered a perspective and approach towards understanding interactions between people and machines, alongside communication, systems, and control of such systems.

The second-order cybernetics proposes that a machine has to identify its environment to be able to adapt its actions with its objectives (Von Foerster, 1995). Glanville implies that Margaret Mead was actually the one who spurred this different approach of cybernetics in 1967 during the First Annual Symposium of the American Society for Cybernetics (Glanville, 2002). Heinz Von Foerster discusses the second-order cybernetics as a realm that recognizes the link between ‘the entity observing’ and the ‘entity being observed’, and that they are of equal value in the system’s function (Von Foerster, 1995).

2.1.2 Pask’s Conversation Theory and Interactive Machines

Gordon Pask’s conversation theory had the motivations of setting up and reproducing interactions amongst different bodies, whether they be humans or machines, with the goal of replicating the natural rhythm of conversation between users (Pask, 1975). For this purpose, it introduced a “strict conversation” model based on a set of guidelines founded within what is categorized as a “conversational language”, nurturing a partnership where both can understand and learn from each other (Pask, 1976).

Pask collaborated with Robin McKinnon-Wood in the early 1950s to create the Musicolour (1953-1958), among the first he developed, where Pask used coloured lights to convey various sets of information. He outlined its functions in “A Comment, a Case History and a Plan” (1971), detailing a light show machine conversating with a musician by responding with different light patterns as it is interpreting their auditory musical outcomes (Pask, 1971).

Pask proposed that Musicolour actually had a “learning capability” that let it “adjust the relationship between the sounds it heard and the visuals it displayed in real-time” (Pask, 1971). The Musicolour machine included multiple-coloured bulbs that each reacted to different sounds that the musician played. When the musician stuck to a certain pitch or stayed at a monotone level, Musicolour would mimic the idea of "boredom" because its light show interactions would decrease and as a result of this the musician would modify the way that they played their instruments in an attempt at redirecting its attentions and increase its lightshow interactions (Pask, 1971). This type of interaction, between man and machine, is an example of Pask's conversation theory about mutual responsiveness and adapting (Figure 2.1).

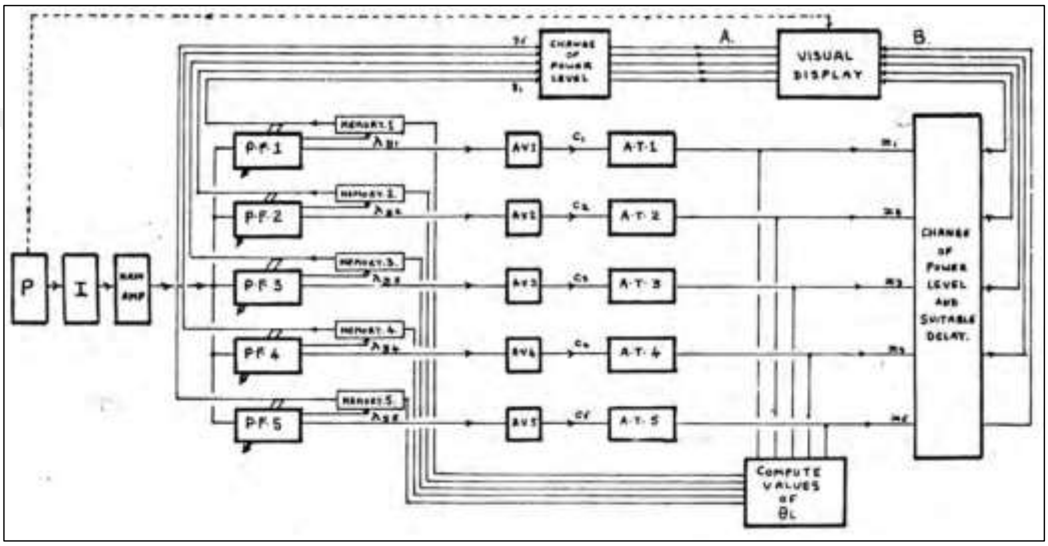

Figure 2.1: Musicolour's Mechanism. (Pask, 1971).

With the intentions of studying conversations spontaneously occurring amongst an independent group of agents, the Colloquy of Mobiles' (1968) machine operated with two 'male' and three 'female' agents capable of communication through visual and audible cues, with both types of agents having two types of motivations (Pask, 1971) (Figure 2.2). The objective for the male agent was to fulfil their motivations (drives) by engaging with the female agent through outputting and receiving light arrays and auditory signals (Figure 2.3). However, in order to achieve this, they required the affiliation of the female agent which was equipped with a reflector positioned in a vertical way to bounce back the light array to the male agent. To meet their goals, the male agents were in competition with each other while being physically linked, which restricted their independent actions. When a male and a female agent with corresponding motivations formed a link, a following sequence of interactions occurred, ultimately fulfilling their motivations. Throughout the interaction exchanges between the agents, they would display different actions like clustering together and avoiding types of collisions, this experiment resulting in unexpected and random behaviours which painted the concepts of self-independent systems that were guided by feedback cycles (Pask, 1971).

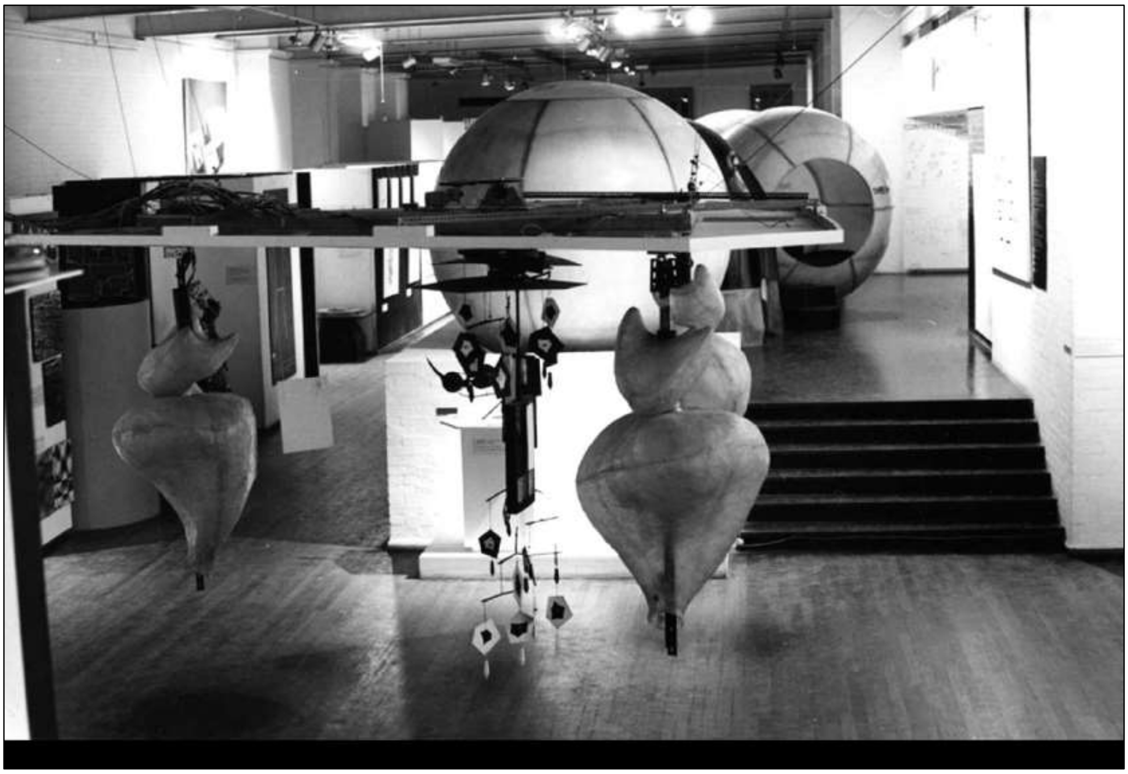

Figure 2.2: Pask's Colloquy of Mobiles Installation. (Rosen, 1968).

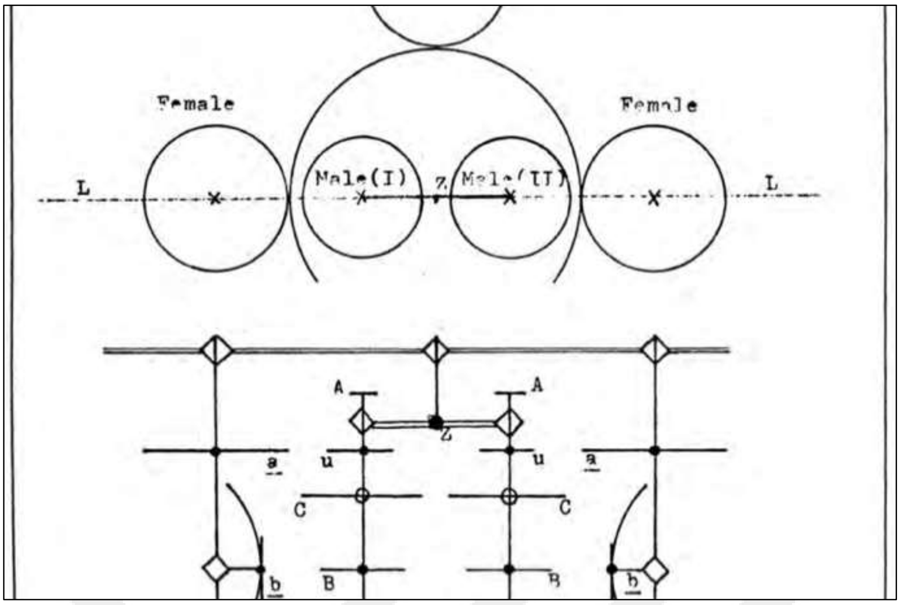

Figure 2.3: The Colloquy of Mobiles' Functional Diagram. (Pask, 1971).

Pangaro and McLeish replicated their own version of the Colloquy of Mobiles in 2018 (Figure 2.4), with the aim being to reform individuals' encounters with a lifestyle surrounded by intelligent machines (Pangaro & McLeish, 2018). They discuss that "the experience of moving among the mobiles of the installation and engaging them via sound, speech, body movements, and facial expressions, hypothetically using enhanced 2018 technology, would offer a rational as well as emotional sense of what it means to live among machines that converse" (Pangaro & McLeish, 2018: 4).

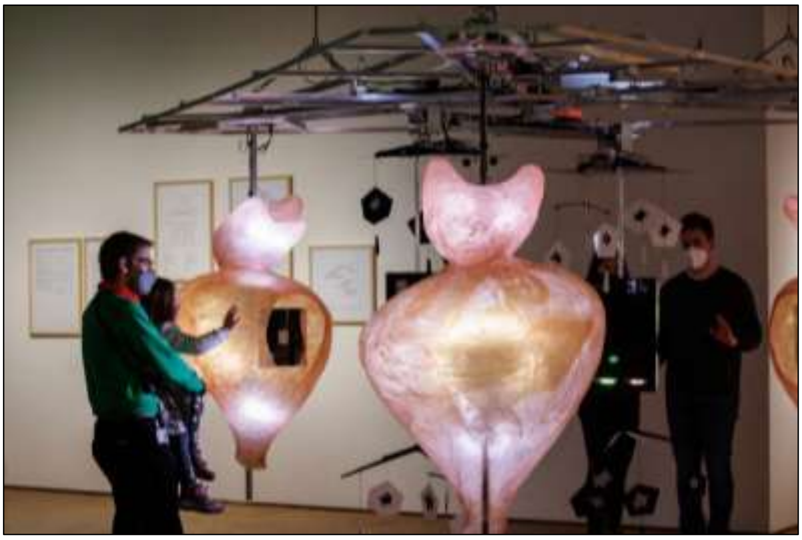

Figure 2.4: The Colloquy of Mobiles at Centre ZKM in 2021. (Rosen, 1969).

2.1.3 The Interplay of Architecture and Technology

With unconventional progressions in construction techniques in architectural frameworks that reformed the basics of architectural practices, the Industrial Revolution greatly affected the world of architectural design. Joseph Paxton's Crystal Palace (1851) was an avant-garde design of its own time, with its innovative use of glass and iron, viewed still as a symbol of progressive and innovative architecture (Berlyn & Fowler, 1851) (Figure 2.5). Paxton conceptualized the Crystal Palace to have the ability to be quickly put together and just as quickly taken apart. It set the base for the ongoing technological discourse and came at a time when architects craved a rise of a new architype (Berlyn & Fowler, 1851).

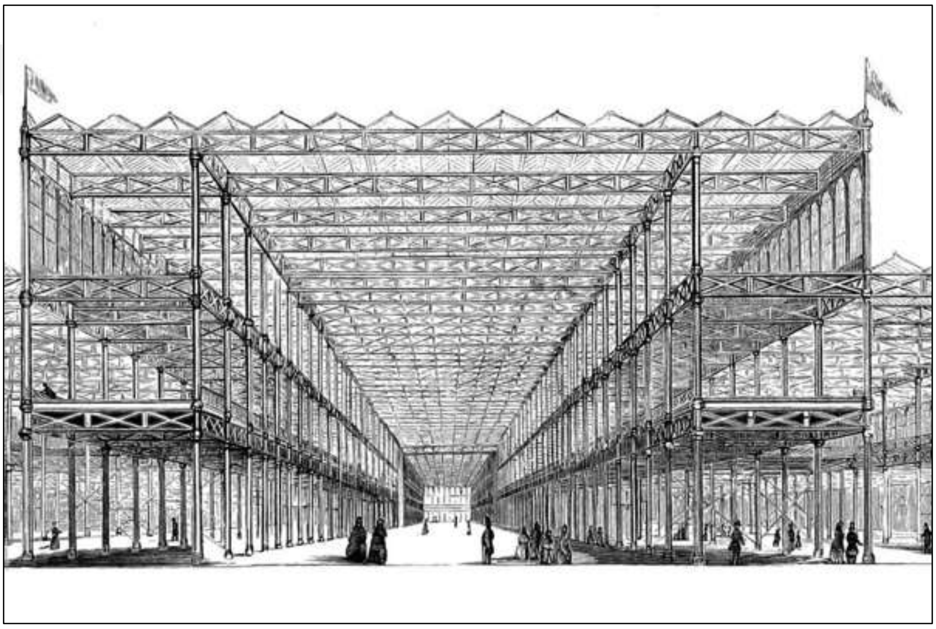

Figure 2.5: The Crystal Palace. (Berlyn & Fowler, 1851).

In 1963, Ivan Sutherland came up with the first graphical interface, as part of his MIT doctoral thesis, as a communication device that recognizes inputs sketched on a computer screen where it then interprets this information (Sutherland, 1963) (Figure 2.6). In "Sketchpad: A Man-Machine Graphical Communication System" (1963), Sutherland portrays the Sketchpad as a digital design construct that allowed for the specification and use of graphical elements on a screen in providing better ways of expressing design ideas (Figure 2.7) (Sutherland, 1963).

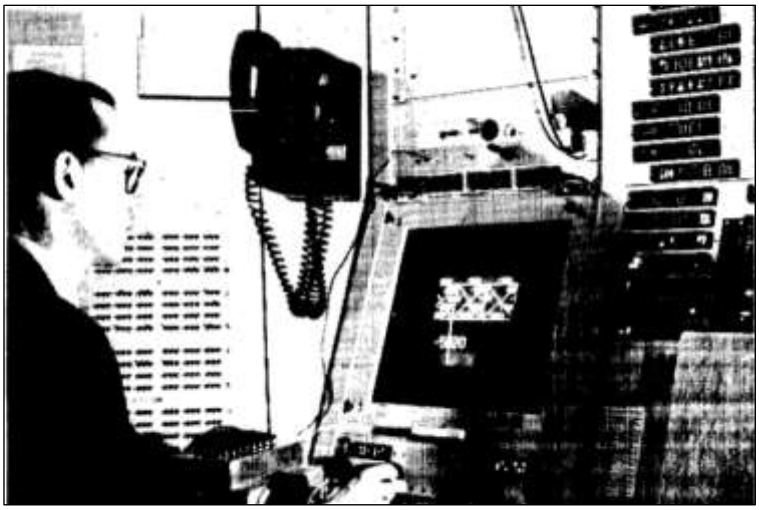

Figure 2.6: Sutherland Using the Sketchpad. (Sutherland, 1963).

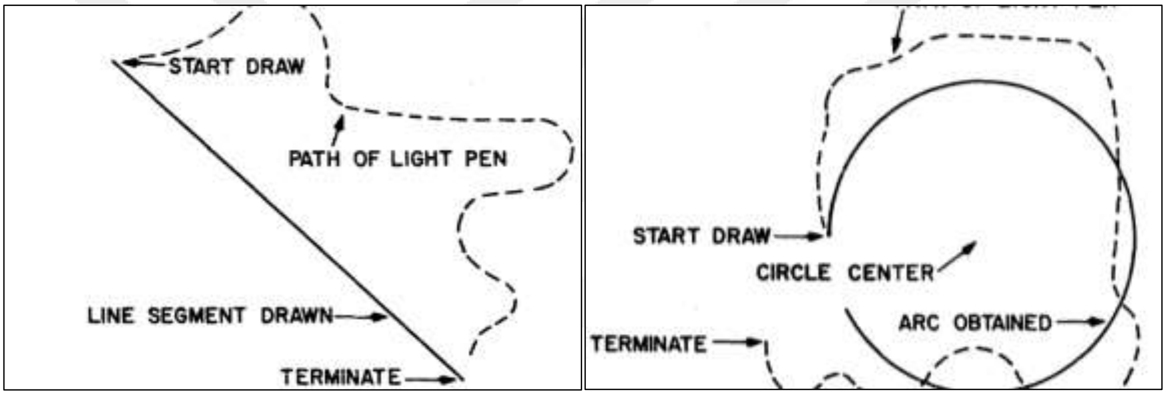

Figure 2.7: Steps for Drawing Straight Lines and Circle Arcs on the Sketchpad. (Sutherland, 1963).

The London Developers Toolkit (2020) game crafted by the design firm You+Pea, shown at Architekturmuseum der TUM’s “The Architecture Machine” exhibition (2020). Pearson portrays, in his “A Machine for Playing In: Exploring the video game as a medium for architectural design” (2020) paper, that the game places the player in the control of designing and promoting a high-end residential development (Pearson, 2020). Through an order of tasks, the game starts with the design of visuals and associated tasks becoming the main priority, resulting in the random generation of the residential tower and sets up a conceptual construct that is perceptible yet clearly defines specific tasks for the player, who in a hypothetical role, functions as an architect working on behalf of a “pair of property developers” (Pearson, 2020: 25).

In the first part of the game, players are tasked with accurately reproducing a "napkin sketch" (Figure 2.8). In the game, the recreation of this action is achieved through a "gesture recognition interface" allowing players to draw exclusively within the game's virtual world (Pearson, 2020). The system then evaluates the player's rendition of a randomly provided sketch and assigns a score where essentially the comparisons of their drawings to the designated path determines their score and so the closer, the higher. The next stage of the game tasks the players with specifying various characteristics that the tower should have, where the player must make sure that there is an equality to avoid making the tower overly costly or insufficient in services to appeal to the clients (Figure 2.9). Pearson states that despite being an architecture game, it serves as an illustration of the role the players within this game play in conceptualizing the creation and portrayal of architecture (Pearson, 2020: 25).

Figure 2.8: London Developers Toolkit "Napkin Sketch" Task. (Pearson, 2020).

Figure 2.9: London Developers Toolkit Tower Characteristics Task (London Developers Toolkit, 2020).

2.2 ARCHITECTURE MACHINES

The conceptual idea of a “machine” can be understood in relation to technology and cybernetics, in the context of architecture, despite its negative understandings that are often times connected to negative understandings, for some linked with robotic or cyborg-like representations. As conceptualized by architects Cedric Price and Nicholas Negroponte, with cybernetician Pask, a ‘machine’ is a construct of an architectural and cybernetic system that has the willingness of continuously ‘learning’ and actively participating with individuals with the intentions of ‘planning’ for future habits and user uses.

Negroponte’s “architecture machine” is an interpretation of a viewing “architecture as a living machine”, that he detailed in “The Architecture Machine: Toward a More Human Environment” (Negroponte, 1970). The ideas related to that of an “architecture machine” were experimented with in the Architecture Machine Group (AMG) (1967) studies that were led by Negroponte himself. The AMG were motivated with creating environments where digital technology interacts seamlessly with architectural elements such as in URBAN5 (1968), SEEK (1970), and the Aspen Movie Map (1978), in persuading the way that people would engage with and experience spaces (Negroponte, 1970). Negroponte proposes that ‘architecture machines' can heighten architectural design and the built environment experiences by observing, interpreting that information, and then modelling the behaviours understood as an attempt at learning about architecture (Negroponte, 1978).

With considerable pressure on man-machine interactions underscoring the role of technology in reshaping the overall architectural experience, this collaboration forms a "dialogue between two intelligent systems (the man and the machine)" that possesses the ability to generate an "evolutionary system" (Negroponte, 1969: 9). In addition, he reviews a "heuristic" method which basically denotes to a "problem-solving" mechanism, using spontaneous schemes to find solutions, rather than just strictly following a set of "predefined" instructions (Negroponte, 1969). Negroponte states that with frequent exposure, a "rote learning" mechanism, a type of routine system, would assume control. Stating that it is a process that uses basic memorization of an event, linked to a specific response (Negroponte, 1969). Nonetheless, in contrast to its heuristic equivalent, it commonly initiates with a limited range of situations it can quickly address (Negroponte, 1969).

In the article, Negroponte talks about the Skinnerian approach, known as operant conditioning (1938), which theorizes that learning a behaviour is influenced by its consequences either as positive or negative reinforcements (Skinner, 1938). With a Skinnerian advance, the positive reinforcement makes selections based on actions that are favoured by the "teacher" (in this case it could be the designer, or any entity overpowering the "student") (Skinner, 1938). These theories are some of the foundations that influenced Negroponte's ideologies in association with "architecture machines" due to the potentials of learning through reinforcement and interactive behaviours.

In 1970, Negroponte wrote his first book "The Architecture Machine: Toward a More Human Environment" in which he unfolds a vision involving computational tools not only reshaping the design process but becoming indispensable elements in shaping our interactions with built environments (Negroponte, 1970). He collaborated on this book with Leon Groisser to break down the narrative around architects having the limitless chances to engage with constantly changing conditions of digital methods and put these thoughts and concepts into experiments they conducted for the AMG (Negroponte, 1970). Negroponte suggested the idea that architecture machines should create a distinctive dialogue characterized by a unique type of interaction between the architect (man) and the machine. He articulates that "such a machine could […] reinforce the dialogue by using the predictive model to respond to you in a manner that is in rhythm with your personal behaviour and conversational idiosyncrasies" (Negroponte, 1970: 13).

In the following chapters of the book, he states that engaging with heuristic mechanisms could assist architecture in acquiring the ability "to learn, and more importantly, the desire to learn", his vision being that through such techniques, machines could undergo a learning and maturing process affiliated to that observed in humans (Negroponte, 1970). In this context, he further states his vision for architecture machines, portraying that they should not merely partake in "dialogue" with the environment but must also "receive direct sensory information from the real world. It must see, hear, and read, and it must take walks in the garden" (Negroponte, 1970: 27).

Negroponte later authored his second book, "Soft Architecture Machines", published in 1975, in which he discussed the predominant objective of rendering the built environment "responsive" to individuals through the construct of machine intelligence, imagining an intelligent platform with a broader and more comprehensive basis of technology into architectural discourse (Negroponte, 1975). Proposing the ideas of having a "dialogue" with architecture machines, he suggested that architects should use this unique "dialogue" to understand the machine, and the opposite (Negroponte, 1975). For the introductory chapter "Introduction to Aspects of Machine Intelligence" co-authored by Gordon Pask for "Soft Architecture Machines" (1975), where Pask advocated for the capacity to stimulate man-machine interactions within architectural contexts by discussing and outlining some aspects of his conversation theory (Pask, 1975). He reflects this process with the dynamic adaptation of not only the machine itself but also of humans as well as the built environment in which they interact. Similar to how the feedback of creating a model influences the model itself, the process of problem-solving has the potential to refine and enhance the representation of the problem. Negroponte drew inspiration from Stanford Anderson, an MIT professor, adopting the concept of "problem-worrying" (1966), that Anderson described as an actively engaged approach to the problem situation, emphasizing continuous involvement and consideration (Anderson, 2002).

Negroponte dissected the processes of the URBAN5, SEEK, and the Aspen Movie Map projects, URBAN5, an early rendition of URBAN2, was an experiment on a design program system that could form "dialogues" and conversations with individuals (Negroponte, 1970).

Using creative miniature world, project SEEK elaborated filling a miniature environment with cubes that were constantly repositioned by a robot arm and prejudiced by the presence of gerbils. In the Aspen Movie Map study, the AMG's intentions were to develop a digital simulation of Aspen through video footage, to let individuals navigate and explore the town in a virtual mode without physically being there themselves.

2.2.1 The Architecture Machine Group

Part of the AMG's endeavours was the focus was of "micro-world", a framework that was designed to redefine the extent of investigations (Negroponte, 1970). The "micro-world" aimed to concentrate on certain issues, deliberately retracting from other considerations to create a focused and controlled environment for exploration. During their initial years from 1967 to 1974, the AMG constructed models drawn from "micro-world" concepts involving users and their environments (Negroponte, 1970).

A particular category of microworld employed by the AMG was the "blocks-world", which essentially involved focused experiments with a number of blocks (Roberts, 1963). These block-worlds functioned as confined environments, strategically constraining the scope of design issues for experimentation. One of the first AMG projects involved the development of URBAN2 which later became URBAN5 (Negroponte, 1970). The introductory rendition of the system, URBAN2, originated as a class project during Negroponte and Groisser's 1967 course "Special Problems in Computer-Aided Urban Design" (Negroponte & Groisser, 1967).

In "The Architecture Machine" (1970), Negroponte mentions that, in URBAN5, instead of assuming the role of the designer, the system functioned as an "urban design clerk" with the capability to "monitor design procedures" (Negroponte, 1970). Disdain to its alterations of graphic input, the primary stress was on "natural language dialogue" where this feature aimed to provide clarity and commentary during the design process, on the choices taken by the user. URBAN5 started conversations with the user by subsequently learning from the user's interactions, and eventually appealing in a dialogue using two kinds of languages: an English language using a "typewriter", and a graphic language inputted through a "cathode-ray tube" (Negroponte, 1970).

The program unified a graphical design user interface where individuals could manipulate cubes measuring 10 by 10 feet (3 by 3 meters) (Negroponte, 1970). Additionally, the systems facilitated interactive communication through a “question-and-answer dialogue” between the “man and machine” (Steenson, 2014). The cube was opted as the fundamental design unit by Negroponte and Groisser, due to its inherent lack of “architectural impositions and many research conveniences” (Negroponte, 1970). With the function of a light pen, the user could draw as well as select squares (representing a 10’ by 10’ block (3 by 3 meters)) on a cathode ray tube screen, where the user assigned “modes” (contexts and features) through a number of 32 buttons (Figure 2.10) (Negroponte, 1970). In the “TOPO” mode, the user has the capability to adjust the topography of a plan while in the “DRAW” mode, for example, the user is able to control both the forms and the viewing mode (Negroponte, 1970).

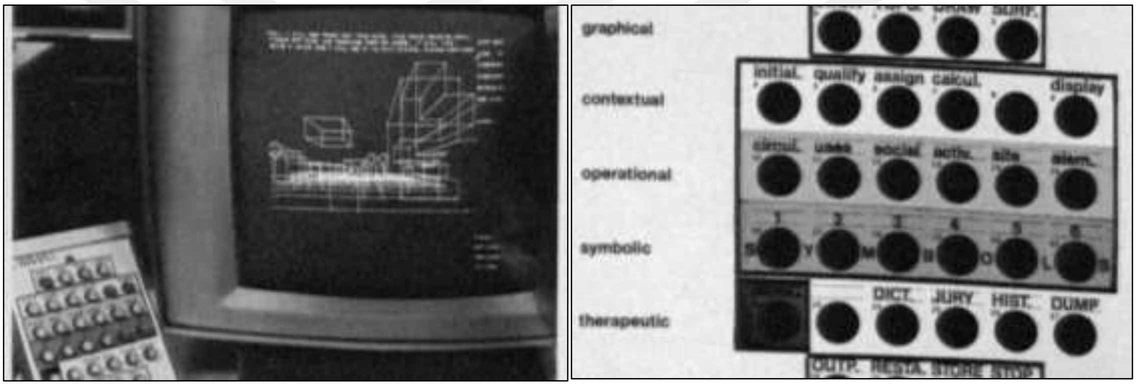

Figure 2.10: URBAN5’s Modes. (Negroponte, 1970).

When the user chose a cube and mode, the program would prompt a question that was pre-programmed from a set of around 500 options, with every mode having its own designated questions (Negroponte, 1970). When two cubes are drawn adjacent to each other, they combine to form a unified volume (Figure 2.11), as the system categorizes them into one of four attributes being “solid” (establishing a prominent boundary for a specific activity), “partition” (suggesting a separation for mutual utilization), “transparent” or “absent”, these volumes can be designated with the feature of “access” (Negroponte, 1970: 75). As cubes were added or removed by the user, the system automatically assigned “implicit” features, as URBAN5 managed features “explicitly or implicitly”, where every cube came with inherent qualities such as “sunlight, outdoor access, visual privacy, acoustical privacy, usability, direct access, climate control, natural light, flexibility, structural feasibility” (Negroponte, 1970: 81). In the concluding phase of the interaction between the user and the machine, a dialogue unfolded between the two, where if there arose an inconsistency between the specified context and the resulting form, the machine alerted the user which had the liberty to decide the coming course of action; whether to overlook the inconsistency, postpone, or modify the form.

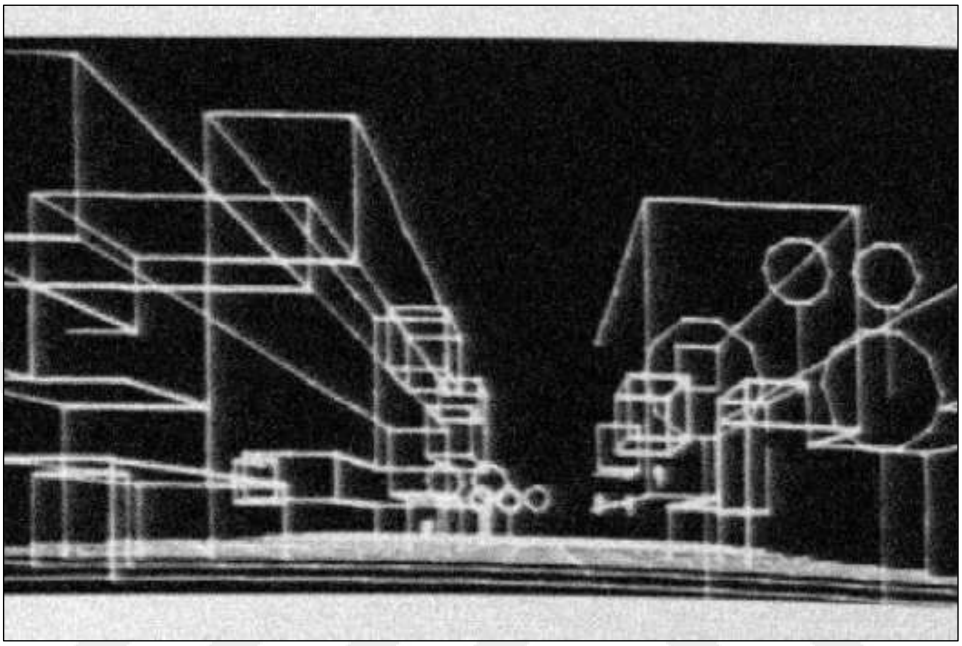

Figure 2.11: Cubes/Volumes Inputted Into URBAN5's Program via Light Pen. (Negroponte, 1970).

The aim for URBAN5, as Negroponte and Groisser envisioned, was to acquire knowledge and adjust it through interactive dialogue with users, where users select modes, envisioning the system showcasing its "intelligence" by being attentive or inattentive to the context at hand (Negroponte, 1970). In URBAN5, preference was given to the text information, emphasizing it over graphic data (Steenson, 2014).

URBAN5 shared similarities with Joseph Weizenbaum's 1966 ELIZA psychotherapist computer program created at MIT to mimic the conversational style of a Rogerian therapist as a participant, inputting queries from a computer terminal, assumed the position of a patient engaging in an interview their psychiatrist, where the program identified and broke down phrases and changed the responses in order to create the impression of a coherent dialogue and sentimental conversations (Weizenbaum, 1966). Negroponte stated that although the project's potentials, they fell short because it could not handle managing multi-tasking across domains that were complexly intricate (Negroponte, 1970).

Another project associated with the theme "blocks-world" was known as SEEK (Negroponte, 1970). In 1970, Nicholas Negroponte worked in collaboration with the AMG at MIT on SEEK as part of the "Software" exposition that Jack Burnham curated for the Jewish Museum in New York. Within this installation, the AMG aimed to harmonize abstract computer-generated visualization models with tangible physical realities while bridging the divide between virtual representations and the real-world conditions of the physical world, the initiative entailed placing gerbils inside a Plexiglas enclosure alongside "zinc-plated" boxes (Figure 2.12) (Negroponte, 1970). The gerbils, through continuous movements, altered the positions of the cubes within the glass enclosure. Guided by programmed instructions from the designers, a robotic arm was employed to manipulate these cubes into a specific sequence, and ultimately if the sequence of the cubes was altered by the gerbils, the robotic arm would reconstruct the cube sequence based on what the programmers thought would align with the aims of the gerbils. Basically, as the gerbils moved and altered the positions of the cubes, the program "observed" their behaviours and then "learned" the preferences of the user (in this case the gerbils) based on their movements of the cubes (Negroponte, 1970).

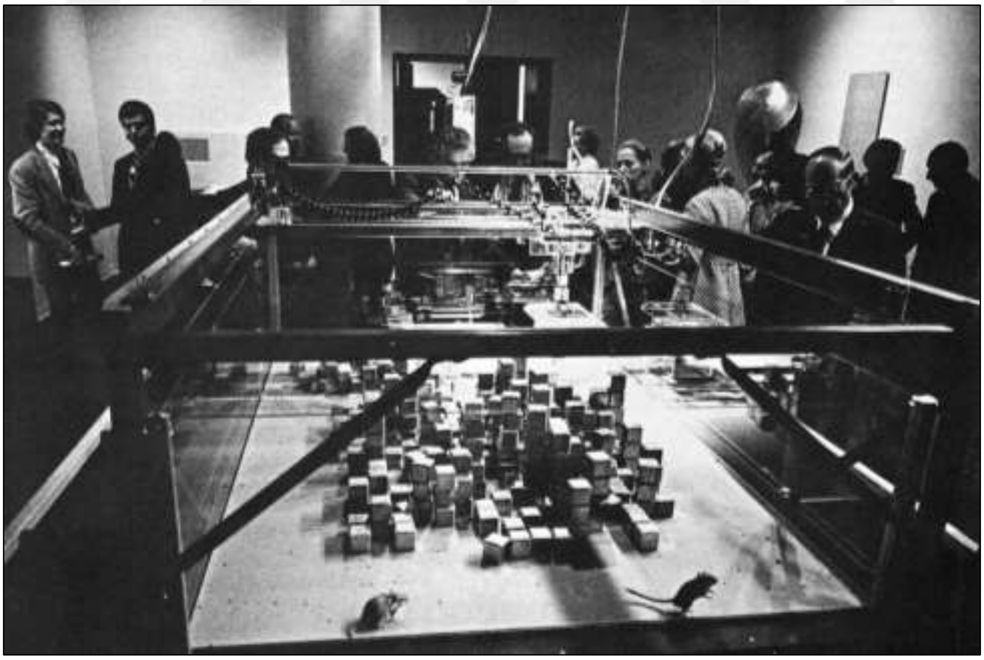

Figure 2.12: SEEK at the Jewish Museum's "Software" Exhibition in 1970. (Katz, 1970).

The computerized control system was created to perceive the surroundings and uphold manner within an environment where gerbils served as experimental users as well as parameters for the computer system (Negroponte, 1970). SEEK explored the interaction of two intelligent agents (man and machine) within a self-contained cycle of dialogue feedback. In the “Software” exhibition catalogue “Information Technology: Its New Meaning for Art” (1970), Katz stated employing a mechanism “that senses the physical environment, affects that environment, and in turn attempts to handle local unexpected events within environment” (Katz, 1970). The system executed six operations: “Generate, Degenerate, Fix It, Straighten, Find, and Error Detect”, as explained by the AMG in “Computer Aids to Participatory Architecture” (Groisser & Negroponte, 1973). These functions were employed to haphazardly arrange, reshape, correct, straighten, locate, and identify errors within the cube environment, where the robotic arm was utilized to stack and shift the cubes into their designated positions (Figure 2.13).

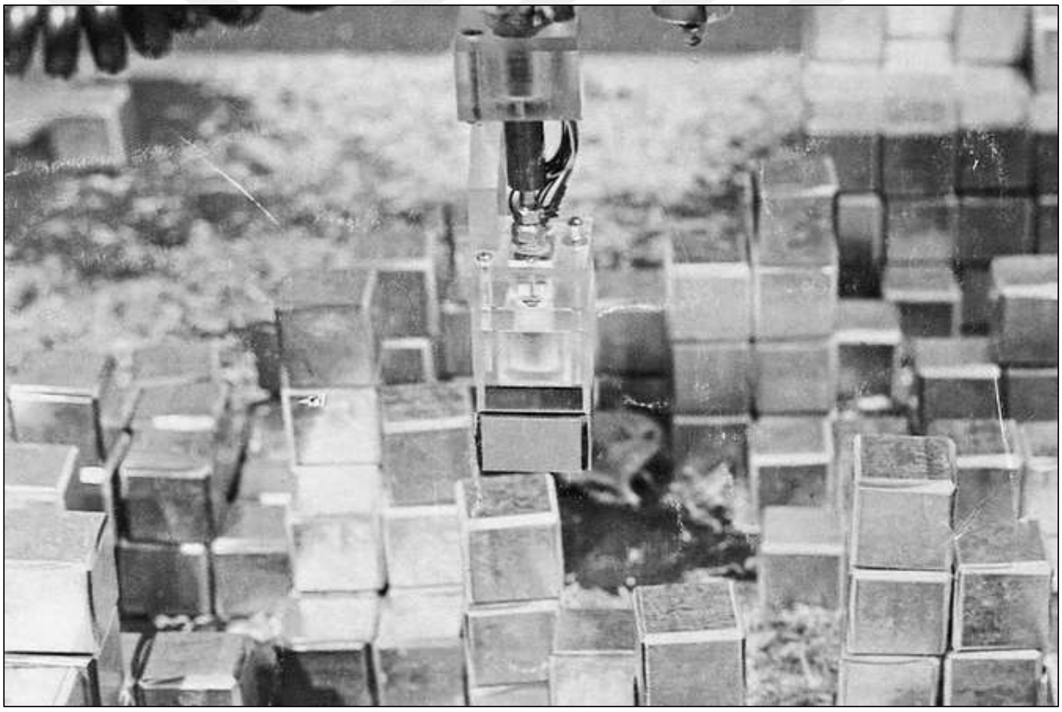

Figure 2.13: SEEK’s Robotic Arm Arranging Cubes Surrounded by the Gerbils. (Katz, 1970).

Some of the failures of SEEK, noted in the “The Architecture Machine”, were that the robotic arm influenced the gerbils psychologically more than it do so physically, altering their environment and that the gerbils demonstrated their intelligence by devising a strategy to overcome the machine, as they started climbing onto the robotic arm (Negroponte, 1970).

The Aspen Movie Map, which the AMG developed between 1978 and 1980 with the intent of creating a cohesive digital simulation of Aspen’s neighbourhoods to allow for virtual user explorations of its streets (Lippman, 1980). Ultimately, by using the video materials they captured by navigating Aspen’s streets with a make-shift camera linked to a car, the main concept was using this visual footage through a digital system that let individuals experience the town through exploring different routes and perspectives, with its primary goal being to create a “dynamic replacement for a topographic paper map” (Naimark, 1982).

Balanced over a truck was a “gyroscopic stabilizer” holding 4 types of film cameras (16mm stop-frame), an optical sensor affixed to the hub of the wheel of a bicycle that trailed after the vehicle (Naimark, 1982). The AMG opted for Aspen as the perfect testing environment because of its functional street layout and manageable size with standard architectural features. Individuals had the capability to digitally manoeuvre along the streets using either a touch screen interface or a joystick (Lippman, 1980). A correlation was established between the video layout and the 2D street map within the database (Figure 2.14), as a connection that enabled the users to freely select any route throughout the city, with a few limitations being staying within the middle of the street, advancing by 10 feet additions (3 meters), and viewing the street from either of the 4 perpendicular angles (Anable, 2011) (Figure 2.15).

Figure 2.14: Aspen’s 2D Dynamic Street Map. (Allen, 1978).

Figure 2.15: Aspen Movie Map (1978) with Joystick and Touch Sensitive Monitor. (Aspen Movie Map, 1981).

The distorting of margins between what is considered to be ‘virtual’ and ‘real’ is what made this project so absorbing, leading to a dissolution of fixed realms (Halpern, 2015). In the documented videos of the Aspen Movie Map at the MIT archives, it is seen that as the map gradually reveals itself, the video surrounding the user creates a sense of historical context, illustrating the transition to an interconnected and responsive system, from an autonomous one. To add, the individual in this context experiences a dual dynamic where they are empowered with a feeling of “authority” over the environment as well as becoming unified into the interconnected system (Halpern, 2015).

2.2.2 Cedric Price and Archigram’s Machines

In contrast to Negroponte’s concept of “architecture as a living machine” specifically concentrating on the explicit integration of digital technologies into the built environment, Cedric Price’s “architecture machine” takes on a broader user-centric approach to architectural design (Price 2003). In 1968, at the MIT “Planning for Diversity and Choice: Possible Futures and their Relations to the Man-Controlled Environment” conference, according to Weiss, Price conveyed his vision for responsive design, to which he used the phrase “calculated uncertainty” (Weiss, 1968). His perceptions and visions for an “architecture machine” were motivated by a deep reconfiguration of architectural design where it is able to constantly change depending on societal, technological, as well as cultural circumstances where he stated that he was “convinced that the valid social life of the activity that one is asked to shelter and encourage is the governing factor of whatever is produced; and that need not always be a building” (Weiss, 1968).

Price’s highpoints his farsightedness in “The Square Book” (2003), appearing as entities with their competence to adjust, transform, and accommodate countless functions and conditions, where his philosophy of architecture machines essentially accentuates a departure from traditional fixed architectural paradigms, and spotlights more fluidity and interactivity within architectural spaces (Price, 2003). Price anticipated the design of the Fun Palace (1961) with Joan Littlewood and system consultant Gordon Pask, a visionary cultural centre aimed to be a dynamic, flexible platform for arts, entertainments, and learning (Mathews, 2003).

Archigram’s work, with similar aspirations as Price, was branded by space-age imagery, with an addition of science fiction and pop culture elements, their projects not intended as real proposals but instead as speculative visions that challenged the architecture of their time and pushed the boundaries of architectural discourse (Selwyn, 2008). They worked on different projects and displayed them in the form of an annual magazine collective. For instance in 1964 they released their fourth issue of “Zoom”, which fixated on a “space comics” theme with the conception of the “megastructure” and in it they starred Plug-in City, Walking City, and Computer City projects (Selwyn, 2008).

Peter Cook’s Plug-In City (1964) was an idea for a segmental “megastructure”, composed of temporary manufactured units that could be “plugged into” a structure where they can be easily removed (Selwyn, 2008). Also presented in their magazine was Ron Herron’s Walking City (1964), which he also designed to reflect a type of “megastructure” that was hypothesized to have legs and be able to travel places, while being able to adjust to diverse surroundings and adapt to its user desires (Selwyn, 2008).

Price honed his concept of an adaptable architecture machine, capable of being modified to suit user’s requirements, drawing inspiration and refining it through Joan Littlewood’s concept of a lively and user engaging theatre that encouraged interaction and spontaneity (Mathews, 2003). Price decided that the Fun Palace should not adhere to a fixed form and layout, as Littlewood had a vision for the project where it lacked a specific program and event regime. Instead of designing a conventional building to accommodate the program, Price conceptualized a simple "framework" that could serve as a foundation for the extension of events (Mathews, 2003) (Figure 2.16).

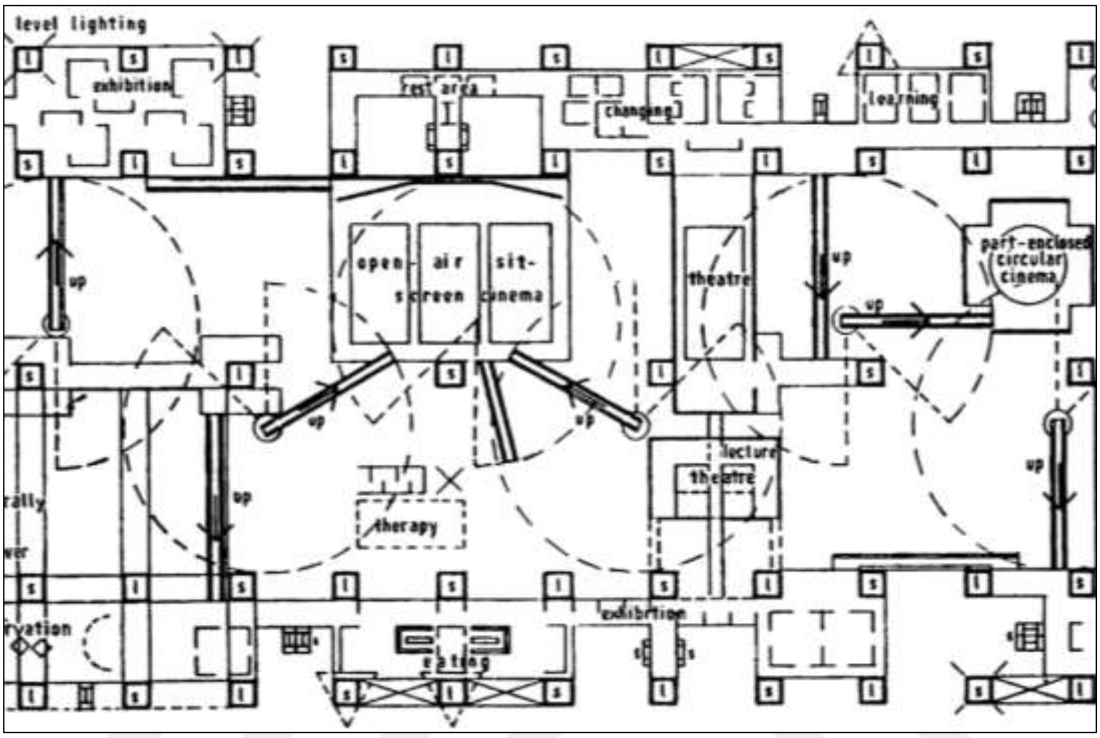

Figure 2.16: The Fun Palace Floor Plan. (Architecture (CCA), 1960).

In the "The Fun Palace as Virtual Architecture" (2006) article, Stanley Mathews stated that the concept of the Fun Palace was envisioned as a dynamic centre affiliated to a "university of the streets", posing assorted opportunities such as language learning, film watching and production, virtual exploration, cooking classes, computer literacy training, and many more, insinuating that the event manifestations were limitless (Mathews, 2006). Price suggested a framework of steel grids equipped with cranes where modular elements could be positioned, as well as designed an external structure suggestive of a "shipyard" and equipped it with movable "gantry cranes" to shift the various components around, where he hoped that the user could design the spaces with the use of cranes (Mathews, 2003).

Pask had a great impact on the design and functionality of the overall design system as head of the cybernetics committee for the Fun Palace, with he focused on developing systems that facilitated user participation, interaction, and dynamic adaptation in its space (Mathews, 2006). In the manuscript "Proposals for a Cybernetic Theatre" (1964), Pask characterized "theatre" as the transmission of data (meta-data), accenting the position of user "feedback" as well as interactions between the people acting and the spectators (Figure 2.17) (Pask, 1964).

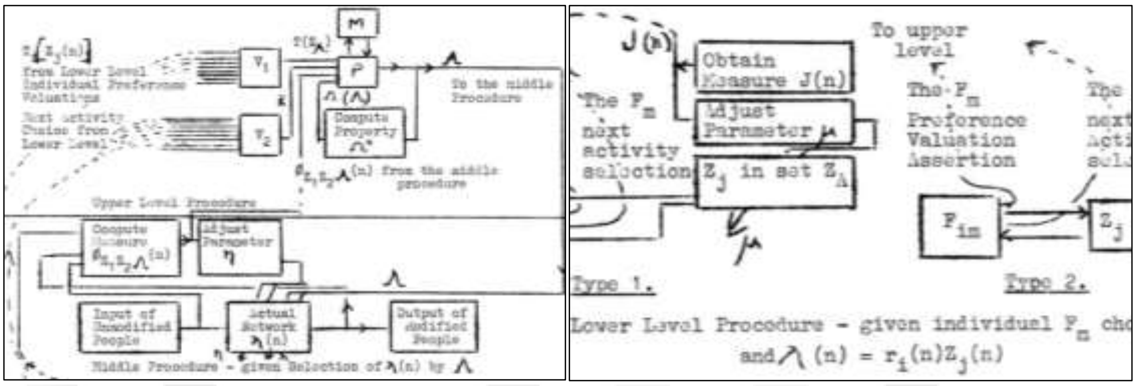

Figure 2.17: Gordon Pask's Cybernetic Diagram of the Fun Palace. (Architecture (CCA), 1960).

The Fun Palace also incorporated principles from game theory, an ideology conceived in the 1920s by mathematician John Von Neumann and expanded on in the 1950s (Halmos, 1973). Game theory, unlike cybernetics which primarily deals with "short-term" adjustments, focuses on devising "long-term strategies" and adjustments for complicated programs (Mathews, 2006). Essentially, game theory was used to manage "long-term" activity approaches in the Fun Palace, whereas cybernetics directed the immediate actions of daily events (Mathews, 2006). The project's program was, in a theoretical sense, to possess the ability to foresee unforeseeable events, as it would use "probability" to adapt to evolving occurrences and activities. In the practical sense, the program would employ "electronic sensors and response terminals" in order to collect and categorize a "prioritized value to raw data" concerning the user's event preferences. This data would then be collected by an "IBM 360-30 computer" to identify predominant user patterns, consequently ordering the guidelines for adjusting both the "spaces and activities" (Mathews, 2006).

In Lobsinger's "Cybernetic Theory and the Architecture of Performance" (2000), she details that, theoretically, the main part of the Fun Palace would predominantly be sheltered by a type of membrane roofing, hanging over from a network of cables, and featured adjustable sky blinds positioned overhead (Lobsinger, 2000). The overhead cranes positioned between the base level and the ceiling could lift and position building elements into different locations, enabling the assembly process (Figure 2.18). As necessary, adjustable decks could be installed in the "side aisles" to serve additional purposes while the interiors of the Fun

Palace consisted of “standard modular modules” made of aluminium and plastic, which were inflatable and had the flexibility to be relocated throughout the entirety of the framework (Lobsinger, 2000) (Figure 2.19).

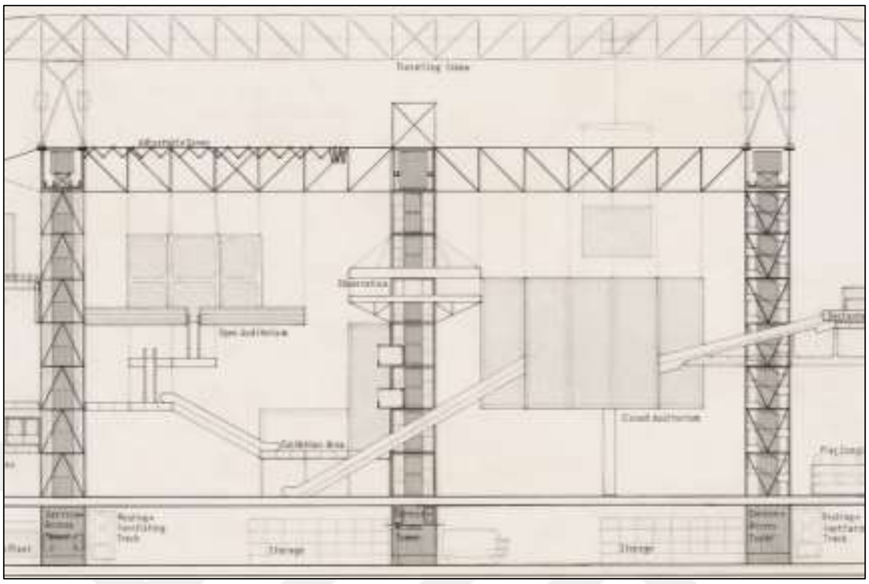

Figure 2.18: A Section From the Fun Palace. (Architecture (CCA), 1960).

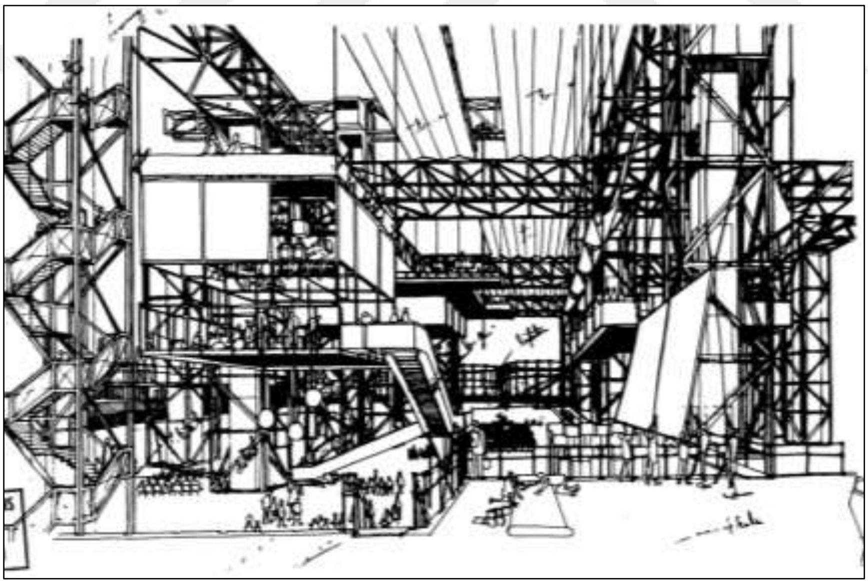

Figure 2.19: The Fun Palace Interiors. (Architecture (CCA), 1960).

Joan Littlewood described in “A Laboratory of Fun” that “within the complex, the public moves about, above the largely open ground-level decks, on ramps, moving walkways, catwalks and radial escalators” (Littlewood, 1968: 132). The Fun Palace did not obey to conventional architectural norms, instead it was designed as a “giant learning machine” intended to spatially fuse with technology all while expediating for users to get accustomed to abstract encounters in a physical and mental way (Lobsinger, 2000: 120). A site for the Fun Palace was essentially arranged by Price and Littlewood, at Mill Meads and situated alongside the Lea River in East London, however the project was shortly paused and so the Fun Palace remained unfinished and was never realized (Mathews, 2006).

Archigram's Plug-In City was a revelation of its own time, first introduced in Archigram's "Zoom" magazine in 1964 (Selwyn, 2008). It was theorized and designed primarily by Peter Cook (1963-1966), with participations from Dennis Crompton and Warren Chalk (Selwyn, 2008). Cook's foresight was effectively abstracting a flexible "megastructure", with the adaptableness of reconfiguring itself to suit the preferences and habits of its users, as a goal of countering the collective urbanization and population density in cities, plus surveying the prospective of segmental construction and mass-produced parts (Figure 2.20) (Selwyn, 2008).

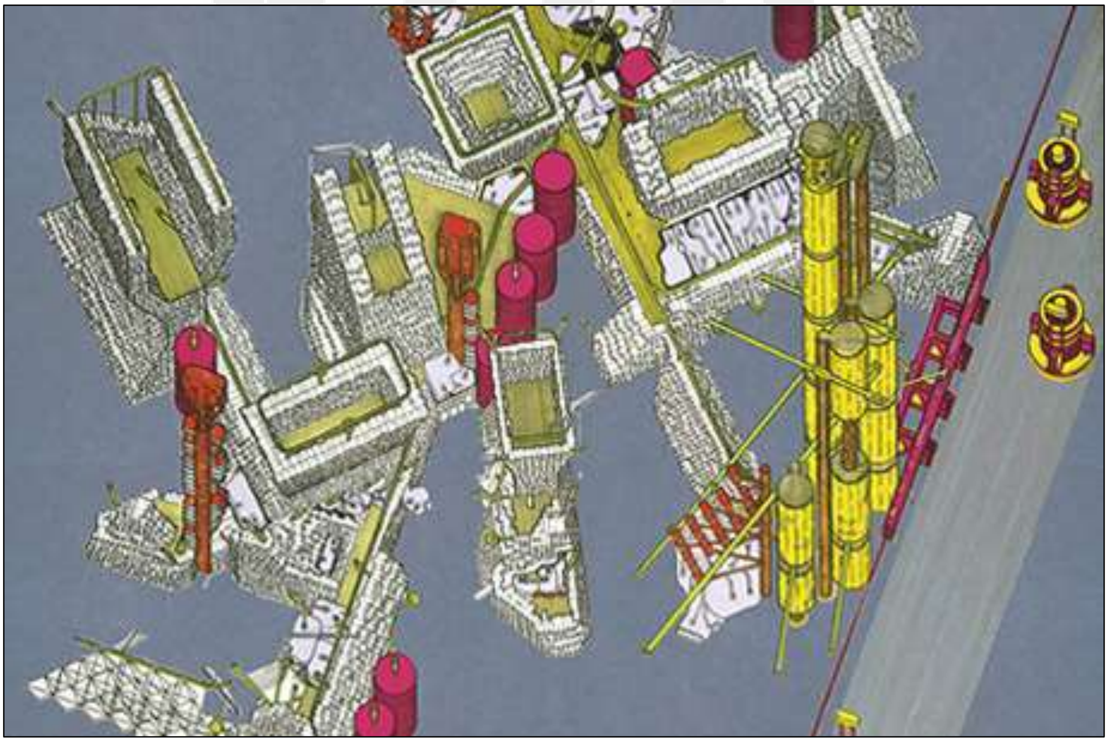

Figure 2.20: Plug-In City Axonometric View. (Archigram, 1964).

Its design was portrayed with the vision of a central 'skeleton' which was encompassed of interconnected outlets that housed required amenities and passage routes (Selwyn, 2008). In premature blueprints of Plug-In City, it resided of a system of cranes situated atop a "railway" at the megastructure's crown, which certified for the movements of modular units and flexible recognitions (Selwyn, 2008: 7). The modular units within the structure would have various lifespans, with the main frame and physical structure expected to endure for 40 years (Figure 2.21) (Selwyn, 2008).

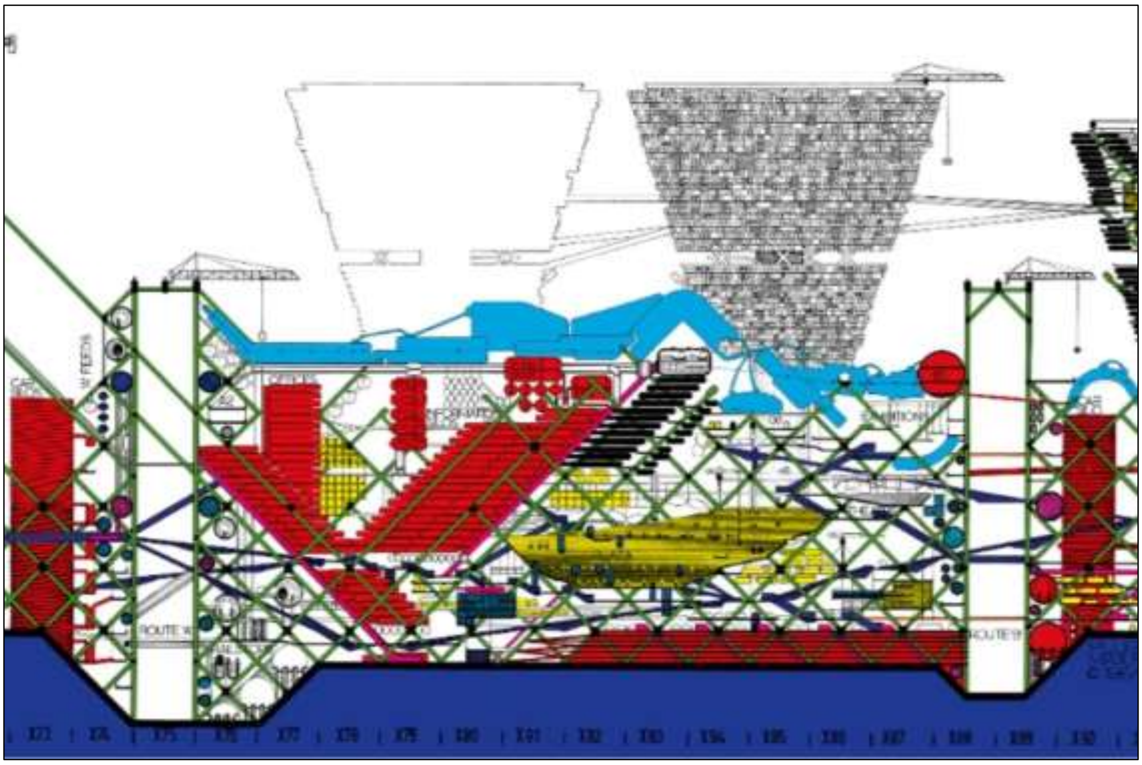

Figure 2.21: Plug-In City: "Maximum Pressure Area". (Archigram, 1964).

3. REDEFINING HUMAN-MACHINE INTERACTIONS IN DESIGN

The relationship between AI and architectural as well as urban design brings about an array of possibilities for creating adaptive and smart cities when invested in cybernetic design principles through the concept of the cyborg. Enhancements in user occupancy and experience, more personalized settings that suit specific needs, as well as the dynamic possibility of adaptability in responding to these changes are some of the potentials offered by these technologies. Architects have the chance to pivot architectural design standards towards more responsive and adaptive solutions with the integration of AI-based systems that implement ML, and in that RL algorithms.

In the first section of this chapter, we will look into how AI constructs, in association to cybernetics are applied in architectural design, refining the development of responsive environments that adapt to changing conditions, with its focus on feedback loops and control mechanisms and viewing urban spaces and layouts as complex systems. Moreover, the idea of the 'cyborg' as a symbol for interactive machines expands upon the concept to imagine urban layouts enhanced with technology through the use of intelligent entities (agents) that can interact with their users and existing surroundings.

The second section will get into how ML, and in that, the use of RL can be used as a design algorithm to enhance built environment performance and further explore new design solutions. Additionally, this section further discusses the use of RL, via ML-Agents in the Unity game engine, allowing architects and designers to implement intelligent and adaptive environments that redefine the connection between humans and the urban setting, in addition to recent ML-Agent research applications in architecture.

In the last section, we will discuss the multi-layered dynamics of curiosity as an influential motivational force as well as in its effects in relating to fulfilment as a goal. This exploration will be juxtaposed with the perspective of agents in ML-Agents to draw parallels to the idea of the flâneur, in addition to how curiosity can have a behavioural effect on them.

3.1 ARTIFICIAL INTELLIGENCE IN ARCHITECTURAL CYBERNETICS

In Haugeland's book "Artificial Intelligence: The Very Idea", he positions AI as a framework that has developed from its historical desires to be intelligent (Haugeland, 1989).

AI oversees the application of predictive models that foresee future conditions based on the past and present data it collects (McCarthy, 2007), where, for example, it predicts usage trends along with energy requirements which lets architects assume and design for changes in user usages and other conditions.

3.1.1 Architectural Cybernetics

Cybernetics, discussed in Chapter 2, being the interdisciplinary study of feedback loops and control mechanisms in systems explores how systems, whether they be biological or technical, self-regulate and adapt to achieve certain agendas (Foerster, 1995). In the context of architecture, cybernetics provides an established framework for developing interactive architectural systems where through the use of sensors and actuators, architects can use those in their designs that aid in dynamically adapting to ever-changing circumstances in real-time instead of using manual changes. In addition to that, cybernetics allows for the design of systems that can essentially learn from data and so adapt and evolve over time such as in ML algorithms, and more specifically RL. This ongoing learning cycle lets systems continuously adapt to changes in real-time as well as make predictions based on its data knowledge and past experiences in making operational decisions and becoming better at responding to system feedback. Cybernetics' interactive feedback loop encourages a more co-operative connection among users in architectural spaces as it lets individuals to actively reshape their surroundings, considering architectural entities as interconnected elements within a larger socio-technical system rather than just as separate pieces of something (Pask, 1976).

Gordon Pask's work titled "The Architectural Relevance of Cybernetics", published in the Architectural Design journal in 1969, emphasizes the importance of perceiving architecture as a dynamic and interconnected cybernetic system that reacts to feedback it receives from its surroundings (Pask, 1969). Pask renders the "predictive power" of the cybernetic theory in architecture where he suggested its ability to integrate architectural systems that are adaptive and have the ability to evolve alongside changes in the surroundings and user behaviours. This stands in contrast to traditional ("pure") architecture, which he described as primarily "descriptive", focusing on categorizing "buildings and methods", and "prescriptive", limited to the "preparation of plans" without the dynamic responsiveness which is a natural approach cybernetics (Pask, 1969). As an instance of urban development, he proposes that it “can be modelled as a self-organising system and in these terms, it is possible to predict the extent to which the growth of a city will be chaotic or ordered by differentiation” (Pask, 1969: 73). He argues that architecture, like any other complex system, can be picked apart and analysed through the lens of cybernetics while also underlining feedback loops (Pask, 1969), which encourages adaptiveness in real-time and views entities as companions for expressive exchanges (Pask, 1980).

From Glanville’s viewpoint, a student of Pask, the essence of innovative designs lies in the process of the exchange of ideas through mutual interaction (Glanville, 2014), as stated in Pask’s conversation theory (Pask, 1980). In other words, “design cybernetics” provides the abstract basis for which the understanding of how design processes work while simultaneously being actively engaged in the process of design itself (Glanville, 2014).

In a 1996 issue of The Guardian Newspaper titled “Dandy of Cybernetics”, Paul Pangaro who was another student of Pask, wrote a tribute where he talked about how both Pask and Glanville were involved in pioneering a cybernetic approach to the subject of architectural design and how they departed from traditional thinking of said framework of design (Pangaro, 1996). In addition, the concept of his vision mirrors the idea of environments able to learn where individuals are portrayed as integral elements in a continuous loop of significance (Pangaro, 1996).

3.1.2 The ‘Cyborg’ as a Symbol for Interactive Machines

Conceptualized by Clynes and Kline, in their 1960 paper “Cyborgs and Space”, with the idea of creating a human-machine construct that had the limitless abilities to independent in itself, to endure inhuman conditions like adapting to outer space as essentially a type of superhuman that was half human and half technological with electronic elements (like artificial organs and limbs), short for “cybernetic organism” (Clynes & Kline, 1960).

The cyborg surfaced during a time where technology and cybernetics were at a hyper progressive state and the public started to imagine a world where such beings with super robotic abilities exist. In Haraway’s “A Cyborg Manifesto” (1984) essay, the representations of the cyborg mirror the conceptions of “hybridity” that fade lines between a human being and a machine (Haraway, 1984). In the way it mediates the interactions we make with the world, she explores the way in which technology, as a broad term and construct, in a way reconstructs our understandings of ourselves (Haraway, 1984).

In an architectural sense and mirroring a technological being that dynamically participates with and reacts to motivations it accepts from its environment; the modern depiction of the cyborg encourages for the concept of adaptive architectural design solutions, where it uses cybernetic constructs to conversate with the other bodies around it, with the intentions of sensing and understanding the broader system it operates in. The idea of an 'architectural' cyborgs represents a conceptual linkage of technology with architectural and urban design where an entity can be configured with digital systems to enhance the adaptability and user experience in urban settings.

It is imperative to note that, while the idea of a 'cyborg' originally conveys adding technological elements into a primarily biological entity, the modern cyborg may be represented as a body that reversely adds biological or human-like attributes into a predominantly technological entity. The modern depictions of the cyborg may not necessarily symbolise an initially biological framework, and instead it can take the form of any technological entity that is mimics human-like traits.

3.2 REINFORCEMENT LEARNING

ML is an algorithm base that lets models learn from information they encounter and so improve their performances using neural network (NN) frameworks (Krogh, 2008). In "What are Artificial Neural Networks?" (2008), Krogh suggests that NN are "inspired by the early models of sensory processing by the brain" as well as that "by applying algorithms that mimic the process of real neurons, we can make the network 'learn' to solve many types of problems" (Krogh, 2008: 195). ML includes several types, with RL as the focus algorithm of this thesis, where an agent, which is a digital entity that is able to learn, essentially learns to make its own decisions when it interacts with its system surroundings (Turner et al., 2019).

In RL paradigm, the agent strategizes to make decisions and learns through interactions based on the feedback, through its NN model brain, it obtains from the system being either positive or negative. "Rewards" are positive reinforcements given to the agent and processed within their NN brains as a result of desirable behaviour, whereas "Punishments" are negative reinforcements that are a result of undesirable behaviour, and through the RL algorithm, agents are able to act a certain way to maximize their reward total by learning to increase the number of wanted behaviour and decrease the number of unwanted behaviour (Sutton, 1992). The agent obtains feedback from the system, by processing this information in its NN, being rewards or punishments which depend on the actions it takes, with the ultimate goal for the agent being “learning what to do, how to map situations to actions, so as to maximize a scalar reward signal” (Sutton & Barto, 1999: 2). The learning cycle of accumulating positive reinforcements propels the agent to increase its received number of rewards and tries to limit the number of punishments it receives (Kaelbing et al., 1996). In the conference paper “Simulation-Optimization Using a Reinforcement Learning Approach”, it was analysed that “RL is the problem faced by an agent that must learn behaviour through trail-and-error interactions with a dynamic environment” (Paternina et al., 2008: 1377).

The RL process operates initially in the environment that represents the play field that the agent uses to accomplish its tasks (Sutton & Barto, 1999). The agent “is connected to its environment via perception and action” (Paternina et al., 2008: 1377). The agent then is enforced with a system of rewards and punishments to take its next action where rewards are positive reinforcements that demonstrate wanted behaviours and punishments that are negative reinforcements that show that the behaviour performed is unwanted (Paternina et al., 2008).

In Sutton’s analogies of RL and its limitations, he states that “one of the challenges that arises in reinforcement learning and not in other kinds of learning is the tradeoff between exploration and exploitation” (Sutton & Barto, 1999: 2). Keeping a balance between these two types of learning outcomes makes sure that the agent is learning properly where it takes new explorations but also use its experiential data to obtain more rewards (Sutton, 1992).

Built on Skinner’s operant conditioning (1947), RL is a psychological basis that explains that an individual’s behaviours are a result of the actions they take (Akpan, 2020). Skinner did case studies on animals to examine how they attempt to learn actions through different consequences whether they be positive or negative (Skinner, 1938). In “The Behaviour of Organisms” (1938), Skinner states that behaviour is affected by its consequences either in the form of positive reinforcements or punishments (Skinner, 1938). He portrays that reinforcements are a type of consequence signifying the likelihood of a type of behaviour happening again or with punishments indicating the decrease in that chance of the actions being repeated.

3.2.1 RL in Architecture

As versions of computer-based models that mimic real-life scenarios, RL algorithms can be used in architectural design through simulation studies, to examine the agent behaviours in different contexts (Intrator & Intrator, 2001). More specifically, these simulation studies fall into the category of Agent-Based Modelling (ABM), which is basically a technique in computational social science that is used to hypothesize very multifaceted virtual environments that are inhabited by "autonomous" and self-referential agents (Macal & North, 2009). With their influences on recreating events that are based on real-life events, the agents used in ABM are imbued with special abilities and unique behaviours that lets them be represented as the closest version to a real human but in a simulated environment (Macal, 2016).

Using an RL-algorithm agent in a digital environment simulated to represent an urban one evokes parallels with the role of a human explorer in the real world due to the fact that individuals depend on things like sensory perception and decision-making to navigate and make sense of their surrounding physical world, using similar cognitive processes to explore their simulated settings with the enforcement of their NN brain models that act as their cognitive frameworks (Janssen, 2005). Just by uncovering and discovering patterns displayed by agents, research can, in essence, realize and learn about specific habits and trends that may relate events we perceive in our world. Agents having the potentials to intricately imitate human behaviour due to having similar cognitive processes through their NN, setting them aside as the most ideal subjects for simulation studies that aim for a richer understanding of how individuals occupy and redefine urban spaces.

Architects have used this genre of RL study, particularly through the Unity game engine's ML-Agents toolkit (2017). Established by Unity Technologies, Unity is a game engine platform used for video game development and other interactive experiences (Juliani et al., 2018). Developed in 2017, Unity's ML-Agents (Machine Learning Agents) is an "open-source project" as a Unity "plug-in toolkit" that acknowledges and uses RL algorithms in different types of projects (Juliani et al., 2018: 11), stressing that "the complexity of the physical world is a primary candidate for challenging the current as well as to-be-developed algorithms" (Juliani et al., 2018: 2).

Applications of ML-Agents can offer architects the chance to generate better designs, despite its game development purposes, in the sense that through the use of RL of agents in architecturally designed spaces they can learn to generate different design solutions. Agents can contribute to better design solutions with their ability and use to assess aspects like energy efficiency and the stability of structures. Another use of this extensive framework for architectural use is through modelling of user behaviours through agent learning to better understand how individuals behave in different urban layouts. Architects can use the knowledge gained from these experiences to understand user preference in a way where they can incorporate this information in their designs. Aside from behavioural modelling, architects can use this platform to experiment with environmental analysis designs in relation to factors like noise levels in a neighbourhood, airflow directions and sunlight advantages.

As part of her graduate master thesis "Co-creation: Space Reconfiguration by Architect and Agent Simulation Based Machine Learning" (2022), Dai used the ML-Agents toolkit in reconfiguring architectural spaces (Dai, 2022). Essentially, instead of just applying ML to agents themselves, her approach was to employs ML to walls to influence the behaviour and configuration of residential spaces. By using agent simulation as the basis of her research, the project trained architectural spaces to "reconfigure" themselves according to their closeness to other objects and agents nearby (Dai, 2022). As a control variable, Dai introduced a "co-creation" method to address the intrinsic simplifications of ML models when it came to complex architectural space applications. Dai states that this method can allow architects to use their "tacit knowledge" and maintain "real-time control over the space reconfiguration process" (Dai, 2022: 304). Ultimately, Dai's research blends ML capabilities with the intuitive knowledge and experience of architects with the aim of her research being to use the intuitive expertise of an architect with a different approach to design using ML-Agents in way to pave the way for architects to work with machines in distinctive ways.

Dai's project consisted of four main parts including "Space, Agent, Machine Learning, and Co-creation". "Space", involved a designated space required for methodology application. "Agent", is the simulation of the behaviour of the agent within the space (Dai, 2022). Next, "Machine Learning" portrayed the use of the ML mechanisms in training. Lastly, architects engage in “Co-creation” with the ML-Agents model that has been trained. Dai opted for 20 one floor residential spaces to use as study subject as “housing is historically consistent” as well as being “architecturally diverse” (Dai, 2022: 307). When converting the residential spaces into 3D models, architectural elements such as walls that could be used for representation were kept which portrays the spaces in a simple and readable manner.

The observations made by Dai included the “building’s room size, room density, and room distance” (Dai, 2022: 307). “High resolution”, among the resolutions tested, was opted for because it resulted in a larger number of walls to train with 50 million steps. Three types of agents where used, the “Master” being the “owner of the building”, the “Staff” with their job being “serving the master and the guest”, and the “Guest, who visits the master” (Dai, 2022: 307) (Figure 3.1). Essentially, in the agent’s behaviour, there is an element of “randomness” that introduces unpredictability mimicking real-life situations and further adding complexity to the environment. In this situation, the training process intentionally used complex behaviours to make sure that the agents make the most use of different rooms. The agents each demonstrate unique behaviours in how they used each space, and these preferences vary between the “guests”, “masters”, and “staff”, the paths taken by agents concordantly mirrors the frequency of space usage (Dai, 2022).

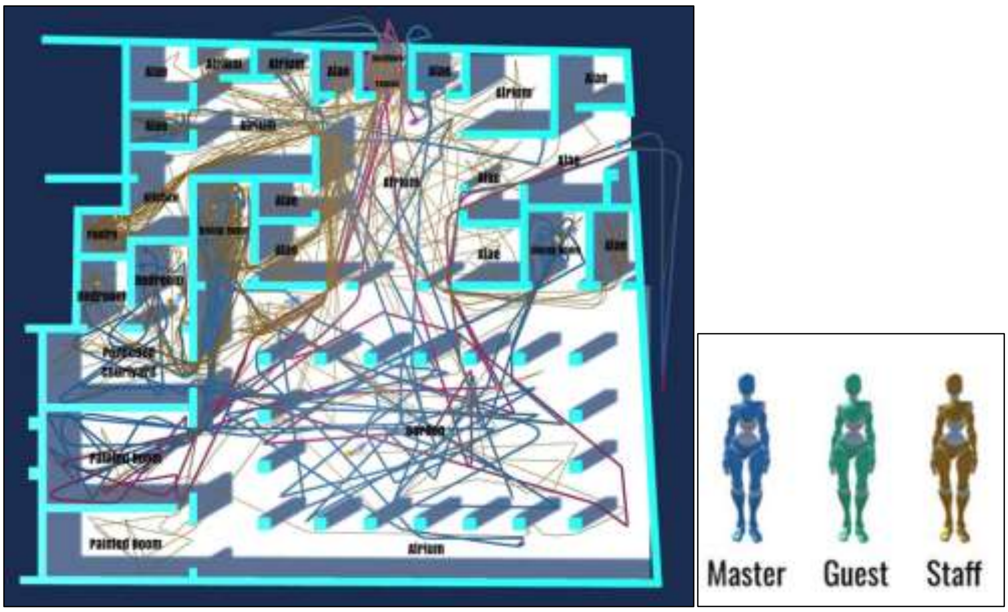

Figure 3.1: Types of Agents and Their Movements Throughout Their Environment. (Dai, 2022).

Each wall, which is categorized as a “gameobject”, was given an NN brain with the aim being to set a target distance between the gameobject and the agents (Dai, 2022). Ultimately, if an agent’s distance to the gameobject was “less than 1 meter or exceeds 5 meters”, the agent received a punishment (a negative reinforcement), shown in “red” (Dai, 2022: 309). However, if the distance fell between “1 and 5 meters”, the agent received a reward (a positive reinforcement), outlined in “green” (Figure 3.2) (Dai, 2022: 309). The training objective was to attain an optimal distance between the gameobject and the agents. Using the Raycast 3D component, the distance measurement between the gameobject and the agent is achieved. Because the Raycast 3D is attached to the gameobject, the walls are then able to identify agents “within a 10-meter radius” (Dai, 2022). After the training process, the trained brain (NN) was used for every building type and further trained as inference behaviour. In this scenario, architects use their expertise to make real-time decisions regarding room design and specifications while an ML model processes information and suggests design recommendations. The model assesses behaviours that are displayed by the agent to propose special arrangements and architects draw on their knowledge and experiences to supervise the process and rectifying any oversights the model might have neglected.

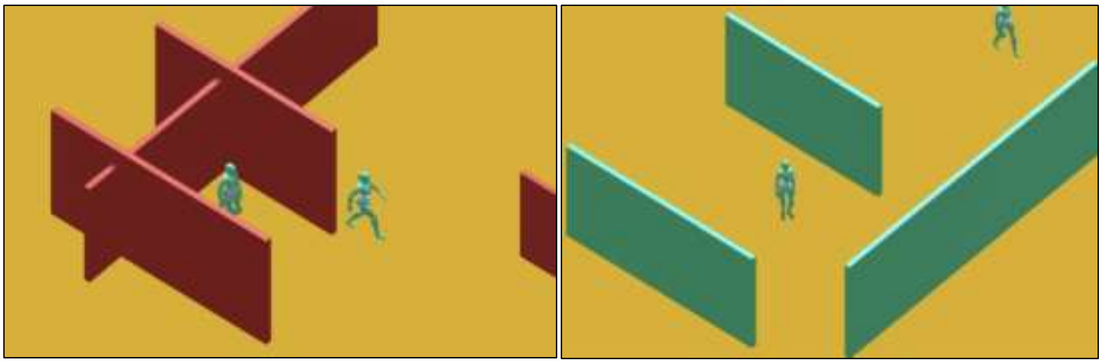

Figure 3.2: Agents Receiving Punishment (Left) and Receiving Reward (Right). (Dai, 2022)

The training results of Dai’s research indicated that the more the agents frequent a room, it tends to become larger, while it shrinks when being less frequented (Dai, 2022). However, the ML algorithm only considers how often the room is used, and neglects factors like the duration the agents spend inside and other unique purposes of different rooms. Other elements within residential architecture like the duration of when an agent occupies a room, and the functionality of different spaces should have been included in the “space reconfiguration parameters" but these factors were not fully acknowledged in the training processes leading to limited understanding of the results (Dai, 2022).

Another study by Chien-Hua Huang titled "Opportunity of Machine Learning in a Material-informed Circular Design Strategy" (2021), explores the use of RL in architectural design application using a tool He called "Reform Standard" (Huang, 2021). In Huang's study, architects use a type of digital workflow where waste geometries that are taken with RealityCapture (RC) are analysed in Rhinoceros and Grasshopper (Figure 3.3) (Huang, 2021). These geometries are then organized for assemblage in Unity's ML-Agents with the use of boundaries to later ensure practical assembly of the parts. This boundary that is shaped like a site parameter influences the overall forms while also providing design control.

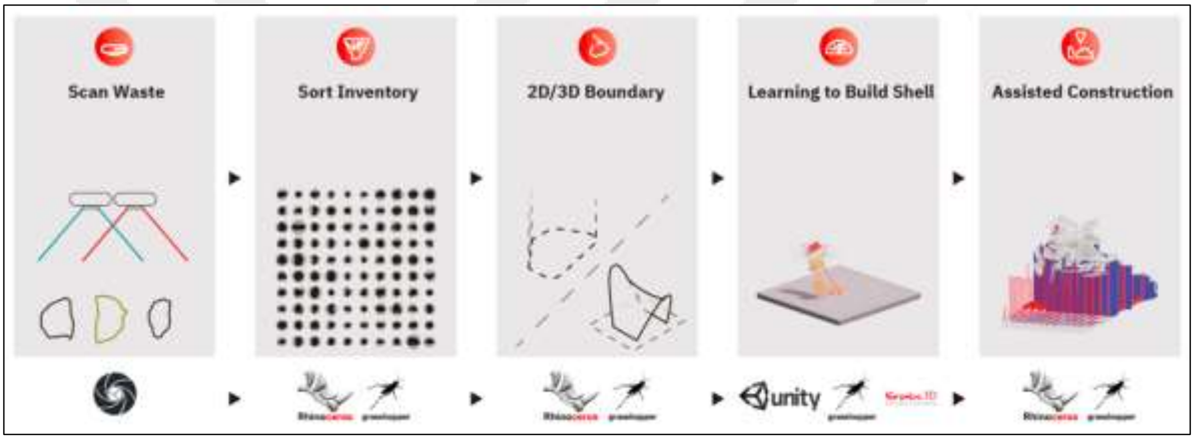

Figure 3.3: The Digital Workflow of the "Reform Standard" Project. (Huang, 2021).

The project explores the design and construction of a "structural shell" using fragmented plastic waste (Huang, 2021). In the first step, the architects input types of waste geometries which are digitalized for analysis in the programs Rhinoceros and Grasshopper. Next, every geometrical part is organized and inspected for Unity assembly. Simply, "projection areas" define the "maximum boundaries" for the assembly process where the boundary points are transferred into Unity for the ML-Agents to modify their actions based on the geometrical structures' arrangements (Huang, 2021). A "2D perimeter or 3D volume" is represented as a boundary and limits the positions of the fragments during the learning process (Huang, 2021: 174). The architect can define the assembly boundary either as a "closed polyline or a closed brep" where with the initial option there is no constraints on the assembly's height in the specific area whereas in the "3D bounding geometry" approach involves solving "bin-packing problems with random geometries" (Huang, 2021: 174). In the research it was specified that the boundaries should not exceed the combined “projection areas or volumes” of all the fragments to make sure that efficiency in RL is maintained (Huang, 2021).

Following this, the divided fragments and revised boundaries are then imported into Unity for a simulation practice to decide the most suitable “structural shell” (Huang, 2021). Huang suggests that the primary advantage of using the RL mechanism in ML-Agents for architects to no longer require to separately plan each complex process of constructing a “structural shell” because the algorithm does it well (Huang, 2021: 175). Architects instead provide a simple layout for the ML-Agents to follow where the toolkit then combines the data it gathered from occurred processes and so creates a methodology that is able to construct a “shell” dependent on different user preferences (Huang, 2021). After the initial training, the methodology is trained again in inference with a NN brain model to enhance progress. The process starts with the agent uses its observations in manipulating and reorienting a fragment with the aim of finding the best position (Figure 3.4). When assembling, the structure undergoes assessment through rewards and punishments, and considering aspects like stability and height (Figure 3.5). To summarize, the process details pinpointing the “connection points, drilling points, and their connection IDs” on the geometry with “plastic blind rivets” selected as joints for assembly due to their compatibility with “plastic fragments, and lightweight” requirements (Huang, 2021: 177).

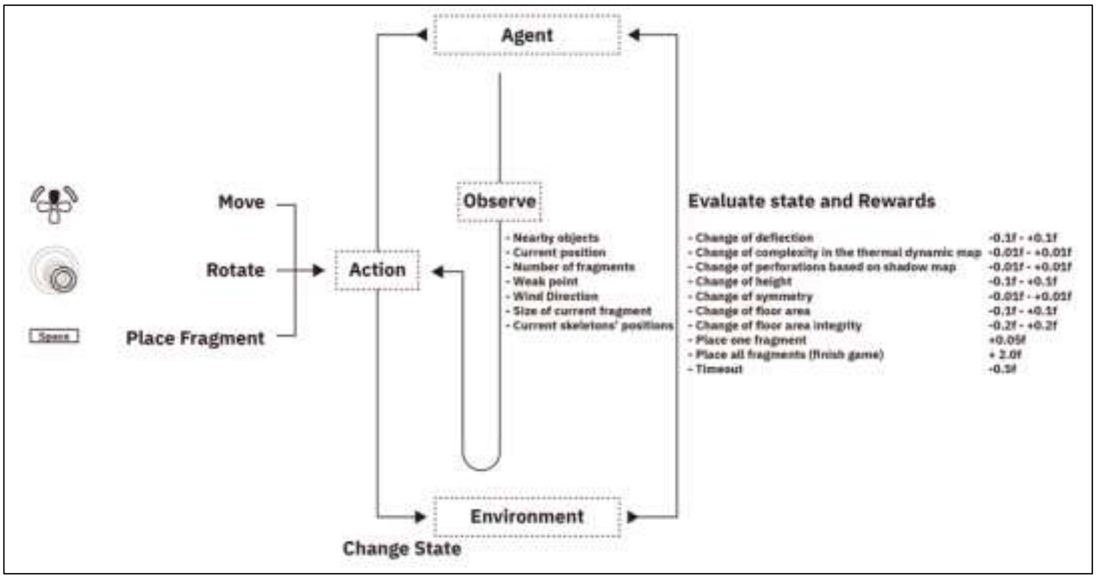

Figure 3.4: Image Showing the ML-Agents Process Setup. (Huang, 2021).

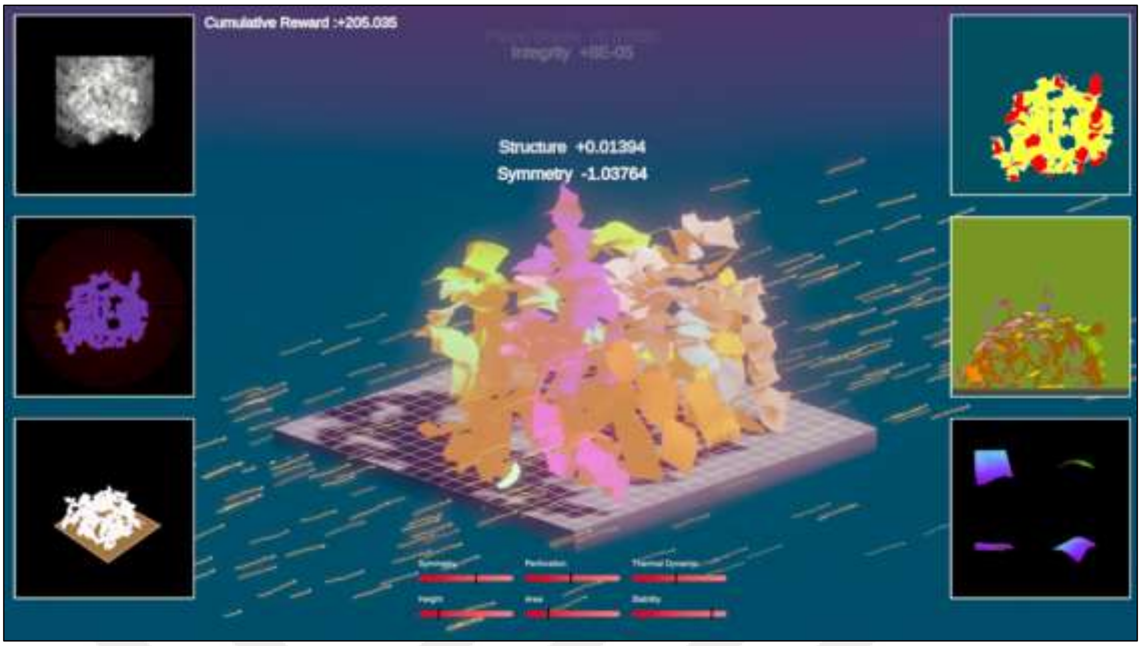

Figure 3.5: ML-Agents Unity Image During the Training Process. (Huang, 2021).

Recent tests in this study have revealed the potentials of using a "design-build" approach that essentially combines "geometrical and structural analysis" with RL uses in architectural practices (Huang, 2021). The findings of his study signified that the effective assembly of a model through physically putting the pieces together proposes the possibility of extending this application to wider varieties in structural designs (Figure 3.6). Nevertheless, there are shortcomings that include slight dissimilarities in dimensions as well as complications in modifying all the actions included in the RL instances. Despite RL's ability to offer "intuitive" solutions, its "trail-and-error" reality might cause a delay in specifying logical clarifications for some certain actions (Huang, 2021: 178). The research concluded that architects have the chance to explore the integration of ML in a type of personalised "artisanal" approach that comes with machine assembly and construction of parts. The study's approach allowed for the intelligent assembly of complex geometries that can only be achieved through machine outcomes, despite the fact that constraints of the nature of RL mechanisms. The "Reform Standard" project is a portrayal of how ML can be applied in "sustainable building practices" and provides architects and designers with insights for future research with the link to efficient building practices (Huang, 2021).

Figure 3.6: The Physical Model of the Recycled Fragments Generated by ML-Agents. (Huang, 2021).

3.3 CURIOSITY AS A MOTIVATIONAL DRIVE IN RL AGENTS

A sense that motivates individuals to want to learn, the more that they are met with these positive feelings of accomplishment, the more that individuals explore journeys that are fuelled with curiosity, ending up with experiences that speak a sense of purpose as a result. “Curiosity is an old, intriguing, and vexing construct in the psychology of motivation” (Silvia, 2012: 157).

Self-Determination Theory (2000) is built on the groundworks of curiosity as a motivational drive, it assess a cognitive framework for understanding the way that individuals behave when they are curious (Ryan & Deci, 2002). The central idea of this theory is linked to intrinsic motivation, a sense that is felt when any task of some sort is completed such as when obtaining rewards, resulting in feeling fulfilled (Deci & Ryan, 2008).

In curiosity-driven learning, a foundation based on intrinsic motivation in RL, the agent is motivated by its intrinsic curiosity to collect rewards when it explores new areas in its environment (Burda et al., 2018). In “Large-Scale study of Curiosity-Driven Learning”, Burda et al. found that when using intrinsic rewards, the agents’ learning speed was faster than that of extrinsic (Burda et al., 2018).

The frameworks of familiarity and schemas are relatively associated with the concept of curiosity and its influences on learning in individuals and is assessed in various studies (Bartlett, 1932). Bartlett’s research on memory recall explores the influence that schemas have on the reconstruction of memory and essentially how familiarity affects the way individuals remember and interpret events (Bartlett, 1932). The feeling that comes with recognition and comfort is linked to the term familiarity where individuals encounter a stimulus that they have previously interacted with or are already acquainted with. On the other hand, schemas are a type of cognitive framework that recognizes and reconfigures types of encountered information depending on aspects like past knowledge and experience as well as expectations.

When an individual is hit with a sense of familiarity, they may experience less motivations due to the feelings of comfort and predictability that accompany it (Bartlett, 1932). The senses of curiosity can arise when individuals are met with information that does not match the one in their existing schemas which prompts them to explore and revise their understanding of this information. While familiarity can influence the way individuals are affected by curiosity, the role that schemas play in reorganizing and interpreting information acts as a base for an individual's learning process.

3.3.1 The Flâneur's Explorations of the City

City discovery as a flâneur, an approach to urban exploration, is not just about actual explorations but also associated with intellectual and emotional interactions that imply picking apart the layers of the city in order to uncover its true meanings and cultural identities (Boutin, 2012). The ideas of the flâneur are in relations to psychological principles that are curiosity, motivation, familiarity, and schemas, in providing a unique lens through which the flâneur views urban exploration as well as being motivated by curiosity which prompts them to engage in aimless explorations and keen observation of the cityscape. Their journeys through the city streets are led by motivated curiosity in a quest to seek out new experiences and perspectives in association with both familiar and unfamiliar neighbourhoods (Hazel, 2012). While they may be familiar with certain areas in the city, their aimless navigations more often than so leads them to explore unfamiliar neighbourhoods and hidden corners that challenge their existing perceptions and schemas in helping them interpret the city.

Lynch argues, in "The Image of the City" (1964), that cities are not just physical entities but are also perceived as psychological and symbolic components of the bigger picture that are conceptualised through the individual's perceptions and interpretation (Lynch, 1964). He conducted research in Boston, Jersey City, and Los Angeles in regard to this topic where he identified five significant elements that add to the true essence and coherence of a city being the “paths”, “edges”, “districts”, “nodes”, and “landmarks” (Lynch, 1964). These elements are identified as fundamental parts that an individual uses to develop their urban mental map. In the book Lynch also explores impact of familiarity and memory as subjective features in the formation of an individuals’ interpretations of the city where he highlights the prominence of sensory experiences and spatial relationships in the development of mental maps that are associated with urban settings.

As an example of how Lynch’s key elements are linked to curiosity, paths are representations of linear elements aligned with movement in the city and encourage individuals to explore new areas out of curiosity due to their simple configurations (Lynch, 1964). Another example being landmark that act as the referential points in an urban setting to attract an individuals’ curiosity and again encourage them to further explore because of their distinct features that sets them apart from other elements. When individuals explore districts that are characterized for representing recognizable and familiar areas, they encounter feelings of familiarity and comfort when perceiving these familiar territories (Lynch, 1964). From the flâneur’s perception of the city, it uses these urban elements, as a basis, to guide its way around its environment, motivated by curiosity to learn as much as it can about it.

3.3.2 The Agent as the ‘Cyborg Explorer’ of the City

In this context of an environment’s explorer, the agent can also fit into these representations as its explorations include navigating its digital environment thoroughly in an attempt to interact with other entities and ultimately gather data to make sense of how the system operates. While it displays adaptive exploration behaviours and regulates its exploration patterns based on its current conditions, its schema may regularly update its knowledge in its NN using past experiences and learning progress. The agent, outlined as the ‘cyborg explorer’ in this thesis, has the motivations to represent the flâneur as an explorer of its environment where just like the flâneur, the agent curiously navigates its digital atmosphere, and lets these senses guide the way that it behaves, and as a result constructs a mental map of its surroundings where it identifies hidden features and patterns that shapes its understanding of it. As it takes on the attitude of the flâneur, with curiosity as a foundational principle, it can take part as a cyborg explorer of the digital landscape on a journey of exploration and learning.

4. METHODOLOGY

As part of this thesis' research, a simulation study is featured to understand how through simulating a cyborg explorer's curiosity, a technological agent with the outlook and approach of a flâneur, within an urban scale to interpret its behaviours and how motivational curiosity pushes it to act in a city scape. A simulation study uses digitalized models that imitate real-life events where they are used to conduct trials, study behaviour, as well as make hypothesis about distinctive circumstances to reproduce the reality of real-world agendas (Intrator & Intrator, 2001). Using a simulation study for this research matters because investigations based on conventional studies or simply observing no longer leads to in-depth engagements and understandings in everyday lives, so using a simulation study allows researchers to really recreate real-life settings and learn from these outcomes. More specifically, this simulation study falls into the class of ABM, an "approach to modelling systems comprised of autonomous, interacting agents", used to theorize very complex virtual environments that are resided by independent and self-referential agents (Macal & North, 2009: 86).

Simulation study processes happen following similar sequences where firstly a digital environment is created that incorporates the essential elements needed to mimic certain scenarios (Intrator & Intrator, 2001). To fully be emersed and be able to analyse the impact of simulation studies on real-world settings, various tests and trials should be made, with different input and control variables to make sure that the settings resemble the real scenarios as much as possible (Maria, 1997). Once the environment is appropriately designed and tested, the simulation model is run using software that obtain information that mirrors the system's behaviour over time and then analyse the data outputted. In that, with their impacts on reconstructing occurrences that are centred on real-life incidents, the agents used in ABM are instilled with exceptional talents and exclusive behaviours that lets them be characterized as the nearest form to an actual human being but in a simulated setting (Macal, 2016). Just by uncovering and discovering patterns displayed by agents, researchers can, in essence, realize and learn about specific habits and trends that may relate to events we perceive in our world (Janssen, 2005).

For this study, this ABM simulation method is applied in use of an RL algorithm, which uses an agent, being a digital entity, to be able to learn from its actions and decisions in "learning what to do, how to map situations to actions, so as to maximize a scalar reward signal" (Sutton & Barto, 1999: 2). Additionally, to use this RL algorithm, Unity's ML-Agents toolkit, an "open-source Unity plug-in toolkit" that recognizes and uses RL algorithms in simulation types of projects (Juliani et al., 2018), is applied to simulate and understand the way that the cyborg explorer behaves according to curiosity in an urban scape. This algorithm and tool is used to mimic an urban environment with a cyborg explorer that parallels the role of a human explorer in the physical world, using similar cognitive processes, fuelled by motivated curiosity, to explore their simulated urban setting.

4.1 UNITY ML-AGENTS TOOLKIT USING RL

Unity's ML-Agents toolkit is the tool used in this simulation study, being an "open-source project which enables researchers and developers to create simulated environments", that was developed by Unity Technologies in 2017, as a type of platform that recognizes the links that RL bring in teaching independent agents, that use these mechanisms, to learn from their actions (Juliani et al., 2018: 4). RL, being the algorithm that employs an agent in learning, through reinforcements (reward or punishment), to "maximize a scalar reward signal", where "rewards" are positive reinforcements obtained by the agent and managed within their NN brains as a result of wanted performance, while "punishments" are negative reinforcements that are a result of unwanted performance, and so within the RL algorithm, agents are capable of acting in a manner to increase their reward total by learning by repeatedly showing wanted behaviour and decreasing unwanted behaviour (Sutton & Barto, 1999: 2). With the use of RL, ML-Agents are "powerful tools for the simulation of visually realistic worlds with sophisticated physics and complex interactions between agents with varying capacities" (Juliani et al., 2018: 2).

The unity interface essentially consists of "three core entities" being "sensors, agents, and an academy", with the agent being able to "collect observations, take actions, and receive awards", where it "can collect observations using a variety of possible sensors" and "the academy is a singleton within the simulation, and is used to keep track of the steps of the simulation and manage the agents" (Juliani et al., 2018: 12). Here, agents copy certain human behaviours through learning from these component as "exploiting a human's domain expertise speeds up learning and helps the agent learn to behave in a manner aligned with human expectations" (Juliani et al., 2018: 20). In Unity, a "target" is a GameObject that attracts the agent to obtain rewards (Juliani et al., 2018).

The ML-Agents interface uses RL algorithms through an interconnected Python system where programs like TensorFlow and PyTorch push the Python C# scripts to be integrated, in training agents to use their NN in things like interactions and movements (Juliani et al., 2018). The "training" of an agent is an RL process that is in a form of sessions, where the agent can "collect observations, take actions, and receive awards" to improve its NN's decision-making (Juliani et al., 2018: 12). In the Python C# script, a "reward function, used to provide a learning signal to the agent, can be defined or modified at any time during the simulation" (Juliani et al., 2018: 12). The agent's reward definition shows the code instructions for receiving rewards and punishments as they guide the agent toward more preferred outcomes and limit its encounters with punishments. As the agent reaches a desired target in the environment it receives rewards as a result of this interaction. Conversely, when the agent interacts with an object defined to be undesirable it results in a punishment for the agent (Juliani et al., 2018).

When developing the algorithm in the ML-Agents toolkit application, firstly creating the playfield that is to become the environment for learning where the guidelines and intentions are expressed through reward definitions for the agent to follow (Juliani et al., 2018). After the environment is created, is designing the agent and its behavioural parameters, sensors, and other inputs that impact the way it learns in the favour of acquiring rewards, with the core being the type of sensory inputs as well as behaviour parameters that define their behaviours, where a Raycast Perception Sensor and Decision Requestor components are attached to the agent. The agent's reward definitions are set so that they encourage positive behaviours and discourage negative outcomes as well as setting a balance between guiding the agent with appropriate feedback. The target is designed and defined with a specific location on the digital environment for the agent to locate in order to obtain rewards.

The reward design ultimately defines the whole behaviour of the agent and uses unique methods in the Python C# script, which let it train using RL algorithms, and initiates it by importing the base namespaces from Unity and ML-Agents (Engelbrecht, 2023). This study's C# script specifies a class with the name "AgentIstanbul" which gets agent qualities from the "Agent" class as part of the ML-Agents toolkit package. Inside this class, two private variables, titled “target” are expressed and “serialized” to grant editing in the Unity Editor. Methods used like “Initialize”, “OnEpisodeBegin”, and “CollectObservations” are used to run starting points, episode beginnings, and the collections of environment observations (Engelbrecht, 2023), respectfully. Also included in this agent’s script are the functions for action processing, heuristic operations, and collision detection, aiding in the progression of the agent’s learning capabilities.

During the RL training process, the agent’s performance is evaluated through direct and detailed observations, in order to make note of and interpret the agent’s behavioural patterns displayed within the environment. All the training sessions are observed and supervised to record the performances and routes taken by the agent in the form of diagrammatic maps of the digital environment.

4.2 STUDY AREA AND ITS CRITERIA

This simulation study, with the ABM method, studies the effects of curiosity as a motivational drive on an agent’s behavioural outcomes in one of Istanbul’s urban settings, more specifically in Sultan Ahmet Camii’s (Blue Mosque) neighbourhood with the agent’s target being the Sultan Ahmet Camii landmark. A selected area from Eminönü, a central business-oriented waterfront region in Istanbul located primarily within the Faith district, was used as a basis for the study (Figure 4.1). More specifically, the study selects a neighbourhood surrounding the Sultan Ahmet Camii, infamously known as the Blue Mosque (Figure 4.2). The site location was selected on 2nd of January 2024. The reason that this neighbourhood was selected is because of the fact that the Sultan Ahmet is widely known by many as an iconic landmark and its reputations as a tourist attraction, built in 1916, and serving as a symbol and integral part of Istanbul’s cultural, historical, and religious identity (Figure 4.3).

Figure 4.1: Selected Sultan Ahmet Neighbourhood Site Location. (Schwarzplan, 2022).

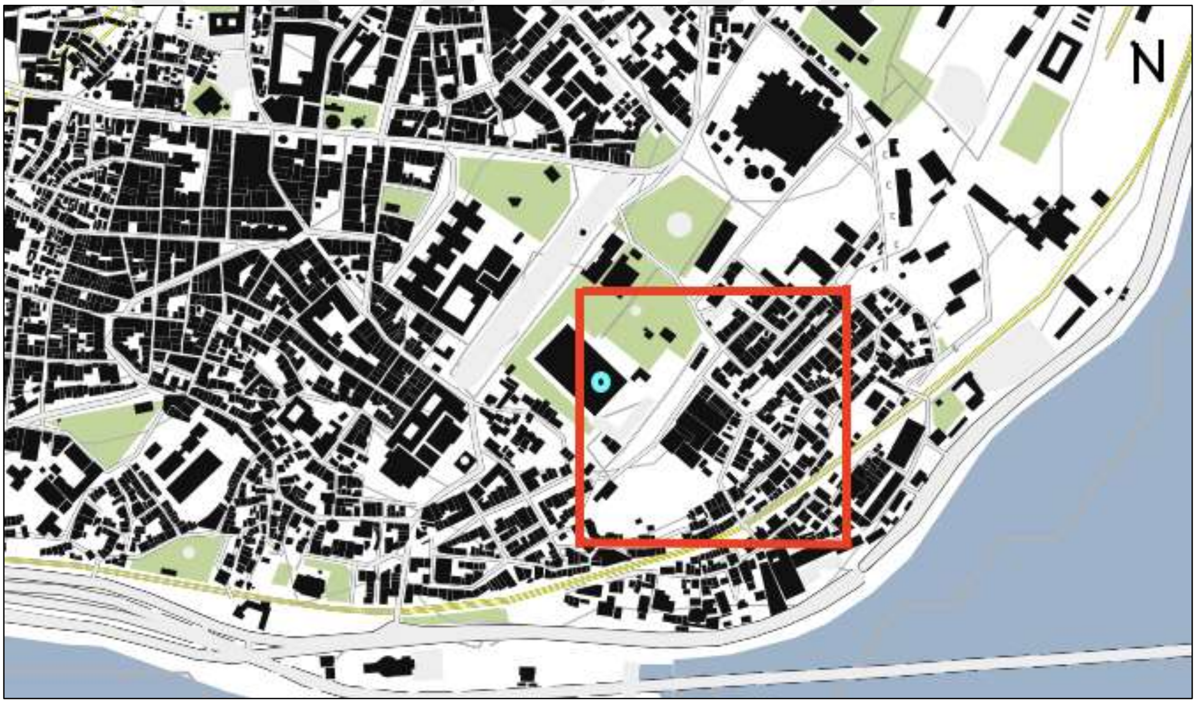

Figure 4.2: Sultan Ahmet Neighbourhood Site Location with Landmark. (Schwarzplan, 2022).

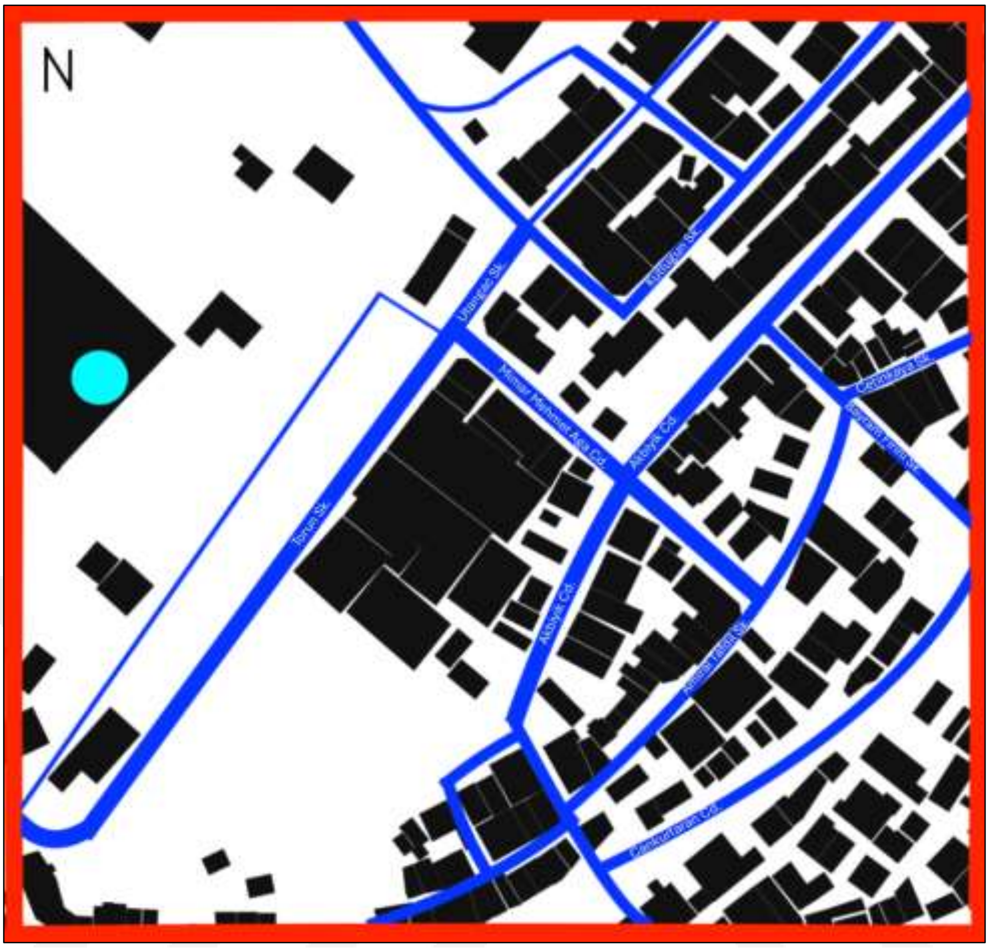

Figure 4.3: Site Plan with Agent's Target as Blue Mosque. (Schwarzplan, 2022).

The study uses the Sultan Ahmet Mosque landmark as the agent's target location, using Lynch's insights on cognitive navigation, where he insists that landmarks are elements in the city that are easy to find because of their significance in acing as identifiable points due to their recognizable features and locations (Lynch, 1964). This attitude of perceiving the urban setting is applied to the agent's goal, using the landmark as their target goal and therefore as a central reference point for competent navigation and goal attainment.

The selected Sultan Ahmet neighbourhood's morphology can be described as a loose grid of an urban typology (Wheeler, 2008), that includes 3 central roads leading to the city's commercial centres, 3 secondary, and 3 tertiary streets. In addition to that, the buildings in the selected area total 262 with the majority of their functions being hotels and restaurants.

The 3D version of the environment was extracted from Topoexport, a tool for exporting types of topography, as an OBJ file to be imported into Unity. After importing the 3D digital map of the Sultan Ahmet neighbourhood into the Unity scene as a foundation for the digital environment, it was scaled down to a scale of 3 in Unity. By using a smaller-scale environment like this, better performance, development processes, facilitating testing and debugging are more easily implemented. As the imported site's complexity is quite high as a testing environment for agents, the site was adjusted slightly to encourage a simpler setting for agents explorations (Figure 4.4).

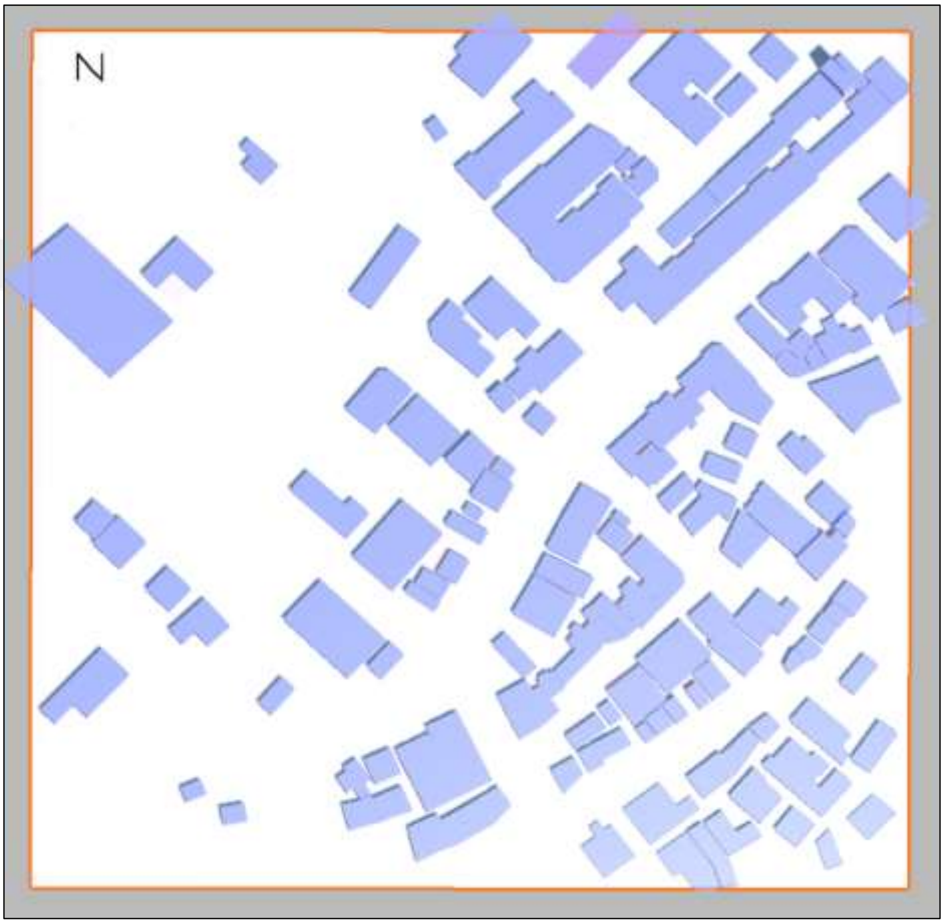

Figure 4.4: Imported Site Map as .OBJ File Into Unity Scene as a Plane. (Created by Author).

The limitations of this study are that this study excludes the streets from the environment to speed up the RL process during training, this decision was influenced by their intricate nature and given the likelihood of agents making random movements and exploring random paths. Also, to further simplify the environment's complexity, the site plan underwent alterations, including flattening and removal of terrains using the Sketchup program and importing the map back into Unity and as a result, a flat surface was adopted as the foundation, upon which the building blocks were positioned which facilitated smoother movement for the agents within the environment.

4.3 SIMULATION SETUP IN UNITY ML-AGENTS

When designing the environment, as part of the modification made to make it simpler for training, the existing buildings (purple blocks) were slightly spaced out to allow the agent to easily find the predefined target (the Sultan Ahmet Mosque) and allow for smoother training sessions (Figure 4.5).

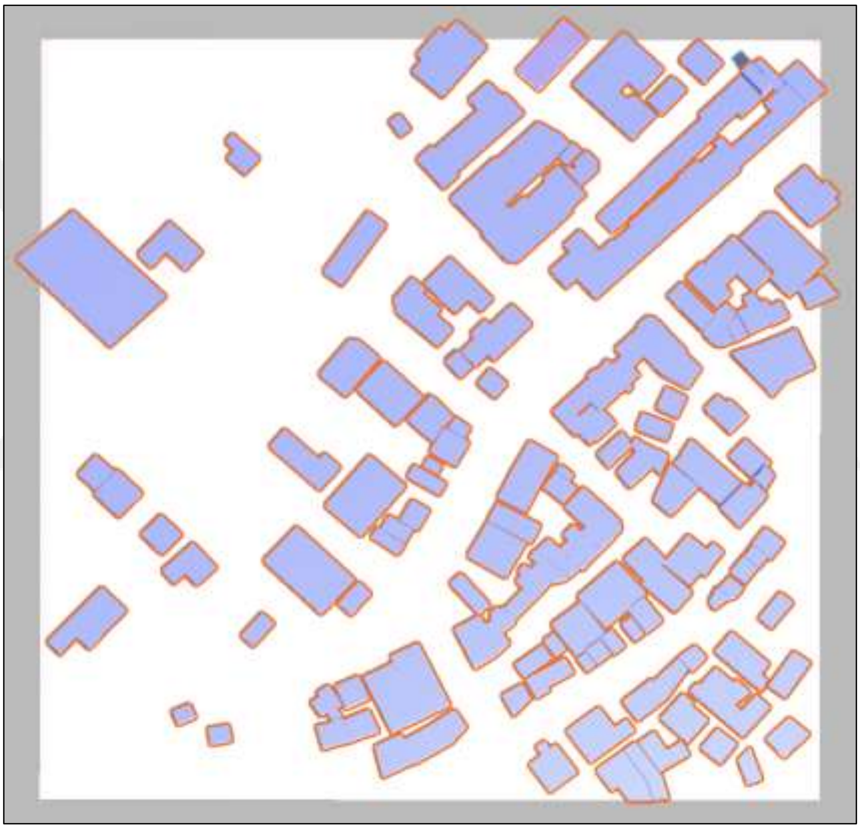

Figure 4.5: Buildings Modified to Promote Simpler Learning Environment for Agents. (Created by Author).

The building blocks are attached with a box collider component, which gives objects their physical boundaries (Engelbrecht, 2023), with the "Is Trigger" property option enabled so that when the agent interacts with the building blocks, a punishment is triggered in the agent, which tell it that this action is undesirable, and it should avoid the building blocks each time. In addition to the building blocks, side walls surrounding the vicinity of the environment are also provided a box collider with the "Is Trigger" option enabled (Figure 4.6). This application is due to the fact that, agents may pass through surfaces or fall off. So, without side walls capable of triggering a punishment, the agent will likely physically go through the building blocks and fall off the plane of the environment.

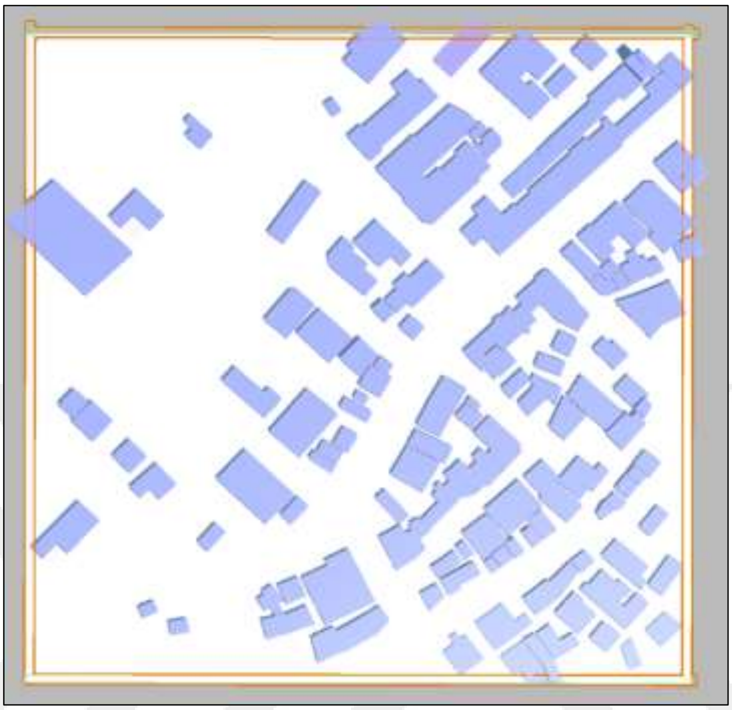

Figure 4.6: Side Walls Provided with a Box Collider with the "Is Trigger" Enabled. (Created by Author).

In terms of the target design, the purpose of this study's target is to serve as a central reference point for the agent to interact with and attain during training, assigned as the Sultan Ahmet Camii landmark. Here, the target's purpose signifies the Sultan Ahmet Mosque landmark location central for the agent to attain through exploration, using its curiosity as a motivational drive, relating it to the way Lynch describes finding landmarks using mental maps and cognitive navigation (Lynch, 1964), where while exploring the environment, the agent forms mental maps based on its experience of routes and views the target as a reference point to maximize its reward total.

When designing the target, a simple red sphere shape was opted to represent the target, imitating the shape of a pin drop on a map (Figure 4.7). Its scale was changed to suit the environment and be in appropriate ratio to the building blocks, with a scale of 2 (Figure 4.8). Its location was assumed the specific location of the Sultan Ahmed Mosque, acting as the reference goal point for the agent. In addition, similar to the building blocks, the target was provided with a box collider with the “Is Trigger” option enabled and therefore, when the agent engages with it, a reward is triggered in the agent, which tells it that this action is wanted, and that it should find the target each time. Also, the target does not change locations while the agent does.

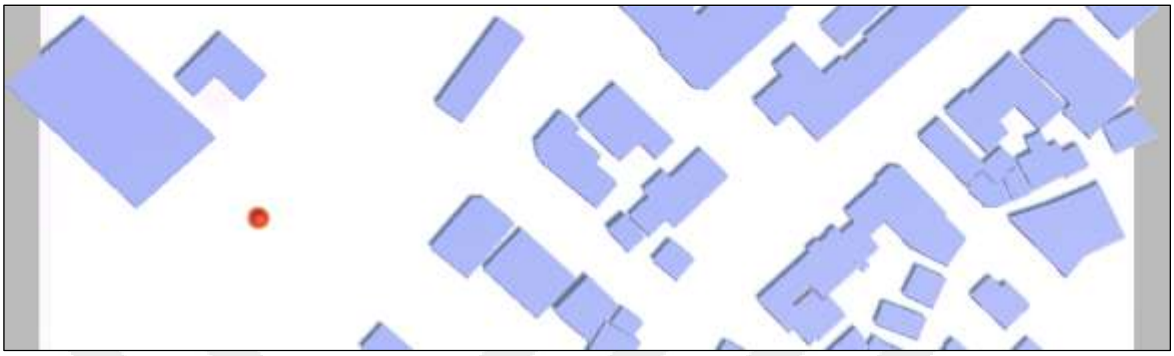

Figure 4.7: The Target With the Location of the Sultan Ahmed Mosque Landmark I. (Created by Author).

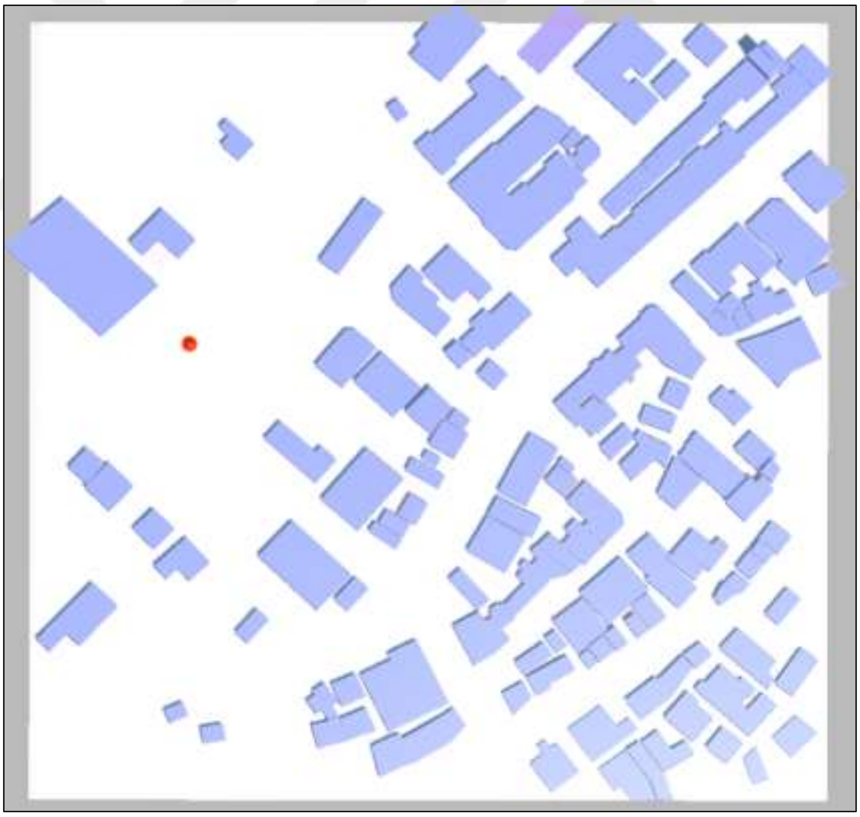

Figure 4.8: The Target With the Location of the Sultan Ahmed Mosque Landmark II. (Created by Author).

ML-Agents are the epitome of intelligent behaviour and by outlining their rewards inside the Python C# script, the agents is able to observe, act, and collect data from their environments (Juliani et al., 2018). When it came to designing the agent's form, a minimal shape was chosen. Also, to keep the overall environment simple and cohesive for the agent to interact with, a blue cube was opted for as the shape of the agent, just like the shape of the building blocks. And just like the digital site environment was scaled down for simplification reasons, the agent was also scaled down (Figure 4.9). While the building blocks have the scale of 3, the target scaled as 2, the agent has that of 1, in an effort to make the site dimensions more comprehensive and in a ratio that mimics real-world settings. The agent is also provided a box collider but here, the agent's "Is Trigger" option is not enabled because of the fact that if both the agent and the object it interacts with have their box colliders set as triggers, they simply will pass through each other without triggering an action outcome and for this reason, only the objects the agent is able to interact with have this option enabled, while the agent does not.

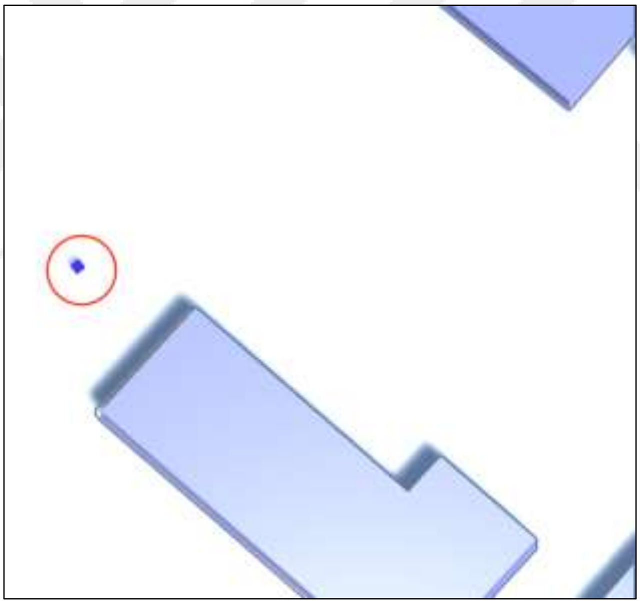

Figure 4.9: The Agent Surrounded by Building Blocks. (Created by Author).

The "Rigidbody" component that is attached to the agent, has a mass of 1, permitting it to engage with other objects in a moderately balanced manner, while also having a pulled downwards gravity trait to replicate real-life.

In the "Behaviour Parameter" component added to the agent, in the "Behaviour Type" setting in the behaviour parameters, the behaviours "Heuristic Only", "Inference Only", and "Default" are displayed for training. In the "Heuristic Only" behaviour type, the way that the agent acts is determined by custom rules predefined and is often used for testing and debugging reasons whereas the “Inference Only” behaviour allows the agent’s behaviours to adjusted by a trained NN model as the agent learns from its interactions and continuously updates its NN to improve its overall performance (Lanham, 2018). The “Default” option is chosen due to the fact that there is no developed NN brain provided before training, this option is automatically selected by Unity for this reason.

Another component added to the agent is the “Ray Perception Sensor 3D”, which is used to perceive the environment through raycasts that are attached to the agent’s gameobject in the Unity scene (Juliani et al., 2018). These perception sensor rays work in the form of virtual sensors that let the agent to mimic the ability to “see” or “detect” objects from its point of view, similar to how real-world sensors work. In this study, these sensors are the agent’s eyes that permit it to only see things in its field of vision, promoting it to explore further in order to find the target within the environment (Figure 4.10). In the Ray Perception Sensor 3D component, collision tags “PinBall” and “Wall” as part of the “Detectable Tags”, are created for triggering reinforcements, rewards for “PinBall” and punishments for “Wall” are added as detectable tags for the sensor to recognize. This is done so that, during ray casting processes in the training, the sensors release rays from the agent’s local position to detect intersections with the target and walls (building blocks and side walls), that are provided with tags.

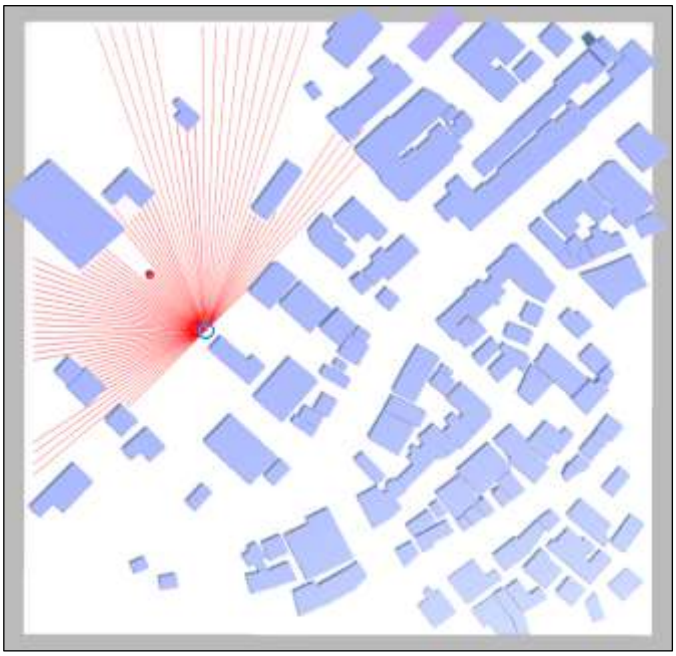

Figure 4.10: The Agent’s Sensor Rays Detecting the Target. (Created by Author).

When the agent's ray sensors perceives a "Wall" tag, either a side wall or a building block, it sense that interacting with it leads to getting a punishment, whereas when it perceives a "PinBall", which is the target's tag, it senses that engaging with it leads to getting rewards (Figure 4.11). The sensor is modified with the length and degree of the rays customized to best mimic a human's prereferral degree of vision. Because of this customization of the sensor's rays, the agent's perception is limited to the area it explores. This means that to find the target, the agent must curiously explore the whole environment in order to find it, with the motivation of receiving rewards (Figure 4.12).

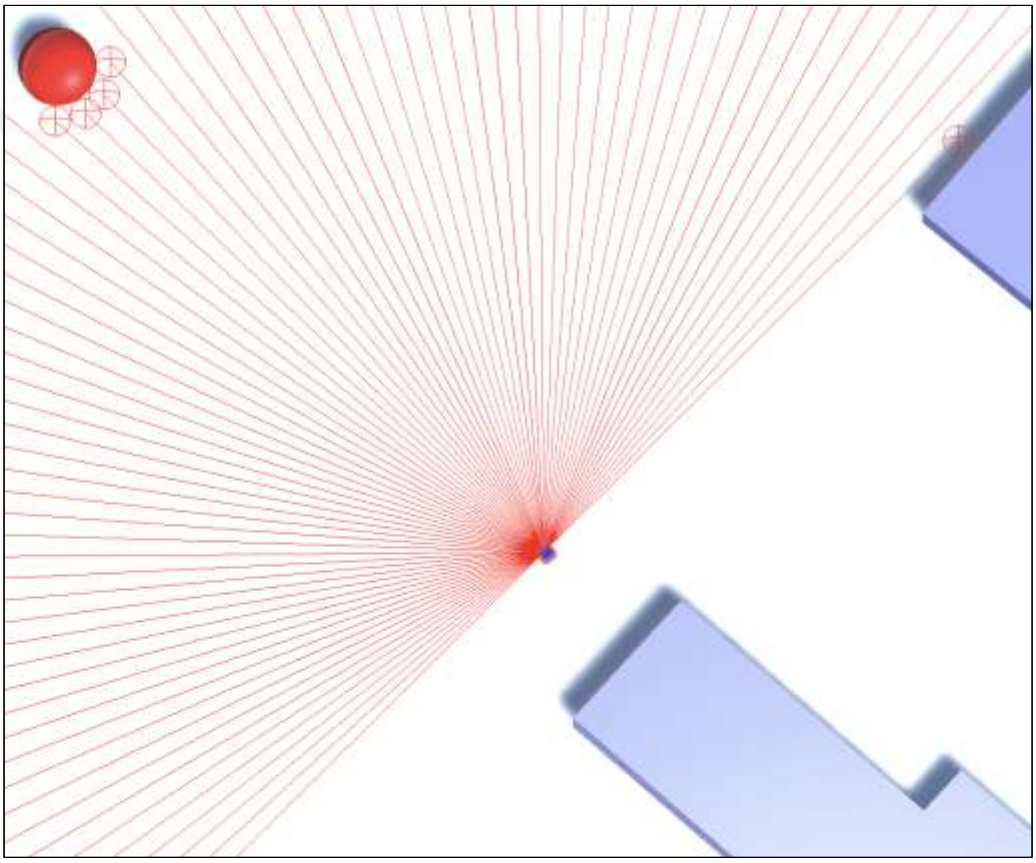

Figure 4.11: The Agent's Ray Sensors Detecting the Target and Building Blocks I. (Created by Author).

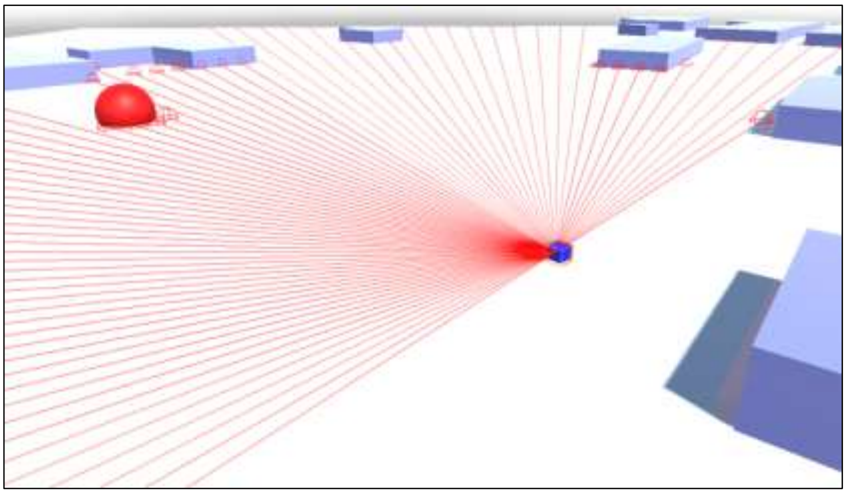

Figure 4.12: The Agent's Ray Sensors Detecting the Target and Building Blocks II. (Created by Author).

The reward design is ultimately how the agent is instructed move as they are what guides its NN brain to be able to make decisions and improve at doing that over time (Juliani et al., 2018). These methods are essentially split into two parts being how the agent acts and how the agent reacts (Figure 4.13) (Engelbrecht, 2023).

```
public class AgentIstanbul : Agent
{
    [SerializeField] private Transform target;
    [SerializeField] private float moveSpeed = 10f;

    private Rigidbody rb;

    public override void Initialize()
    {
        rb = GetComponent<Rigidbody>();
    }

    public override void OnEpisodeBegin()
    {
        //Agent
        transform.localPosition = new Vector3(24f, 0.05f, -18f);

        //BallPin
        target.localPosition = new Vector3(-21f, 0.2f, 6f);
    }

    public override void CollectObservations(VectorSensor sensor)
    {
        sensor.AddObservation(transform.localPosition);
        //sensor.AddObservation(target.localPosition);
    }

    public override void OnActionReceived(ActionBuffers actions)
    {
        float moveRotate = actions.ContinuousActions[0];
        float moveForward = actions.ContinuousActions[1];

        rb.MovePosition(transform.position + transform.forward * moveForward * moveSpeed * Time.deltaTime);
        transform.Rotate(0f, moveRotate * moveSpeed, 0f, Space.Self);
    }

    public override void Heuristic(in ActionBuffers actionsOut)
    {
        ActionSegment<float> continuousActions = actionsOut.ContinuousActions;
        continuousActions[0] = Input.GetAxisRaw("Horizontal");
        continuousActions[1] = Input.GetAxisRaw("Vertical");
    }
```

Figure 4.13: Part of the Script that Allows the Agent to Act. (Created by Author).

The first method used, contains setting up the agent's behaviour upon its starting point when the training starts (Engelbrecht, 2023). Firstly, a C# class named "AgentIstanbul" inherits from the ML-Agent's "Agent" class which entitles that "AgentIstanbul" (the name of this agent's script) is expected to receive agent functions and qualities from the ML-Agents base "Agent" class. The "target" variable here is used to allow for the assignment of a specific target object directly from the Unity Editor, which is chosen as the Sultan Ahmed Mosque landmark in the digital environment.

The next step is used to reset the agent and the target's positions for every episode, where each episode is a series of encounters between an agent and its environment (Engelbrecht, 2023), the agent's position is reset to a chosen location at the beginning of each episode. The agent's respawn location is modified a few times, to study the effect of curiosity as a motivational drive on its behaviours from different position coordinates. Three position coordinates, according to difficulty level, will be tested for the agent's behaviours: (7.3, 0.05, -30), (20, 0.05, 11.5), (24, 0.05, -18). The reason for not randomizing the position coordinates of the agent every episode is because of the implication of the agent randomly respawning inside a building block which causes a bug in the system. Also it is used to reset the location of the target, that is referred to as "BallPin", respawns it to the fixed location coordinate (-21f, 0.2f, 6f), the location of the Sultan Ahmet landmark.

The following method is applied as it is accountable for collecting observations that are pieces of information that outline things like the location of objects and their distances where it is then arranged into an organized observation vector and is passed to the NN model brain (Engelbrecht, 2023). The method "CollectObservations" as a whole permit the agent to see and perceive its surroundings and therefore learn to interact with it.

Next, this method contains the actions formed by the agent, through its NN, that outputs a set of continuous actions that are expressive of the agent's ulterior behaviour, like when the agent detects the target, the NN understands that this object leads to a positive reinforcement, and so it allows the agent to move towards it. The instructions in this method control the agent's movements by like letting it move forward along the X-axis while and allowing rotation to the agent based on the actions it receives and controls the degrees of rotation around the vertical axis along the Y-axis.

The subsequent method is only used as testing strategy because of the fact that training in the early phases tend to lack data or the environment may be too complex for the agent to interact with. It is used to manually control inputs from the keyboard to be reflected onto the agent's actions, like moving left and right as well as forwards and backwards.

In the last code part of the agent script (Figure 4.14), this last method is used if the agent interacted with a gameobject. If the object has a "PinBall" tag (only attached to the target object), a response a reward of 17f (a positive reward for reaching the target) is given to the agent. On the other hand, if it interacted with a gameobject that has a "Wall" tag, attached to all the building blocks and side walls, as a result this method punishes the agent with a negative reward of -5f (a punishment for hitting the wall).

```
private void OnTriggerEnter(Collider other)
{
    if (other.gameObject.tag == "PinBall")
    {
        AddReward(17f);
        EndEpisode();
    }
    if (other.gameObject.tag == "Wall")
    {
        AddReward(-5f);
        EndEpisode();
    }
}
```

Figure 4.14: The Part of the Script that Allows the Agent to React. (Created by Author).

Regarding the testing and training phase, in the behaviour parameters, the "Behaviour Type" is changed from "Default" to "Heuristic Only" to make sure that the environment and the agent are working properly. The "Behaviour Type" is then changed back to "Default" to start training. Before starting the training session in the PC's terminal window, the environment is copied 72 times. This is so that agents in each environment case can then learn independently and self-sufficiently from one another, essentially without being persuaded by the other gameobjects (Figure 4.15). Also, this is done to speed up the RL training processes.

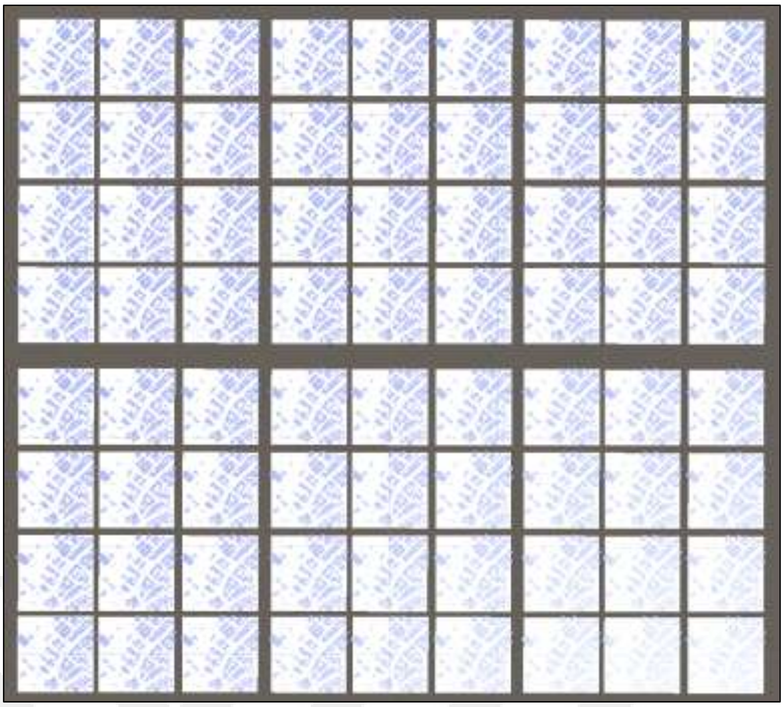

Figure 4.15: The Digital Environment Copied 72 Times to Speed Up the Training Process. (Created by Author).

With the Python script being executed in the PC's terminal window, which works as the catalyst where with this command, the training sessions start (Engelbrecht, 2023) (Figure 4.16). The agent's performances are presented with values, where the "Mean Reward" portrays that a higher value means better performance whereas a "Std of Reward" implies that when it's a higher value there is better inconsistency in the agent's performance (Engelbrecht, 2023).

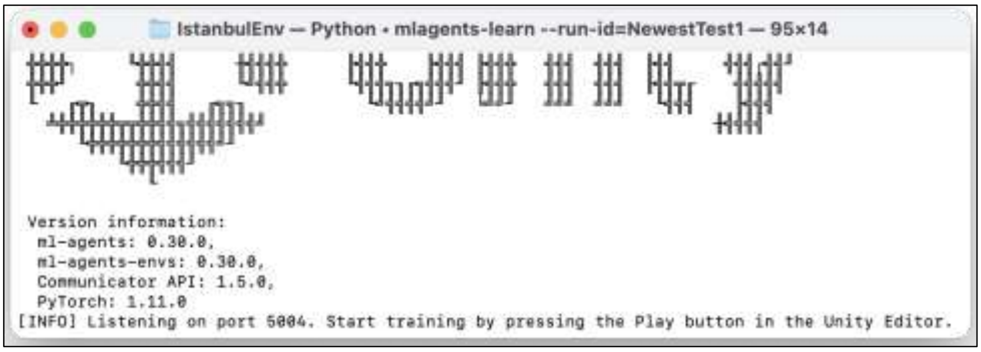

Figure 4.16: The Training Sessions Occur in the PC's Terminal Window. (Created by Author).

When the agent reaches its maximum steps, here it was 500,000 steps, the training process then end as one full learning session. Every training session had the 500,000 predefined training steps and lasted for 25 minutes. If it is assumed that every agent step takes 0.00005 minutes to complete, then with 500,000 steps every episode would essentially last for 25 minutes, calculated as so:

Total time = Number of training steps x Time per step.

Total time = 500,000 steps x 0.00005 minutes/step.

Total time = 25 minutes.

The entire training is categorized into two main parts designed to compare the agent's behavioural pattens without an NN brain model and then with one, characterized as "No Brain" and "Brain" agents. The agent without an NN brain was trained for 150 minutes that resulted with an accumulative of 3,000,000 steps taken. Likewise, the agent with an NN brain was also trained for a total of 150 minutes with a total step count of 3,000,000 with the training process entirely lasting for 300 minutes (5 hours) long with 6,000,000 agent steps.

For the agent's respawn locations, 3 coordinates were used, as (7.3, 0.05, -30), (20, 0.05, 11.5), (24, 0.05, -18). With each one associated with a type of 'difficulty level' in its starting positions, (7.3, 0.05, -30) is marked as "Easy", (20, 0.05, 11.5) categorized as "Medium", and (24, 0.05, -18) marked "Difficult" (Figure 4.17). The levels were opted according to the kinds of routes that were able to be taken, their perceptibility, and their distance from the target. In addition, all levels included training the agent without an NN brain model and then with ("No Brain" and "Brain").

Figure 4.17: Agent's Difficulty Level Locations, Easy, Medium, and Difficult. (Created by Author).

4.4 EVALUATION METHOD OF SIMULATION

The evaluation method of this simulation study was directed in an observational form, of the paths that the agent took in its explorations to find the target. The observations made were by directly watching the agent during the training sessions as it navigated the digital environment. Every training session was observed in detail, while recording the routes taken in a diagrammatic format that represents the digital environment, with the starting difficulty level locations present to track the patterns efficiently. A group of different routes taken was created as "Primary", "Secondary", and "Random". The categorization of the types of routes were as a result of the detailed observations made. "Primary" routes are the ones taken most often by the agent in its explorational findings of the target, "Secondary" routes are the second most taken by the agent, while "Random" routes were the ones that did not match the agent's repetitive path taking and were randomly taken at random intervals.

5. RESULTS

In this chapter, the results and findings of the simulation study examining the impact of curiosity as a motivational factor on a cyborg explorer are presented with the aim of exploring how an AI agent's curiosity impacts their exploratory and behavioural patterns in a simulated environment that mimics that of an Istanbul setting. Using ABM, the study discovered analogies that brings this relationship into focus with this chapter detailing the findings from a simulation method using Unity's ML-Agents game engine toolkit setup that engages an agent in explorations to find a target. Referred to as the 'cyborg explorer' because of its flâneur-like thinking and manner, the agent is taught by RL with its task being navigating a digital rendition of Istanbul's urban environment specifically focused on the Sultan Ahmet Camii (Blue Mosque) neighbourhood, with the Sultan Ahmet Camii landmark as its target.

In the "Easy" level, the training panels showed that while the "No Brain" displayed progress in collecting rewards after 150,000 steps, the "Brain" did after 100,000 (Figure 4.18) (Figure 4.19). The "Brain" agent was able to interact with the target more, as it memorized the routes that lead it to more rewards.

```
Cyborg's Behaviour. Step: 50000. Time Elapsed: 102.194 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 100000. Time Elapsed: 195.741 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 150000. Time Elapsed: 287.273 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 200000. Time Elapsed: 375.909 s. Mean Reward: -4.901. Std of Reward: 1.472.
Cyborg's Behaviour. Step: 250000. Time Elapsed: 463.349 s. Mean Reward: -3.121. Std of Reward: 6.149.
Cyborg's Behaviour. Step: 300000. Time Elapsed: 551.553 s. Mean Reward: 2.849. Std of Reward: 10.539.
Cyborg's Behaviour. Step: 350000. Time Elapsed: 639.137 s. Mean Reward: 5.214. Std of Reward: 10.972.
Cyborg's Behaviour. Step: 400000. Time Elapsed: 726.938 s. Mean Reward: 6.284. Std of Reward: 10.996.
Cyborg's Behaviour. Step: 450000. Time Elapsed: 815.727 s. Mean Reward: 8.460. Std of Reward: 10.721.
Cyborg's Behaviour. Step: 500000. Time Elapsed: 904.585 s. Mean Reward: 10.734. Std of Reward: 9.929.
```

Figure 5.1: "Easy" Level Training of "No Brain" Agent. (Created by Author).

```
Cyborg's Behaviour. Step: 50000. Time Elapsed: 163.877 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 100000. Time Elapsed: 259.691 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 150000. Time Elapsed: 349.378 s. Mean Reward: -3.952. Std of Reward: 4.685.
Cyborg's Behaviour. Step: 200000. Time Elapsed: 438.074 s. Mean Reward: 2.875. Std of Reward: 10.547.
Cyborg's Behaviour. Step: 250000. Time Elapsed: 525.771 s. Mean Reward: 5.834. Std of Reward: 10.999.
Cyborg's Behaviour. Step: 300000. Time Elapsed: 614.380 s. Mean Reward: 7.673. Std of Reward: 10.872.
Cyborg's Behaviour. Step: 350000. Time Elapsed: 703.317 s. Mean Reward: 10.661. Std of Reward: 9.964.
Cyborg's Behaviour. Step: 400000. Time Elapsed: 793.423 s. Mean Reward: 11.985. Std of Reward: 9.229.
Cyborg's Behaviour. Step: 450000. Time Elapsed: 883.724 s. Mean Reward: 12.221. Std of Reward: 9.072.
Cyborg's Behaviour. Step: 500000. Time Elapsed: 973.994 s. Mean Reward: 12.278. Std of Reward: 9.033.
```

Figure 5.2: "Easy" Level with NN Brain Model Training Results for "Brain" Agent. (Created by Author).

In the "Medium" level, the training panels displayed that both the "No Brain" "Brain" displayed progress in collecting rewards after 100,000 steps (Figure 4.20) (Figure 4.21). The

"Brain" agent was faster in its interactions with the target because it showed a more secure reward accumulation.

```
Cyborg's Behaviour. Step: 50000. Time Elapsed: 99.479 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 100000. Time Elapsed: 192.352 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 150000. Time Elapsed: 282.618 s. Mean Reward: -4.867. Std of Reward: 1.669.
Cyborg's Behaviour. Step: 200000. Time Elapsed: 372.028 s. Mean Reward: -0.196. Std of Reward: 8.821.
Cyborg's Behaviour. Step: 250000. Time Elapsed: 461.402 s. Mean Reward: 8.168. Std of Reward: 10.155.
Cyborg's Behaviour. Step: 300000. Time Elapsed: 551.501 s. Mean Reward: 10.014. Std of Reward: 9.480.
Cyborg's Behaviour. Step: 350000. Time Elapsed: 641.751 s. Mean Reward: 11.901. Std of Reward: 8.323.
Cyborg's Behaviour. Step: 400000. Time Elapsed: 730.698 s. Mean Reward: 13.390. Std of Reward: 6.928.
Cyborg's Behaviour. Step: 450000. Time Elapsed: 820.046 s. Mean Reward: 13.401. Std of Reward: 6.915.
Cyborg's Behaviour. Step: 500000. Time Elapsed: 909.951 s. Mean Reward: 14.192. Std of Reward: 5.891.
```

Figure 5.3: "Medium" Level Training of "No Brain" Agent. (Created by Author).

```
Cyborg's Behaviour. Step: 50000. Time Elapsed: 157.924 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 100000. Time Elapsed: 302.002 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 150000. Time Elapsed: 443.856 s. Mean Reward: -4.094. Std of Reward: 4.372.
Cyborg's Behaviour. Step: 200000. Time Elapsed: 585.297 s. Mean Reward: 1.382. Std of Reward: 9.984.
Cyborg's Behaviour. Step: 250000. Time Elapsed: 726.280 s. Mean Reward: 5.716. Std of Reward: 10.996.
Cyborg's Behaviour. Step: 300000. Time Elapsed: 867.879 s. Mean Reward: 7.614. Std of Reward: 10.881.
Cyborg's Behaviour. Step: 350000. Time Elapsed: 1009.610 s. Mean Reward: 11.415. Std of Reward: 9.575
Cyborg's Behaviour. Step: 400000. Time Elapsed: 1150.713 s. Mean Reward: 10.911. Std of Reward: 9.843
Cyborg's Behaviour. Step: 450000. Time Elapsed: 1293.943 s. Mean Reward: 13.023. Std of Reward: 8.466
Cyborg's Behaviour. Step: 500000. Time Elapsed: 1436.034 s. Mean Reward: 13.392. Std of Reward: 8.146
```

Figure 5.4: "Medium" Level with NN Brain Model Training Results for "Brain" Agent. (Created by Author).

In the "Difficult" level, the training panels exhibited that both the "No Brain" "Brain" progressed in getting rewards after 150,000 steps (Figure 4.22) (Figure 4.23). The "No Brain" agent obtained more rewards despite the fact that it did not have an NN to help guide its decisions.

```
Cyborg's Behaviour. Step: 50000. Time Elapsed: 161.196 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 100000. Time Elapsed: 307.051 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 150000. Time Elapsed: 452.526 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 200000. Time Elapsed: 593.761 s. Mean Reward: -4.946. Std of Reward: 1.086.
Cyborg's Behaviour. Step: 250000. Time Elapsed: 735.012 s. Mean Reward: -3.619. Std of Reward: 5.336.
Cyborg's Behaviour. Step: 300000. Time Elapsed: 875.321 s. Mean Reward: -1.676. Std of Reward: 7.879.
Cyborg's Behaviour. Step: 350000. Time Elapsed: 1014.879 s. Mean Reward: 2.894. Std of Reward: 10.552
Cyborg's Behaviour. Step: 400000. Time Elapsed: 1153.768 s. Mean Reward: 5.560. Std of Reward: 10.991
Cyborg's Behaviour. Step: 450000. Time Elapsed: 1294.581 s. Mean Reward: 5.841. Std of Reward: 10.999
Cyborg's Behaviour. Step: 500000. Time Elapsed: 1435.234 s. Mean Reward: 8.484. Std of Reward: 10.716
```

Figure 5.5: "Difficult" Level Training of "No Brain" Agent. (Created by Author).

```
Cyborg's Behaviour. Step: 50000. Time Elapsed: 100.889 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 100000. Time Elapsed: 194.380 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 150000. Time Elapsed: 284.785 s. Mean Reward: -5.000. Std of Reward: 0.000.
Cyborg's Behaviour. Step: 200000. Time Elapsed: 373.752 s. Mean Reward: -4.878. Std of Reward: 1.635.
Cyborg's Behaviour. Step: 250000. Time Elapsed: 460.225 s. Mean Reward: -0.351. Std of Reward: 8.982.
Cyborg's Behaviour. Step: 300000. Time Elapsed: 547.947 s. Mean Reward: 5.054. Std of Reward: 10.959.
Cyborg's Behaviour. Step: 350000. Time Elapsed: 635.591 s. Mean Reward: 8.375. Std of Reward: 10.740.
Cyborg's Behaviour. Step: 400000. Time Elapsed: 723.669 s. Mean Reward: 9.277. Std of Reward: 10.500.
Cyborg's Behaviour. Step: 450000. Time Elapsed: 811.344 s. Mean Reward: 10.607. Std of Reward: 9.989.
Cyborg's Behaviour. Step: 500000. Time Elapsed: 900.793 s. Mean Reward: 11.751. Std of Reward: 9.377.
```

Figure 5.6: "Difficult" Level with NN Brain Model Training Results for "Brain" Agent. (Created by Author).

Studying the cyborg explorer's behaviour, through an observational method, displays its navigational trends, as "Primary", "Secondary", and "Random" routes taken (Figure 5.1). The agent's paths were recorded in the form of a diagrammatic representation of the digital map of the environment, with the starting difficulty level locations present to track the path patterns efficiently.

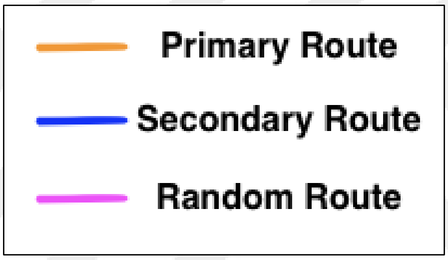

Figure 5.7: The Agent's Behavioural Patterns are Categorized Into 3 Types. (Created by Author).

The "Easy" level's behavioural pattern results exhibit similar movements between both types of agents as they took the same primary route, took random routes at random times with the "No Brain" agent taking more varied random routes (Figure 5.2). The interpretations made regarding this behaviour entail that both agent types displayed similar movement patterns, primarily favouring a shared primary route, with some sporadic deviations into random routes. Due to not having much information about the environment, the "No Brain" was more curiously motivated to get to the target because it explored more random paths. The "Brain" used its NN, which remembered the routes that led it to the target, to take the primary routes to receive rewards and was not as motivated by curiosity to explore many paths, sometimes it would but it was a random instance. The cyborg explorer led by its motivational curiosity, showed behaviours alike to the flâneur as it navigated to different locations and explored new features of the environment while having a certain preference for specific paths.

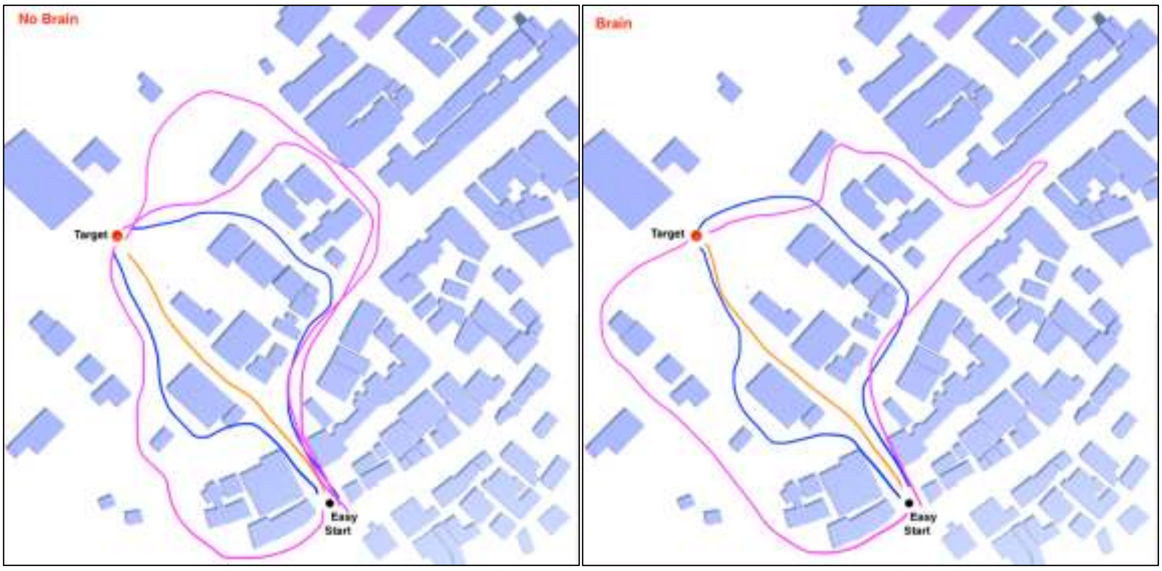

Figure 5.8: “Easy” Level Training Evaluation of “No Brain” and “Brain” Agent. (Created by Author).

The “Medium” level’s behavioural pattern results unveil different movements between both types of agents as they took differing primary routes, with the “No Brain” agent taking more wide-ranging random paths and the “Brain” preferring shorter ones (Figure 5.3). The interpretations of this result are that as both agents were finding the target, they were more motivated by curiosity to explore different routes that could lead them to it, but as they ‘learned’ to find it, mostly the “No Brain” agent continued to explore different paths while the “Brain”, due to familiarity and past experiences stored in its NN, continued to take the same paths that lead straight to the target. Linked to the cyborg explorer approach and its motivational curiosity, it means that it was influenced by curiosity to find the target as it began to get familiar with the layout because of repetitively reaching it. The cyborg explorer displayed exploratory behavioural patterns that are similar to the flâneur as it discovered different locations and investigated new features of the environment, with a unique preference for certain routes.

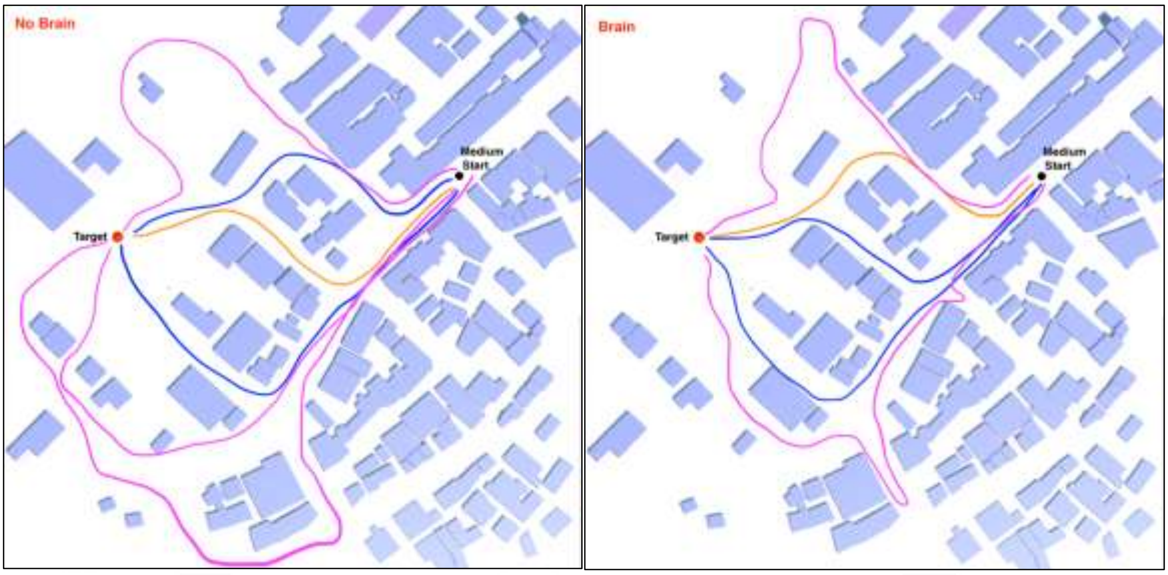

Figure 5.9: “Medium” Level Training Evaluation of “No Brain” and “Brain” Agent. (Created by Author).

The “Difficult” level’s explorational trends express similar movements between both types as they took the same primary and secondary routes, similar random routes at random times with the “Brain” agent taking slightly more diverse random paths (Figure 5.4). The interpretations made in terms of this behaviour are that both agents had the same preferences in paths taken, with curiosity having an effect on the type of path they often took. In associations with the cyborg explorer, it was curiously motivated to explore as many paths as it could to reach the target, and when it did, it still continued to explore different routes but had certain path preferences that showed repeated patterns in its behaviours. The cyborg explorer exhibited exploratory behavioural trends that are similar to the flâneur in curiously navigating different locations and finding new features of the environment, with some preference for certain ways to the target.

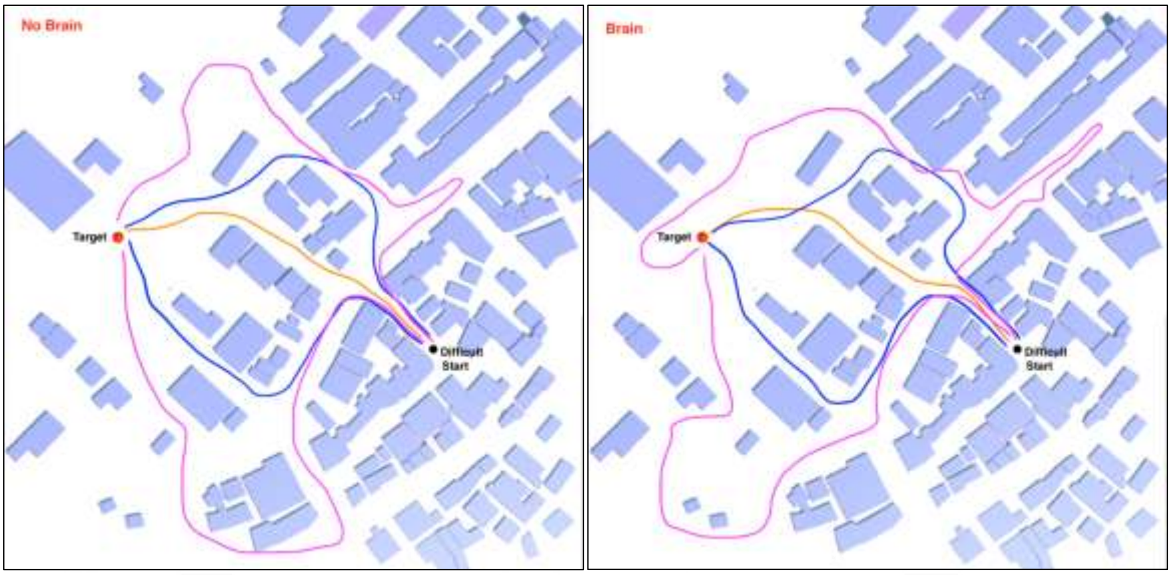

Figure 5.10: “Difficult” Level Training Evaluation of “No Brain” and “Brain” Agent. (Created by Author).

Through the observational interpretations of training sessions across different difficulty levels, distinctive behavioural patterns surfaced with a link to the cyborg explorer and how it behaves when motivated by curiosity, presenting adequate comprehensions into their navigation tactics and tendencies. The cyborg explorer kept taking the paths it recognized and experienced more, forming a schema, being a cognitive construct affected by experience, that shaped its behaviour and lead it to prefer certain routes. In addition, with the cyborg explorer missing an NN brain model, when it found a moderately easy or straightforward path that led it to the target, it continued to take that path and ignored challenging routes. From the cyborg explorer’s curious perception of the city, it became familiar with certain routes through repetitive exposure to the environment, this familiarity developing from its repeated encounters with the environment which allowed it to cultivate a basic understanding of the layout. Without an NN brain model present, the cyborg explorer’s familiarity is not the actual result of true learning in the sense of adapting behaviours, based on rewards or feedback, and instead it unintentionally “learned” certain paths simply by repeated interaction with them.

However, when an NN brain model is present and similar to the way human beings make decisions, the cyborg explorer “learned” to associate environmental conditions with actions that lead it to desirable results (rewards) and avoid actions that lead it to undesirable consequences (punishments). It also learned to recognize efficient paths to reach the target due to its NN brain continuously regulating its parameters that develop its decision-making abilities. In addition, it learned the target's (Sultan Ahmet landmark) value as a reference point and created a mind map where it memorized the paths leading to the target landmark. Over time, the cyborg explorer became familiar with the environment's layout, driven by its curiosity to navigate and interact, its behaviours became more competent in collecting rewards while adapting its approaches based on past experiences. It performed forms of curiosity-driven explorations, with influence and encouragement from its adaptive schema in its NN, revealing familiar patterns that have been linked with positive results in the past.

5.1 DISCUSSION

This discussion identifies the limitations of this study that may affect the practicality that this research can provide for architects, even as the study produced comprehensions into how individuals in real-world settings may behave when influenced by curiosity, this part of the chapter details the limitations and constraints that were faced with. Firstly, this simulation study used a very small and simple sample size of an Istanbul neighbourhood environment that may not fully encompass the trueness of its urban features. When importing the site map into the Unity application, it was scaled down to a simple dimension due to the original size being too complex for the agent and also due to computational rendering power. Secondly, the environment was simplified by removing its 3D features with the flattening of the site plane and removing the natural terrains of the digital environment as well as the exclusion of streets, which resulted in a completely flat simulation environment, which may not fully portray the real natural essence of Istanbul's rich topography and layers. Lastly, it is important to acknowledge that real-life urban environments are multi-layered in their hidden details and complex formulations and so simulation studies can only try to imitate the real thing, and even when it can do so, it still lacks the realness aspect.

6. CONCLUSION

Using RL algorithms in ML-Agents mimics real-world situations where cyborg explorers, agents with the concept of having a flâneur outlook on exploration, can use their curiosity about their environments to be motivated in navigating and learning about it, which in turn gives architects the chances to learn about real-world behavioural patterns through the perceptions of a cyborg explorer. Because of unhinged changes in the preferences of users and social and environmental aspects, with methodologies driven by RL, that are inspired by feedback mechanisms theorized in cybernetics, architects can use RL in Unity's ML-Agents, featuring agents that have the ability to learn from their positive and negative actions in environments, in a similar way to how human beings would, in simulation studies that mimic real-world scenarios that are too complex to study using conventional methods, by studying curiosity-driven behavioural patterns and preferences in agents to find design solutions that apply to real-life.

This thesis' simulation study used an RL algorithm using the Unity's ML-Agents toolkit to ultimately examine the behavioural effects that motivational curiosity has on a cyborg explorer's exploration patterns in an urban layout. It studied the impacts of curiosity as a motivational drive on a cyborg explorer's behavioural tendencies in one of Istanbul's urban settings, focused on the Sultan Ahmet Camii's (Blue Mosque) neighbourhood with its target being the Sultan Ahmet Camii landmark.

The exploration of motivated curiosity in the cyborg explorer revealed distinctive patterns across 3 types of difficulty starting locations in the environment. The cyborg explorer displayed curiosity motivated behaviours and exhibited diverse forms of explorations through the learning processes catalysed by RL, amplifying it in the light of a curiosity motivated flâneur explorer.

The thesis targeted to answer these questions: What is curiosity's motivational influence on the behaviour of a cyborg explorer? Are there any chances for architects to actively engage and interact with the cyborg explorer throughout the RL cycle? Alternatively, what can an architect gain from such experience in relations to their design solutions?

According to these questions, the results of this study strengthen the multi-layered relationship between motivational curiosity and behavioural patterns in finding that:

a. Curiosity does certainly have an effect on the cyborg explorer as it motivated it to explore different areas of the environment, developing unique path preferences and using the Sultan Ahmet landmark target as its point of reference to create a comprehensive mental map of the environment.

b. During the RL learning process, architects can form interactions with the cyborg explorer through making modifications and controlling the C# script reward design, where they can give instructions, ultimately communicating with it.

c. From such experiences, architects can use the cyborg explore approach to generate adaptive design solutions by testing it in different simulated urban scenarios that imitate real-life agendas, to explore behavioural patterns and user preferences.

With these insights and with this approach, architects can learn from curiosity-based exploration whereby closely observing and interpreting a cyborg explorer's actions, they can opt for personalized user-centric design solutions with the objective of designing for individuals' unique preferences. In addition, this approach informs urban planning principles in landmark place finding as an attempt to improve the navigability of an urban space. Also, the cyborg explorer's ability to create a mental map of an urban layout supports the need for easily recalled urban spaces, allowing users to easily explore those spaces without encountering confusion.

Even as this study resulted in comprehensions regarding how users in real urban settings may behave as they are motivated by curiosity, it is necessary to also consider the limitations that may overall decrease the applicability and generalizability of the interpretations made. Because the simulation study used a small sample dimension of a neighbourhood in Istanbul, it may not entirely fulfil the true urban features it represents. Also, among other modifications, with the digital site size simplified by removing the natural terrains which resulted in a completely flat surface, so it may not depict the real essence of Istanbul's rich topography. In addition, although simulated environments are ideal at mimicking real-life settings, they cannot capsulate their full essence as real-world environments are delicately detailed and complex, and so the cyborg explorer behaviours can only mirror that of human beings to a certain extent.

In conclusion, this research dissects the details that come with curiosity in forming the cyborg explorer's behaviours by looking at how curiosity's motivational effects and directs it, through ML-Agents, in the way that it thinks and takes actions, where aspects in relation to how people in the real-world take certain streets in a neighbourhood and why they are curiously motivated to take them are discovered. These understandings can conceptually and realistically enrich urban designs in using RL algorithms to ultimately learn more about how a cyborg explorer curiously behaves based on consequences and can explore spaces to display certain behavioural patterns in an urban setting. The prospects that this study present architects with the opportunities to work directly with a cyborg explorer in using it as a design instrument to better their urban designs by challenging it in simulated setups that replicate real-life outlines to discover information that they may not be able to achieve with methods that are currently being used today. Whereby carefully observing and interpreting a cyborg explorer's movements, architects have the openings to get experiences from this curiosity-based exploration where they can design for adaptive design solutions that are user-centric with the intention of designing for users' exclusive habits.

REFERENCES

Akpan, B. (2020). Classical and operant conditioning—Ivan Pavlov; Burrhus Skinner. Springer Texts in Education, 1(1), 71–84. https://doi.org/10.1007/978-3-030-43620-9_6

Allen, R. (1978). Retrieved November 15, 2024, from https://rebeccaallen.com/projects/aspen-movie-map

Anable, A. (2011). The architecture machine group's Aspen movie map. Television & New Media, 13(6), 498–519. https://doi.org/10.1177/1527476411423673

Anderson, S. (2002). Quasi-autonomy in architecture: the search for an "in-between." Perspecta, 33, 30-37. https://doi.org/10.2307/1567294

Archigram. (1964). Portfolio. Retrieved November 13, 2024, from http://archigram.net/portfolio.html

Architecture (CCA). (1960). Cedric Price fonds. Retrieved November 20, 2024, from https://www.cca.qc.ca/en/archives/380477/cedric-price-fonds

Bartlett, F. C. (1932). Remembering: a study in experimental and social psychology. Cambridge University Press.

Berlyn, P. & Fowler, C. (1851). The crystal palace: its architectural history and constructing marvels. James Gilbert Publications.

Boutin, A. (2012). Rethinking the Flâneur: Flânerie and the Senses. Dix-Neuf, 16(2), 124–132. https://doi.org/10.1179/dix.2012.16.2.01

Burda, Y., Edwards, H., Pathak, D., Storkey, A., Darrell, T., & Efros, A. A. (2018). Large-scale study of curiosity-driven learning. https://doi.org/10.48550/arXiv.1808.04355

Clynes, M. E. & Kline, N. S. (1960). Cyborgs and space. Astronautics, The Cyborg Handbook, 14(9), 26-27, Routledge. Retrieved November 20, 2024, from https://web.mit.edu/digitalapollo/Documents/Chapter1/cyborgs.pdf

Cook, P. (1964). Plug-in city. In Amazing Archigram/ Zoom (4th Issue), (Unrealized Project).

Crompton, D. (1964). Computer city. In Amazing Archigram/ Zoom (4th Issue), (Unrealized Project).

Crooks, A. T. & Heppenstall, A. J. (2011). Introduction to agent-based modelling. In Agent-Based Models of Geographical Systems, 85-105, Springer Netherlands. https://doi.org/10.1007/978-90-481-8927-4_5

Dai, A. (2022). Co-creation: space reconfiguration by architect & agent simulation based machine learning-Issuu. Retrieved December 18, 2024, from https://issuu.com/annidai0202/docs/_book_space-co-creation

Dai, A. (2022, June). Co-creation: space reconfiguration by architect and agent simulation-based machine learning. In the International Conference on Computational Design and Robotic Fabrication, 304-313. Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-19-8637-6_27

Deci, E. L., & Ryan, R. M. (2008). Self-determination theory: A macro-theory of human motivation, development, and health. Canadian Psychology, 49(3), 182–185. https://doi.org/10.1037/a0012801

Engelbrecht, D. (2023). Unity ml-agents. In Introduction to Unity ML-Agents: Understand the Interplay of Neural Networks and Simulation Space Using the Unity ML-Agents Package, 87-135, Berkely, CA: Apress. https://doi.org/10.1007/978-1-4842-8998-3_6

Fyfe, C. A. (2005). Artificial Neural Networks. In Do Smart Adaptive Systems Exist? 57–79, Heidelberg: Springer. https://doi.org/10.1007/3-540-32374-0_4

Glanville, R. (2002). Second order cybernetics. In Systems Science and Cybernetics, Encyclopaedia of Life Support (EOLSS), UNESCO, 7.

Glanville, R. (2014). How design and cybernetics reflect each other. In: Proceedings of RSD3, Third Symposium of Relating Systems Thinking to Design. Retrieved December 5, 2024, from https://openresearch.ocadu.ca/id/eprint/2053/1/Glanville_Reflect_2014.pdf

Groisser, L. B. & Negroponte, N. (1973). Computer aids to participatory architecture, MIT.

Halmos, P. R. (1973). The legend of John von Neumann. The American Mathematical Monthly, 80(4), 382-394.

Halpern, O. (2015). Inhuman vision. In Media-N, Journal of the New Media Caucus, Art & Infrastructures: Information. December 19, 2024, from https://median.newmediacaucus.org/art-infrastructures-information/inhuman-vision/

Haraway, D. J. (1991). A cyborg manifesto: science, technology, and socialist-feminism in the late twentieth century. In Simians, Cyborgs and Women: The Reinvention of Nature, 149-181, Routledge.

Haugeland, J. (1989). Artificial intelligence: the very idea, The MIT Press.

Hazel Hahn, H. (2012). The flâneur, the tourist, the global flâneur, and magazine reading as Flânerie. Dix-Neuf, 16(2), 193–210. https://doi.org/10.1179/12z.00000000017

Herron, R. (1964). Walking city. In Amazing Archigram/ Zoom (4th Issue), (Unrealized Project).

Huang, C. H. (2021, March). Reinforcement learning for architectural design-build-opportunity of machine learning in a material-informed circular design strategy. In Proceedings of the 26th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA), (Vol.1), 171-180.

Intrator, O. & Intrator, N. (2001). Interpreting neural-network results: a simulation study. Computational Statistics & Data Analysis, 37(3), 373-393. https://doi.org/10.1016/S0167-9473(01)00016-0

Janssen, M. A. (2005). Agent-based modelling. Modelling in Ecological Economics, 155(1), 172-181.

Juliani, A., Berges, V. P., Teng, E., Cohen, A., Harper, J., Elion, C., Goy, C., Gao Y., Henry, H., Mattar, M., & Lange, D. (2018). Unity: a general platform for intelligent agents, arXiv (Cornell University). https://doi.org/10.48550/arxiv.1809.02627

Kaelbling, L. P., Littman, M. L., Moore, A. W. (1996). Reinforcement learning: a survey. Journal of Artificial Intelligence Research, 4, 237-285. https://doi.org/10.1613/jair.301

Katz, K., Burnham, J. (Ed.) (1970). SEEK 1969-70. In Software: Information Technology: Its New Meaning for Art, 23.

Krogh, A. (2008). What are artificial neural networks? Nature Biotechnology, 26(2), 195–197. https://doi.org/10.1038/nbt1386

Lanham, M. (2018). Learn unity ml-agents—fundamentals of unity machine learning: incorporate new powerful ml algorithms such as deep reinforcement learning for games, Packt Publishing Ltd.

Lippman, A. (1980). Movie-maps: an application of the optical videodisc to computer graphics. ACM SIGGRAPH Computer Graphics, 14(3), 32–42. https://doi.org/10.1145/965105.807465

Littlewood, J. (1968). The fun palace: a laboratory of fun. In the Drama Review: TDR, 12(3), 127-134.

Lobsinger, M. L. (2000). Cybernetic theory and the architecture of performance: Cedric Price's fun palace. Anxious Modernisms: Experimentation in Post-war Architectural Culture, Cambridge, The MIT Press, 119-139.

London Developers Toolkit. (2020). You+Pea. Retrieved November 20, 2024, from https://www.youandpea.com/london-developers-toolkit

Lynch, K. (1964). The image of the city. MIT Press.

Macal, C. M., & North, M. J. (2009). Agent-based modeling and simulation. Proceedings of the 2009 Winter Simulation Conference (WSC). https://doi.org/10.1109/wsc.2009.5429318

Macal, C. M. (2016). Everything you need to know about agent-based modelling and simulation. Journal of Simulation, 10(2), 144–156. https://doi.org/10.1057/jos.2016.7

Mathews, J. S. (2003). An architecture for the new Britain: the social vision of Cedric Price's fun palace and Potteries Thinkbelt, Columbia University, (Doctoral dissertation), Retrieved from ProQuest Dissertations and Theses - (AAT 3074308).

Mathews, S. (2006). The fun palace as virtual architecture: Cedric Price and the practices of indeterminacy. Journal of Architectural Education, 59(3), 39–48. https://doi.org/10.1111/j.1531-314x.2006.00032.x

Maria, A. (1997, December). Introduction to modelling and simulation. In Proceedings of the 29th Conference on Winter Simulation, 7-13.

McCarthy, J. (2007). What is artificial intelligence, Stanford University.

MIT. (1981). Aspen Movie Map (Raw Laserdisc). Internet Archive. Retrieved January 12, 2024, from https://archive.org/details/ASPEN4

Mitchell, T.M. (1997). Artificial neural networks. Machine Learning, 45(1), 81-127.

Naimark, M. (1982), The optical videodisc and new media forms. In Video, 80, San Francisco International Video Festival, 24- 25.

Negroponte, N. & Groisser, L. (1967). Special problems in computer-aided design.

Negroponte, N. (1969). Toward a theory of architecture machines. Journal of Architectural Education, 23(2), 9–12. https://doi.org/10.1080/00472239.1969.11102296

Negroponte, N. (1970). The architecture machine: toward a more human environment. Cambridge; The MIT Press.

Negroponte, N. (1975). Soft Architecture Machines. Cambridge; The MIT Press.

Pangaro, P. (1996). Dandy of cybernetics. The Guardian.

Pangaro, P. & Mcleish, T. J. (2018). Colloquy of Mobiles 2018 Project. In Symposium on Cybernetic Serendipity Reimagined. Retrieved November 15, 2024, from https://www.pangaro.com/published/Pangaro-McLeish_Colloquy-2018_AISB-2018_distro.pdf

Pask, G. (1964). Proposals for a Cybernetic Theatre. In Theatre Workshop & System Research, London.

Pask, G. (1969). The architectural relevance of cybernetics, Architectural Design, 7(6) 494-496, John Wiley & Sons Ltd.

Pask, G. (1971). A comment, a case history and a plan. In Cybernetics, Art and Ideas. 76-99, Studio Vista.

Pask, G. (1975). Conversation. cognition and learning: a cybernetic theory and methodology. Elsevier.

Pask, G. (1975). Introduction to aspects of machine intelligence, In N. Negroponte, Soft Architecture Machines, Cambridge, MA: The MIT Press.

Pask, G. (1976). Conversation theory: applications in education and epistemology, Elsevier.

Pask, G. (1980). An initial essay: towards a unification of architectural theories, Gordon Pask Archive, Department of Contemporary History, University of Vienna, 44(4), 2-8.

Paternina-Arboleda, C. D., Montoya-Torres, J. R., & Fabresgas-Ariza, A. (2008). Simulation-optimization using a reinforcement learning approach. In 2008 Winter Simulation Conference, 1376-1383. https://doi.org/10.1109/wsc.2008.4736213

Pearson, L. C. (2020). A machine for playing in: exploring the videogame as a medium for architectural design. Design Studies, 66, 114–143. https://doi.org/10.1016/j.destud.2019.11.005

Price, C. (2003). The square book. Wiley-Academy.

Roberts, L. G. (1963). Machine perception of three-dimensional solids. Harvard Engineering Journal, 33(1), 87-110.

Rosen, M. (1968). Media Art Net | Pask, Gordon: The Colloquy of Mobiles. Medienkunstnetz.de. Retrieved December 20, 2024, from http://medienkunstnetz.de./works/colloquy-of-mobiles/

Rosen, M. (1969). Gordon Pask The Colloquy of Mobiles 1969. ZKM. Retrieved December 20, 2024, from https://zkm.de/en/artwork/the-colloquy-of-mobiles

Ryan, R. M. & Deci, E. L. (2002). Overview of self-determination theory: an organismic dialectical perspective. Handbook of Self-Determination Research, 2(3-33), 36.

Sadler, S. (2005). Archigram: architecture without architecture. Cambridge: MIT Press.

Schwarzplan.eu - International city site plans as CAD maps. (2022). Schwarzplan.eu. Retrieved January 4, 2024, from https://schwarzplan.eu/en/

Selwyn, N. (2008). A circus of ideas: revisiting Archigram's visions for education and architecture in the information age, London Knowledge Lab, United Kingdom. https://doi.org/10.31235/osf.io/eh4mx

Silvia, P. J. (2012). Curiosity and motivation. The Oxford Handbook of Human Motivation, 157-166.

Skinner, B. F. (1938). The behaviour of organisms: an experimental analysis. D. Appleton-Century Company.

Steenson, M. W. (2014). Architectures of information: Christopher Alexander, Cedric Price, and Nicholas Negroponte & MIT's architecture machine group. (Doctoral dissertation), Retrieved from ProQuest Dissertations and Theses - (AAT 3615100).

Sutherland, I. E. (1963). Sketchpad: a man-machine graphical communication system. In Proceedings of Spring Joint Computer Conference, 329-346.

Sutton, R. S. (1992). Introduction: the challenge of reinforcement learning. In Reinforcement Learning, Boston, Springer, 1-3, Boston, Springer.

Sutton, R. S. & Barto, A. G. (1999). Reinforcement learning. Journal of Cognitive Neuroscience, 11(1), 126-134.

Turner, O. C., Aeffner, F., Bangari, D. S., High, W., Knight, B., Forest, T., Cossic, B., Himmel, L. E., Rudmann, D. G., Bawa, B., Muthuswamy, A., Aina, O. H., Edmondson, E. F., Saravanan, C., Brown, D. L., Sing, T., & Sebastian, M. M. (2019). Society of toxicologic pathology digital pathology and image analysis special interest group article*: opinion on the application of artificial intelligence and machine learning to digital toxicologic pathology. Toxicologic Pathology, 48(2), 277–294. https://doi.org/10.1177/0192623319881401

Weiss, R. S. (1968). Planning for diversity and choice. Social Science & Medicine (1967), In Proceedings of the Conference 'Inventing the Future Environment, 2(4), 513–514, MIT Press, Cambridge, Mass. https://doi.org/10.1016/0037-7856(68)90071-1

Weizenbaum, J. (1966). ELIZA-a computer program for the study of natural language communication between man and machine. Communications of the ACM, 9(1), 36-45.

Wheeler, S. M. (2008). The evolution of built landscapes in metropolitan regions. Journal of Planning Education and Research, 27(4), 400-416.

Wiener, N. (1948). Cybernetics: or control and communication in the animal and the machine. The MIT Press, Cambridge.

Wiener, N. (1950). The human use of human beings. The MIT Press, Cambridge.

Von Foerster, H. (ed.). (1995). Cybernetics of cybernetics: or the control of control and the communication of communication. Future Systems Inc.