Republic of Iraq
Ministry of Higher Education and Scientific Research
University of Al-Qadisiyah
College of Computer Science and Information Technology
Department of Computer Science

Designing and Implementation Testbed for Intelligent Agent Architecture

A Thesis

Submitted to the Council of the College of Computer Science and Information Technology at the University of Al-Qadisiyah in Partial Fulfillment of the Requirements for the Degree of Master in Computer Science.

By

Ahmed Majeed Kareem Al-Baghdadi

Supervised by

Prof. Dr. Ali Obied Sharad

2021 A.D 1443 A.H

بسم الله الرحمن الرحيم

إِنَّمَا يَخْشَى اللَّهَ مِنْ عِبَادِهِ الْعُلَمَاءُ ۗ إِنَّ اللَّهَ عَزِيزٌ غَفُورٌ

صدق الله العلي العظيم

(سورة فاطر: الآية 28)

Supervisor Certificate

*I certify that thesis entitled "***Designing and Implementation Testbed for Intelligent Agent Architecture***" is prepared and written under my supervision at the department of Computer Science / College of Computer Science and Information Technology / University of Al-Qadisiyah as a partial fulfillment of the requirements of the degree of Master in Computer Science.*

Signature:

Supervisor Name**: **Pro. Dr. Ali Obied Sharad

Date: / 12 /2021

Head of Department Certificate

*In view of the available recommendations, I forward the thesis entitled "***Designing and Implementation Testbed for Intelligent Agent Architecture***" for debate by the examination committee.*

Signature:

Head Name: Qusay Omran Mosa

Head of the Department of Computer Science

Date: / 12 / 2021

Certificate of the Examination Committee

*We, the undersigned, certify that (**Ahmed Majeed Kareem**) candidate for the degree of Master in Computer Science has presented this thesis entitled (**Designing and Implementation Testbed for Intelligent Agent Architecture**) for debate examination. The examination committee confirms that this thesis is accepted in form and content and displays a satisfactory knowledge in the field of study based on the candidate demonstration during the debate examination held on: **29-November-2021**.*

Signature:

Name: Ali Hussein Hasan

Title: Prof. Dr.

Date: / 12 / 2021

(Chairman)

Signature:

Name: Rana Jumaa Surayh Al-Janabi

Title: Assist Prof. Dr.

Date: / 12 / 2021

(Member)

Signature:

Name: Salwa Shakir Baawi

Title: Dr.

Date: / 12 / 2021

(Member)

Signature:

Name: Ali Obied Sharad

Title: Prof. Dr.

Date: / 12 / 2021

(Member)

Signature:

Name: Dhiah Al-Shammary

Title: Assist Prof. Dr.

Date: / 12 /2021

(Dean of College of Computer Science and Information Technology)

Dedication

I would like to dedicate this work:

To God is the ruler of all affairs, the Creator of the pen and the light, the Most Gracious, the Most Merciful. And to the master of beings and the light of God in the earth and the heavens, the savior of humanity from the darkness of ignorance to the tolerant Sharia. And to his faithful family, and at their head, the master of the guardians, the Commander of the Faithful, Ali bin Abi Talib. And to everyone who taught me, guided me, advised me, and took my hand on the path of knowledge and science And to those who bore my hardships and were patient with me so beautifully that I could do this work, asking God to grant me and all my brother's success and payment.

Ahmed

Acknowledgments

My God, I became smaller when your gifts increased my gratitude, your blessings made me loose from the lights of faith, and you struck me with a brilliance of blessings of glory, and you made me with necklaces that do not dissolve, and you encircled me with collars that do not droop.

All thanks and appreciation to my supervisor, Professor Ali Obeid, for his efforts with me, as he is directing, listening, and discussing, as he builds knowledge in my mind little by little without getting tired or tireless.

Thanks and appreciation to the Department of Computer Science and Information Technology, and especially my dear teachers.

And to my family, friends, and everyone who helped me a lot or a little to complete my literature.

Abstract

The most famous characteristic of the machine in our modern age is autonomy. This autonomous in the conditions of a static and dynamic environment requires building models, adopting statistical hypotheses, and delving a lot into probability. The feature of autonomous in static environments is something that has been overcome, but a lot of talk about dynamic environments, changing and multi-events is the great challenge that presents many solutions.

The study that is presented in this thesis includes a system for empirical evaluation of competing for theoretical and architectural proposals, more precisely we built a Gridworld testbed that simulates a real dynamic environment, and the system consists of a main pillar dynamic of environment and an embedded agent. The environment and the agent are determined to a certain degree, which allows the individual to control the characteristics of each of them, so we can experimentally investigate the different behaviors of inference strategies within the meta-level-reasoning by adjusting parameters for the environmental variables and adjusting the parameters of the embedded factor. Our hypothesis is to propose a system that detects the appropriateness of certain strategies within certain environmental changes. The Gridworld testbed has been demonstrated and has been shown to be the available system for evaluating agent architectures. The Gridworld testbed is simplified, is easy to deal with, and qualified to be a platform for testers in that it does not require much effort to deal with. In addition, the Gridworld architecture is not just a simulated environment but has an embedded agent as well.

This study carried out different experiments under different levels of dynamism and commitment and we compare our results with Tileworld results under the same criteria of dynamism and commitment levels. The Girdworld testbed outperforms the Tileworld testbed most of the time.

CONTENTS

List of Figures

List of Tables

List of Algorithms

Chapter 1- General Introduction

Chapter one

General Introduction

1.1. Introduction

Recently, there is a growing interest in the scalability of intelligent agent behavior in unpredictable and dynamic environments. Agents' actions are computational resources, their deliberations about what to do take time, so in dynamic environments, things change and some things change as the agent engages in thinking.

In such an environment, new options appear and existing options in the agent's deliberation period disappear. Scholars have presented solutions in this subject. Among those solutions is meta-level reasoning, reasoning during implementation and applying decision theory [1-3]. The inference was also developed among agents [4].

Decision theory deals with solving the problem of decision-making among a group of options within a variety of methods. Among those external influences are the continuous environmental changes, and these influences may be other than changes in the environment, such as the difficulty of accessing or the environment or surrounding with its details. As for the internal influences, they are what the agent relies on that occupy almost all the joints of life at the present time, what the agent adopts from the mechanism and a strategy in detecting his environment and algorithms for making a decision in an environment that it knows or is ignorant of about important parts of his surroundings.

Decisions taken need to be implemented, a plan must be drawn up to implement those decisions, and this is the responsibility of the rational agent.

The set of procedures is the basic structure of the plans. The plans vary according to the diversity of their procedures and objectives. Several plans may be presented to implement a specific goal. The costs of the plans are one of the most important things that the rational agent looks at; in addition to the value of the achievement compared to the goals he has confidence in achieving. The agent's task is further complicated by the addition of strategies to process, update and approve plans.

1.2. Deliberation

Deliberation typically begins by trying to understand what options are available to a given agent [5]. Such deliberation function can be decomposed into two distinct functional components:

- *Option generation*: in which the agent generates a set of possible alternatives; the option generation represented via a function option, which takes the agent's current beliefs and current intentions to determine a set of options that will hereafter refer to as desires. The intuitive interpretation of a desire is that, in an ideal world, an agent would like to achieve its desires set. However, in any moderately realistic scenario, an agent will not be able to achieve all its desires. This is because desires are often mutually exclusive[5].
- *Filtering*: in which the agent chooses between competing alternatives, and commits to achieving them. When an option pass successfully

through the filter function, and is chosen by the agent as an intention the agent is said to have made a commitment to that option. Commitment implies a temporal persistence, in that, once an intention has been adopted it should not immediately evaporate. A critical issue here is to determine how committed an agent should be to its intentions. In other words, how long should an intention persist, and under what circumstances should an intention vanish?

The mechanism an agent uses to determine when and how to drop intentions is known as a commitment strategy. The following three commitment strategies are widely used in the literature of rational agent [6]:

- *Blind commitment (fanatical commitment)*: where agents with such commitments will continue to maintain an intention until it believes the intention has actually been achieved, and they do not leave it even if it is impossible to achieve.
- *Single-minded commitment*: where agents with such commitments will continues to maintain an intention until it believes the intention has been achieved, or else that it is no longer possible to achieve the intention.
- *Open-minded commitment*: where agents with such commitments will maintain an intention as long as it is still believed possible.

Bold agent represents a single-minded agent, which never stops to consider whether or not its intentions are appropriate. However, a cautious agent represents an open-minded agent, which stops to reconsider its intention after the execution of every action. Hence, we are presented with a dilemma, that is, the agent that does not stop, to reconsider its intentions sufficiently enough will attempt to achieve its intentions even after it is clear that they not appropriate. Similarly, the agent that constantly reconsiders its intentions may spend insufficient time actually working to achieve them, and hence runs the risk of never actually achieving them, and/or consuming valuable computational resources.

There is clearly a trade-off to be struck between the degree of commitment and the rate of intention reconsideration.

1.3. Reconsideration Strategy (RS)

The finding of a suitable policy for deliberation process is represented as one of the most complex problems in the design of BDI (Belief-Desire-Intention) agents.

Simply put, the idea is that at any given time, an agent will have a variety of intentions, which are related to states of affairs that the agent has committed to achieving.

An agent selects strategies that are suitable for achieving these intentions; when a specific strategy for a given intention fails, typically, the agent may re plane to find another course of action for this intention. On the other hand, the intentions of a rational agent would not remain constant. Logically such this agent to reconsider its intentions from time to time [7].

Deliberation (deciding what intentions to accomplish) and means-ends logic (deciding how to achieve certain intentions) are the two major activities that BDI agents' decision-making consists them [6].

Deliberation is a computationally expensive operation, so a BDI agent should only deliberate when necessary this requires suitable intention reconsideration strategy [8]. The investigating the performance of intention reconsideration policies in environments where the following parameters are varied:

- Accessibility (the extent to which an agent has access to the state of the environment);

- Determinism (the degree of predictability of the system behavior for identical system inputs) and
- Dynamism (the rate of change of the environment, independent of the activities of the agent),

These methods centered on algorithms for automated plan generation, which would take as input a description of the current world state, environment, a goal to achieve, and the action options available to an agent, and would provide as output a plan to achieve those goals states.

1.4. Motivation

In artificial intelligence, the choice of action in a dynamic environment is one of the most important designs of intelligent agents; it is reacting planning that has a central role in the agent structure[9].

It is clear that we need intelligent behavior in a vast world with many, changing and diverse events, and this intelligent behavior and one of its main premises is deliberations that are predictive.[10].

From the foregoing, it becomes clear the importance of adopting the process of disclosing the contexts of the deliberations and the mechanisms for controlling them and their timing by replacing plans according to appropriate thresholds to achieve the most desired goals.

Decision-making strategy and reconsideration of decision using available resources at reasonable cost is a challenge for the intelligent agent, the decision making is an intention according to explanation of Bratman when he produced the BDI agent (belief, desire ,and intention) [4], that means the intention is under agent's control to adopt it or to update this intention by replaced it, contrary to belief and desire which are outside the control of the agent.

The agent who adopts policies that tend towards achieving the goal in light of cases governed by randomness is only conditional plans[11]. Disclosure of these policies and their repetition at work is what we want to stand on in what we offer.

When the agent deals with the available options and surveys, is the agent guided by his mechanism of reconsidering the available options and achieving the highest goals, or is it being overly preoccupied to achieve more goals that will waste many opportunities?

1.5. Research Hypothesis

The three main components that underpin an agent's rational behavior are deliberation (to denote deciding what to do), means-ends reasoning (to denote how to do it), and some control mechanism. All these processes are time and resource-constrained. Our hypothesis is to design and implement a system that detects the appropriateness of certain strategies within certain environmental changes. Until now, current models have assumed that the means-ends reasoning process takes place completely offline in the form of precompiled simple plans. However, the agent may as well be endowed with the capability to construct its own plans. Here we assume that the agent uses simple plans, and a given agent is concerned mainly with deciding what to do rather than how to do it. So, our model will test the deliberation process for each agent rather than means-end reasoning. In other words, the proposed model care with a reconsideration strategy for each agent and it calculate the effectiveness for each agent depending on its own reconsideration strategy.

1.6. Problem Statement

Decision theory deals with solving the problem of decision-making among a group of options within a variety of methods in different environmental conditions. Situated agents that act and react autonomously without human intervention, try to reach their goals in maximizing (scores, profit,…). A dynamic environment with a different level of dynamism that changes over time puts these agents facing big challenges to take the right decision in a proper time under uncertain and non-deterministic conditions.

Situated agents have different reconsideration strategies to reconsider their current plan and degree of commitment to its current plan. So, the behavior of these agents will be different.

The research problem in this thesis is an improved understanding of the relationship between agent design and environmental factors. In addition, what affects the agent's perception of its environment due to its own reason such as sensors, speed, and time, or because of rapid changes around it. By this evaluating the degree of suitability of the agent to a given level of changes in the dynamic environment will show.

In other words; the research problem of this thesis is how to design and implement dynamic testbed platforms in a simple way for intelligent agent architecture such as situated agents, to evaluate agent's behavior under its rules for reconsideration strategy, and degree of commitment in its current plan.

1.7. Aims and Objectives

The aim of this thesis; to create a simulated environment with an embedded agent that enables experimenters to test reconsideration strategies adopted by intelligent agents in dealing with unpredictable dynamic environments. This environment can evaluate agents working with reconsideration strategies, which is, in short, a mechanism adopted by the agent that describes the degree of commitment to its plans with regard to changes in the dynamic environment.

The main objective of the thesis is to design a simulation for a dynamic environment consisting of a grid of cells (squares) on which various objects can exist. These objects can be any one of the following, agents, and holes.

To achieve the main objective of the thesis, as a first step, to build a dynamic environment that simulates reality, and this environment has parameters that control the level of its change over time. With this environment exists there is an evaluator enables the evaluation of the agent's performance.

The second stage to present a complete system is to build a rational agent with specific capabilities such as movement, observation, adoption of plans, and action. Maybe its plans change with the change of its environment. This agent has parameters that control the level of his interaction with environmental changes. The effect of the designed environmental change on the agent emerges at a certain level of commitment to its plans or drop them. These environmental changes are considered circumstances that the agent lives and interacts with.

1.8. Research Contributions

The contributions of this thesis are introducing an improved understanding of the relationship between agent design and environmental factors. In the future, when faced with a performance domain for an agent, one should be able to draw on such an understanding to choose more wisely from the wide range of implementation possibilities available. And the Gridworld testbed has been demonstrated and has been shown to be an available system for evaluating agent architectures. Also, a Gridworld testbed is simplified, is easy to deal with, and qualified to be a platform for testers in that it does not require much effort to deal with. In addition, the Gridworld architecture is not just the simulated environment but has an embedded agent in the environment as well. The Girdworld testbed outperformed the Tileworld testbed most of the time.

Another contribution is introduced EEA-estimator is efficient enough that it can itself be used as the filter override mechanism for the more complex deliberation components.

1.9. Scope

The proposed system was designed and implemented using the Python programming language, version 3.9, through the PyCharm platform, which is produced by JetBrains Company. Where a number of libraries were employed in the creation of the program as (threading, timing, ..). The Python language has advantages, including ease of programming and the ability to call libraries within other languages such as Java. The proposed system works on a computer with the following specifications:

- Processor: Intel(R) Core(TM) i5-1035G1 CPU @ 1.00GHz 1.19 GHz.
- RAM 8 GB.

- Hard disk: 1 (TB)
- Operating System: Windows 10 Pro, 64-bit operating system

1.10. Literatures Review for Testbeds

This section provides a brief, overview of the works close to our research and test environment design:

1. The TILEWORLD test bed is a global benchmark [12-14], an unexpected dynamic system that provides controlled, observable, and traceable experiences built for adaptive agent architecture. This environment is an incubator for the agent, tiles, obstacles, and holes in its two-dimensional grid profile. The capabilities of the agent available within this system are the movement vertically and horizontally, taking into account that the agent does not collide with one of the obstacles or the limits of the network, as the agent's task is to close the holes with tiles. The holes have both capacity(C) and value (score). For the agent to push the tiles next to the gap towards it to close, the agent will have a rating of the score(S). Every experiment has a period limit; agent performance is calculated by the trial's score when the time expires. External events exist in Tileworld, objects (holes, obstacles, tiles) appear and disappear over time to give a simulation of the real world. The rate at which objects appear and disappear is under the control of the experimenter, in addition to capacity and score, which are properties of objects. The most important feature of Tileworld is the ability to control these parameters in a way that allows us to explore the different characteristics of the organized world (a world that changes relatively or slowly). Finding the systematic relationship between world features and those of the embedded agent is the overarching goal of this exploration. The Tileworld framework is

distributed with a fundamental agent architecture, which is likewise defined to permit control by the experimenter (see the accompanying conversation). The Tileworld does not enable the agent to switch on or use the sensors, but rather gives the agent an accurate and complete view of it (full observation). It becomes up to the agent or embedded agent designer how the data is used.

2. The independently developed NASA (National Aeronautics and Space Administration) Tileworld (NTW)[15] [6], the 2D test bed has tiles and has no holes or obstructions. External events are caused by winds blowing on the tiles in the grid.

   Simulators here have two characteristics:

   First, the NTW test system has no implicit proportion of accomplishment that closely resembles the thought of a score. What the agent should do and what establishes achievement is left completely to the experimenter. The *second* is the idea of the interface between the agent and its current world. The TILEWORLD agent calls the test system as a subroutine and gives data to and fro utilizing a common information structure (shared). The NTW agent and the world test system run no concurrently: The agent presents orders on the world, which are placed in a line (queue) and in the end executed. Operators can be programmed to fail probabilistically: A grasp operation might not result in the agent holding the tile, and a move might result in the agent being displaced to an adjacent location other than the one intended. The agent is given no indication of whether an operator has succeeded or failed and must explicitly sense the world to ascertain the effects of its actions.

3. The MICE simulator presented by Marenostrum Institut de Ciencies de l'Espai' (MICE) collaboration [8][16], is a grid-situated test system, intended to help examination to organize the problem-solving behavior of

various autonomous agents. Consists of only a number of agents, those agents can be objects like tiles, obstacles, and forest fires, the world is inhabited by only agents, and diversity may be their condition. The move command is the MICE operator, from cell to neighboring cell moves the agent. To hold things by the agent, it must use the link command. External events caused by agents have effects on the world (to simulate rain, a cell is wet and slippery). The main difference between the MICE simulator and the NTW and Tileworld simulators is that MICE makes even less of a commitment to world physics; the experimenter defines an agent's sensing and effecting capabilities and also the effect of actions taken simultaneously by the agents. MICE might be viewed more as a framework for building testbeds rather than a simulator in and of itself. (Copies of Phoenix and Tileworld built with MICE designers. See[8], As instance).

4. PHOENIX [14,18] is a framework for carrying out and testing numerous autonomous agents in a perplexing climate. The procedure is putting out fires; the world comprises of a guide with shifting territory, heights, and climate. Flames can begin at any area and spread according to the encompassing lands. Agents are putting out fires units (tractors or bulldozers as commonly) that change the shape of the land to control the flames. A distinction has to be made between agents, simulators, and the environment for Phoenix. The simulator has three core functions: the first is to manipulate and update the map, the second is to synchronize the action of the world and the agent, which are executed as autonomous assignments, and the third is to accumulate information. The PHOENIX world incorporates a portrayal of Yellowstone National Park (from Defense Mapping Agency data) and the undertakings that execute fires. PHOENIX agents produce assignments that mimic a fire chief, a number of tractors, lookouts, helicopters, fuel big haulers, etc. Agent's errands

incorporate getting through the map, stopping a fire line, estimating (prediction) the course of flames, arranging the assault on the fire by a number of tractors, checking progress and distinguishing disappointments in assumptions, and recuperating from disappointment. Tasks embed themselves (by sending messages) onto a timetable kept up by the PHOENIX simulator. Agents and fires only are types of objects in the PHOENIX world. However, agent and fire determine their behavior using the information found in cells, each cell in the environment possessing information. For instance, the black road designed on which the bulldozers are moving quickly, and the flames escalate faster towards the designated path uphill. As for the external events, they also have an indirect effect, with the blowing of the winds, the eruption of fire accelerates. The PHOENIX agents have restricted sensory and actual capacities; for instance, tractors have a 200-meter range of seeing (albeit the view isn't influenced by height), and they're moving and cutting fire lines at rates systematized by the U.S. Forestry Service.

5. TRUCKWORLD [19,20] is a testbed multi-agent system, designed for reaction running theories to be tested with it, and provides supportive images (examples) for reasoning theories about dynamism and unpredictability [21,22]. The principal responsibility is to give a reasonable world (realistic) to its clients. However, without sensors or actual impacts (physical effects). An agent is a truck comprising 2-arms; two load coves; a number of sensors; and different parts, for example, a bunch of tires, a gas tank, and way and speed regulators. It works in an environment comprising of streets and areas. Streets associate the positions, objects populated them. The test system itself places not have many limitations on the conduct of objects; it may have a degree of complexity. Model objects can be showing by TRUCKWORLD, for example, Trucks can increase their fuel by using fuel barrels, they are

objects, safe driving on slippery roads, trucks turn into tire chains, they are objects, and so on. External events: Rainstorms blow frequently in the world, roads become slippery and dirt roads are muddy due to these rains. Without tires to stick to muddy roads, trucks run the risk of getting into the mud. Rainstorms moisten things whose action (behavior) is influenced by humidity. The events are related to random factors and environmental characteristics (rain in hours of the day is more bearable than other hours).

6. The World Cup Robot Soccer (RoboCup) is a study by the scientist Hiroki Kitano in 1993, and it was announced as a simulation that allows research for multi-agent artificial intelligence systems in 1995, is the first project to use football for research and education. The idea is that a soccer game is an excellent testbed for robotics and artificial intelligence research[23].

   The goal of the study is to develop the soccer robot team, and the other goal of the project is to evaluate various theories, algorithms and architectures of agents. Among the characteristics of this system is that it is a dynamic environment, the state of change is in real time (not as in a game of chess), the accessibility is incomplete, and the control is distributed and not centralized. The project has a software link (simulation) that allows researchers to participate in the program, with the reactive behavior of the agent and gain business strategies and planning mechanisms within real-time and multi-agents. This system covers a number of areas: agent architecture, planning, real-time knowledge, real-time reasoning and acting in an unpredictable environment, multi-agent systems.

   The design of the agent have to perform multiple actions such as hitting a ball, blocking it, passing it and hitting it with the head, all of these actions are behaviors subject to standards and limitations. The agents in the system have to play a social game based on the principles of autonomous agent design, collaboration, thinking and acting in real time. The main problem of designing the agent to deal with such a research environment is to combine the two approaches of reactive and deliberation, reactive, for example, is the speed of movement with the ball and in the field, while deliberations depend on planning and reasoning, filtering, and update intentions. From the above, it is possible to see the challenges facing the agent, the first of which is the dynamic environment of the game, the movement of the opposing team, which may represent a state of unexpected dynamism, the movement of cooperating partners, time-limited communication resources, the mechanism of distributing plans and building them among collaborators.

1.11. Layout of thesis

The rest of the chapters are in the following order:

- The second chapter is an introduction to the most important concepts in our research, such as the intelligent agent, its characteristics and architecture, and environmental characteristics. And in addition to standing on the main implications of the research, such as the concept of decision-making and the uncertainty of the environment, meta-reasoning, and the similar testbed environment.
- The third chapter contains our proposed system Gridworld testbed, exposure to the system specifications and its mechanism of action, and how used the system and the The mechanism of displaying the results with their meanings.

- The fourth chapter includes an experiment with results that are listed and explained in detail, with a comparison with the results of other work to support our research effort.
- The fifth chapter is a conclusion of the above efforts and what we hope for as future proposals.

Chapter 2 – Agent and Environment

Chapter 2

Agent and Environment

2.1. Introduction

In recent times, controlled experiments within AI research are in a significantly increased state, with changes to the system or environment by designers in the specifications of that system that affect the aspects of the system in terms of efficiency. Two vocabulary that are well-known in the community of researchers [10][24] , namely:

1. Benchmarks that are units of measurement for key concepts.
2. Test beds (virtual environments)

According to our view, there are some challenges related to the appropriate use of the experimental method, and this does not reduce our impulse that this approach has support for artificial intelligence.

Two main goals behind the benchmarks and the testbed, the first of which is to set clear indicators (matrices) through which we can choose between systems that have similar specifications (competing), at the same time we need a theoretical coverage of the practical effort[10]. The scientific competence of the testbeds and the benchmarks architectural measures lies in their ability to uncover the scientifically agreed upon interesting aspects to evaluate the performance of the system.

The practical (experimental) control achieved by employing the test bed is considered a distinct assistant in the perception and interpretation of the reasons for the behavior of the systems under examination and testing. Artificial intelligence systems are distributed in different locations with highly complex, interfering and complex environmental influences, the most important aspect of the test environment is the simulation that has a literature that must be adhered to when designing the testbed, and achieving a high level of reproduction status for the realistic environment. The controller of the virtual environment has the right to change the parameters (features) of the environment and calculate the effects on the system that is under test, provided that these variables are random and not random.

There is a very important question, this question stems from the difference between realism and the ability to control practical (experimental), this difference can be reduced to an acceptable degree when testing systems that operate within an environment described as deterministic, but the subject is more complex when the talking in the stochastic environment and this is what we will address in the environmental characteristics.

In other words, the focus of our attention does not revolve around interacting with simple systems within uncomplicated environments, the fact that artificial intelligence systems distributed around the globe must deal with extremely complex environments, and we do not close our eyes to simple designs, which provides us with a benefit when we simplify Complex systems into multiple simple forms combined in a system described as complex.

One of the best solutions is to pay attention to designing an environment close to reality and to give satisfactory and clear results for systematic experiments near those environments.

2.2. Benchmarks and Test Beds

Benchmarks have a clear aspect in computer science as a tool. An example illustrates this. Matrix multiplication is a tool in computer science, in other words, a good Benchmark. The designers of the computer's CPU do not tolerate without focusing on the speed of the processing unit, so the use of matrix multiplication to give them an image of the speed, then matrix multiplication as a tool is a criterion for evaluating the serious performance of the processing unit.

A Sussman anomaly was an early Benchmark for AI planning programs (the three-block problem) [25]. Sussman anomaly Support many researchers in uncovering the mechanism of their schematic work, and the reason for fame simulates multiplication matrices with what it represents of an important class of problems that have interactions and interrelationships between the branching objectives easy to describe.

Let's first know the behavior of the agent, The behavior of the agent is according to the context around it, as it is aware of what is happening in order to adopt a specific action, the agent who can adapt to the surrounding circumstances is the intelligent agent, and it has characteristics and features with details that we will mention in the subsequent sections.

Benchmarks draw us the behavior of the agent at a bright level, this agent that we are interested in, and usually focus on part of the behavior of the agent and not all the behaviors, so testing all behaviors means putting the agent in several tests, in other words, adopting a number of Benchmarks. Whether you are a scientist, a designer, or as a consumer, this dictates what you want to know about the agent's behavior. For instance, a consumer of search algorithms within artificial intelligence, this consumer may appear excited by what he sees from the outputs of the algorithm for min- conflicts heuristic in the problem of the n-queens in time is constant [26].

2.3. Intelligent Agent

From artificial intelligence and distributed programming came intelligent agents. The two sciences have united to appear smart customers, and it is called the science of distributed artificial intelligence[27]. The idea of the intelligent agent in 1950, and nothing actually came into being until the end of the seventies. The topic was widely known in the early 1990s, along with the Internet[28]. There is an analogy between an agent and a directed object being distributed programming and artificial intelligence. Researchers speak of artificial intelligence, not human, when referring to the intelligent agent as it deals with devices such as computers [29-31].

The definition of the intelligent agent suffered a lot of divergence in views from the researchers; we hope that there is clear vision that is useful in defining the intelligent agent, which gives a simplified and comprehensive knowledge base on this concept.

The main idea in defining the agent, which does not contradict the opinions of the researchers [32-34], is an entity with a purpose, there is a motive to build it in assisting the user in a specific task and in a specific environment, for this intelligent agent has basic features (autonomous and learning). Autonomous is a feature that distinguishes it from other programs. The autonomous agent is in addition to the previous definitions of the meaning of the autonomous, such as his ability to self-act and make decisions. The agent has an internal encapsulated state.

When the environment is dynamic, the agent must be aware of changes, it has knowledge that enables it to carry out its task. The learning feature is a human feature that enables us to be smart. Communication and cooperation, they are not two main qualities, we have known their role, but the reason behind making them non-principal is the possibility of designing an intelligent agent without them, for example an agent may learn through communication, but it is possible to learn from monitoring and this simplifies the internal structure of the agent, as for cooperation between agents it is to simplify the complex task, break it down and distribute it to other agents. The entity's behavior spontaneously provides us with information that we may be interested in and did not request. As for mobility, it has an effective role in migrating the agent to another location to carry out the task.

The distinction between human intelligence and machine intelligence is concept that must be understood. Intelligence is automatic for the agent because it does not exercise human intelligence, so we see the intelligent agent falls within the framework of its ability to autonomy and learn, or we name agents according to the purpose for which they are designed, such as information agents Nwana [34].

2.4. Agent characteristics

The different opinions about the interpretation of some of the characteristics (what is means autonomous) by researchers, one of the reasons that gave various definitions of the intelligent agent. What we have chosen from the characteristics is the most important focus of researchers, directly or indirectly, to define the intelligent agent. However, these characteristics are not the only ones that all researchers have adopted.

2.4.1. Autonomy and Intelligence

Carrying out autonomous actions is the most important thing agreed upon by researchers. An autonomous agent has the ability to control his actions and behavior, it acts without the interference of others (user or agents), in addition to his the initiative in achieving his goals[30].

Autonomous and intelligence some scholars equate them. Meaning, they consider the ability to take initiative on the part of the agent an advantage for being an intelligent agent. Foner believes Julia is an intelligent because it is autonomous and has the initiative[31]. But the researchers are not talking about human intelligence, but machine intelligence[29]. That is why Petrie spoke of autonomous and not about intelligence, because he links the word intelligence with human intelligence, which is difficult to describe.

2.4.2. Learning

Intelligence is a word that has the quality of learning. The question arises for the agent to be intelligent; does it have to be learning? A number of researchers believe that learning is an important attribute of an agent[31], while others never mention it[30].

When an agent is present in a dynamic environment, it is not possible to know what it will face, so the agent is distinguished from others when it has the ability to adapt and learn from his environment. What is a learning program? It is a program that uses memory to solve problems after saving things. Learn the agent from monitoring the user, its environment, or other agents. The agent is in contact with others (user or agent) by means of natural symbols or language that guarantee its communication, and it is a two-way letter. The second way to learn is from the agent's own programming[32]. Thus, the methods of machine learning are either by communicating or by programming the agent itself.

2.4.3. Co-operation

Complex problems may require more than one agent to cooperate in solving them[28]. Cooperating agents agree on the goals to be reached and achieved. When using agents, cooperation is the main distinguishing part of them[33].

Foner deals with problem solving in cooperation between agent and user[31]. The cooperation is in two dimensions, it may be between the user and the agent, and it may be between the agent and other agents. Agents ’collaboration breaks tasks down into smaller, distributed problems for agents to solve. This cooperation takes place in the presence of communication between the assistants.

If we imagine that there is no communication, then how can cooperation be? The answer is through observing. The agent is the one who determines whether or not the second party needs it. This is a manner work without communication[28].

2.4.4. Lifelike

Agents lifelike, it is the style of delusions, so emotions and social interactions are within the capacity of the agent. Various techniques and mechanisms are required to create lifelike characters, including speech recognition and animation. In addition to psychological sociology and methods of dialogue, it is necessary to understand them [28]. It is interesting to be agents with human characteristics, such as emotion, that can be portrayed as a realistic human face[34]. Some researchers argue that showing realistic traits through an agent is an imperative[35]. A spontaneous agent, not a mechanical one, behaving unpredictably, it is a lifelike agent.

2.4.5. Mobility

The movement through the network, such as the internet. Is the agent capable of this movement? Researchers believe that moving the agent across the network provides a new method for the process of retrieving data and transactions[36]. The agent can move from one server to another to find the best journey for you, and give you the result by that. The disadvantage in this matter is that the agent may present to your personal computer many options that consume time and oblige you to stay connected to the Internet.

2.4.6. Situated

In recent years, the concept of software agent has become an important area of research in both Artificial Intelligence (AI) and computer science. The range of applications varies from small systems for personalized email filters [4] to large and complex systems such as air-traffic control [37]. Intuitively, agents are systems that can reason and decide their course of action, for instance satisfy their (design) objectives and/or goals [38,39]. An agent is said to be rational if it chooses to act in its own best interests, given a belief set it has about the world.

One goal of the AI community is to engineer computer programs that can act as autonomous, rational agents, and can independently make rational decisions about what actions to perform. But it is not enough to have programs that think of good actions to perform regardless of their environment's state – hence, agents behavior needs to be situated and aligned with their environment state [39].

- Definition 1. *Situated Autonomic software can be envisaged as a system which acts and/or reacts autonomously to external stimuli, generated from sensing its environment*[4].
- *Definition 2. Situated agents do not use long-term planning to decide what action sequence should be executed, but select actions based on the locally perceived state of the world and limited internal state*[38].

As illustrated by Figure 2.1, the agent takes sensory input from its environment and reacts if required based on a given action groundings or policies. The environment that an agent occupies may be physical (in the case of robots) or a software environment (in the case of a software agent). In the case of physical embodied agent, actions will be physical such as moving objects around. The sensor input received by an agent can include video feeds. However, in the case of software agent, actions will be software commands such as UNIX command, which removes a file, and sensor input will be obtained by performing command such as   which obtains a directory listing [7].

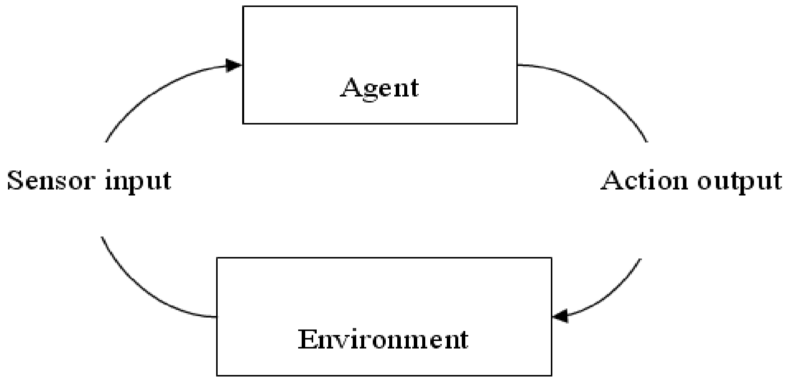


Figure (2.1): The agent takes sensory from the environment and produces as output actions that affect it[40].

In almost all realistic applications, agents have at best partial control over their environment. Figure (2.2) shows what is adopted.

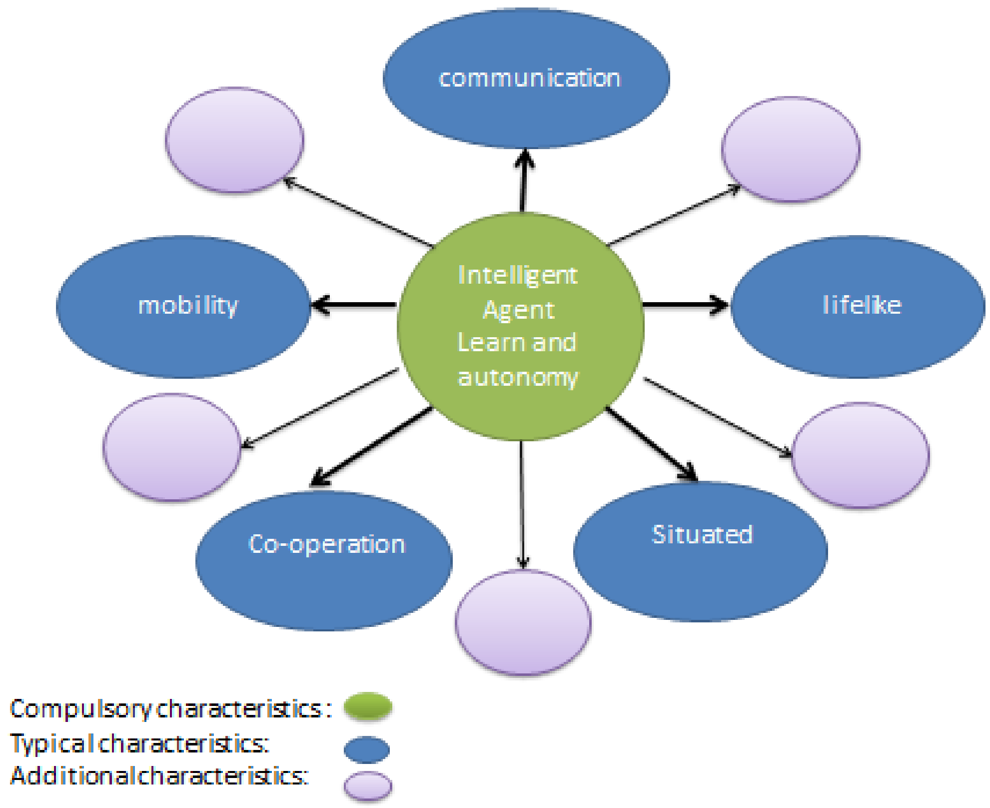


Figure (2.2): Intelligent Agent Characteristics[31]

2.5. Agent Architectures

Defining the intelligent agent in advance makes it easy for us to understand the various structures within an agent design. What we mention here is a number of structures based on the action of the agent, i.e. the mechanism of interaction of the agent with its environment[41].

The main point in this part is to give an idea of the design of an agent that achieves the previous characteristics such as autonomy, reactiveness, proactive, social ability and so on.

2.5.1. Purely Reactive Agents

The agent who relies on the mechanism of choosing its acting without relying on its awareness of previous information Therefore, the actions of the agent are selected based on the current state (the agent's perception of current information) [42]. We have in this type of agents a direct response to its environment.

It is may be called Simple reflex agents. A simple example of a purely reactive agent, the water level agent in dams, it controls the water level at a certain level in the dam, closes and opens the gate, as well as the thermostat is an example of a purely interactive agent.

2.5.2. Utility-Based Agents

An agent has set of accessible runs. Each run has a bunch of activities to execute a goal. A utility means a numeric assessment on how perfect a particular executing is given the current awareness information from the world. A type of utility based agent, including a function that calculates the utility as a real number dependent on information from the perception of the environment.

With a high degree of utility, the agent accomplishes the operation process. This method ensures that there are attempts by the agent to improve its performance to the maximum extent.

In most environments, to generate high quality behavior we do not rely solely on goals. For example, a series of procedures achieves the goal by the arrival of the car, but part of those procedures is not safer or less expensive. The comparison of the happy state with the unhappy is done by means of a measure of performance. The term happiness has been replaced by the term utility (it's a term scientific). The meaning of utility is the quality of benefit. In this way, it is easy to distinguish between procedures that are more desirable.

A utility function of agent is basically a capacity of the measure of performance. when there is an agreement between the external measure of performance and the internal function of utility, choosing action's high utility by agent will be rational with the external measure of performance, see Figure (2.3) [43] [44] .

The utility-based agent makes decisions based on utility, although the goals are insufficient, when the goals are insufficient, we have two situations: The first is the goals are conflicting (such as speed and safety), here the utility is on the barter (appropriate tradeoff). Secondly, there are a number of goals, but they are not achieved with certainty. The benefit here depends on the probability of success versus the importance of the goals.

We should not fail to mention that our partial observable and stochastic environments are widespread in the real world. A rational agent makes decisions based on utility; this is the utility the agent expects in those environments.

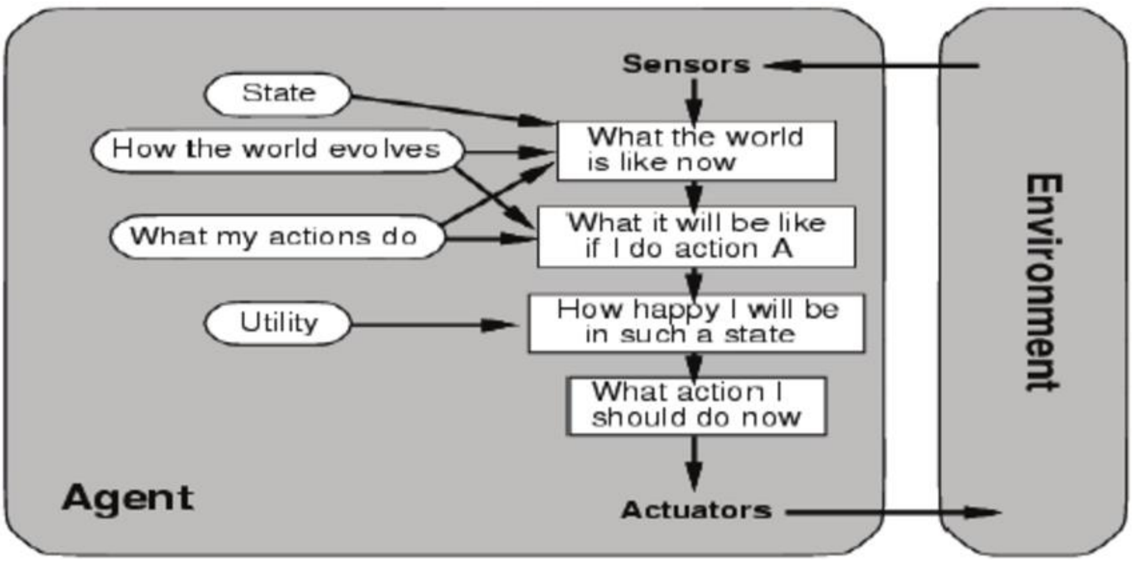


Figure (2.3): Utility Agent Architecture [43].

2.5.3. Goal Based Agents

The agent relies on an action plan to achieve the goal. The process of selecting a plan is not simple; the agent uses research and planning processes. Depending on the goal and perception, plans are drawn up, after which the agent checks if the goal can be achieved according to perception. In addition, the previous information may be used to define the procedures (selection process) and to identify what needs to be done, See Figure (2.4) [40].

For example, awareness is not sufficient to determine the necessary measures, in traffic intersections, the car has to turn left and right, or it is moving straight. Making the right decision here depends on the driver’s target location and the rider’s desire to have an influence on decision-making.

When an agent has to contemplate long, fluctuating steps and turns in order to find a line to achieve the goal, it is much more difficult. Therefore, research and planning are partial areas of artificial intelligence that involve finding and adopting a series of measures that achieve the goal [44].

The agent has a prospective study of the effects of its actions, and no single procedure is adopted. For example, when the driver sees a red lamp, he does not relay one action only, which is to press the brakes directly, but rather takes several procedures; its decisions are more flexible as they can be adjusted.

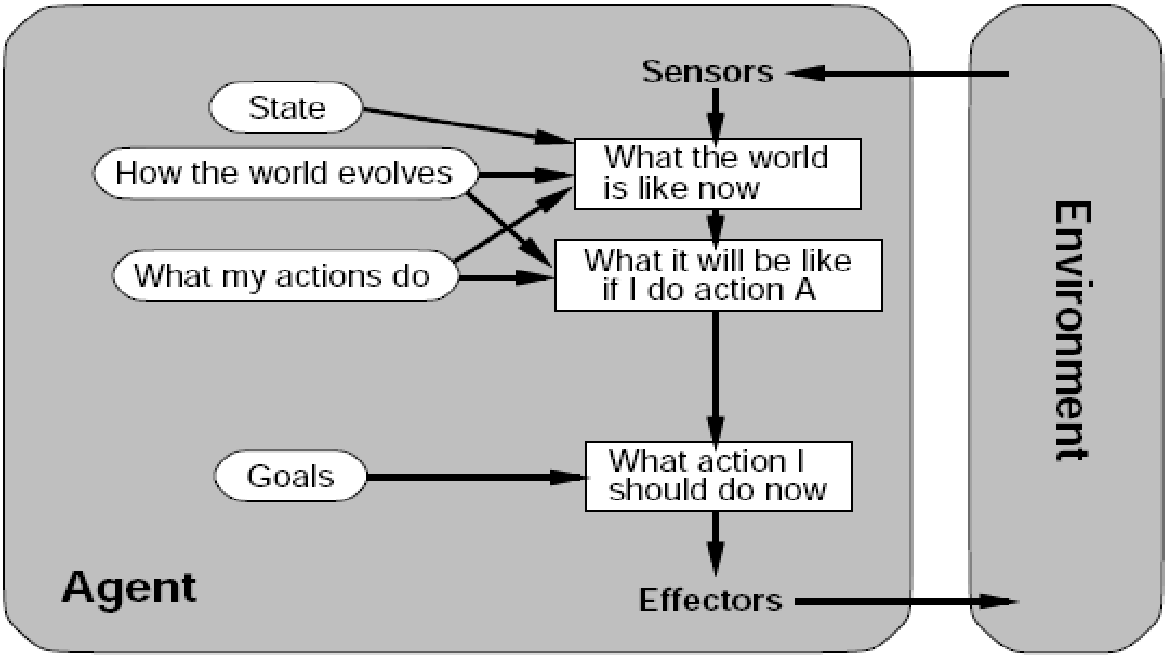


Figure (2.4): Goal Based Agent Architecture [40].

2.5.4. Rational agents

Wooldridge and Jennings provide a list of the abilities expected to be contained in the design of a rational agent[45]:

- ✓ Reactivity ability: Perception of the environment is among the capabilities of the rational agent, and in a timely manner it has a response to the changes occurring in the environment, this in response to the design of capabilities of the rational agent.
- ✓ Proactiveness ability: Taking the initiative, adopting goal-oriented behavior by initiative, in response to the design of capacity of the rational agent.
- ✓ Social ability: the ability to communicate and interact with other agents, in response to the design of capabilities of the rational agent.

Doing something is considered right, what does that mean? It is a question that can be answered, but the answer will be within a framework that interests us, which is the behavior of the agent.

The agent is immersed in the environment, from which a set of procedures is issued according to his perception of that environment, this group or sequence of procedures has consequences for the environment and it is a sequence of states. If the sequence of states in the environment is desirable, we say that the agent has a good performance, that desirability or approbation for the performance of the agent is captured through performance scale.

The focusing of the evaluation process is on the states of the environment and not on those of the agent. Noting that the factors for measuring the efficiency of the agent are varied according to the purpose for which the agent was designed. To clarify the idea of relying on environmental conditions(states) in the evaluation, for example, the vacuum cleaner, if we rely on the agent’s states in the evaluation, it will pick up the dirt and record a point for it, then throw it and pick it up again, also a point is calculated for him, while when the evaluation is on the cleanliness of the room after dividing it into Squares, and every clean square takes a point, so the scale is according to what we want and not according to what his agent want. The foregoing rationality has three pillars, the first is the measure of performance, which determines the level of achievement and evaluates it. Secondly, prior knowledge of the environment by the agent, in addition to the agent's awareness of the sequence of states in the environment, and the third, possible actions of the agent [44]. Note Table 2.1 that is briefs how are the agent makes decision.

For every conceivable percept sequence, a rational agent should choose an activity that is relied upon to boost its measure of performance, given the proof given by the percept succession and whatever underlying information the agent has[44].

2.5.5. Hybrid agents

For an agent's reactive and proactive behavior in the same structure, it's problem, the solution that is the introduction of a hierarchy of interacting sub-system layers, this is the well-used option [41]. In hybrid architecture, at least two horizontal layering, one of them to agent's reactive behavior and other to agent's proactive behavior, but stay issue in this structure, how agent control to take a decision between the two-layer or who is layer take the control option.

All layers in a horizontal layering architecture are sharing in the perception (input) and the action (output), a comeback on the start, must there is a control function that is considered a bottleneck in the agent's decision making. This function decides any one of the layers has control of the agent. See Figure 2.3 [41].

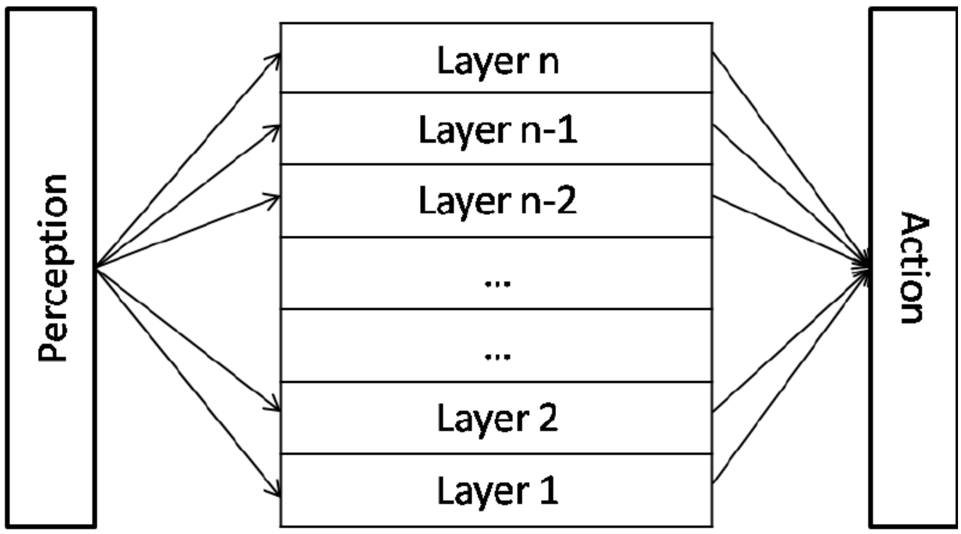


Figure (2.5): Horizontal architecture

In the second type of layered architecture, that is vertical layering, see Figure 2.6 [41], at least one layer interaction with sensor and action(input and output), vertical layering has two kinds the first is one-pass and the second is two-pass architecture. Through each layer and in a sequential direction the control and information streams, the last layer creates an action output that is described as one-pass architecture. Otherwise, in two-pass architecture, the shape of the control and information streams are in two directions, flow up and flow down through the layers.

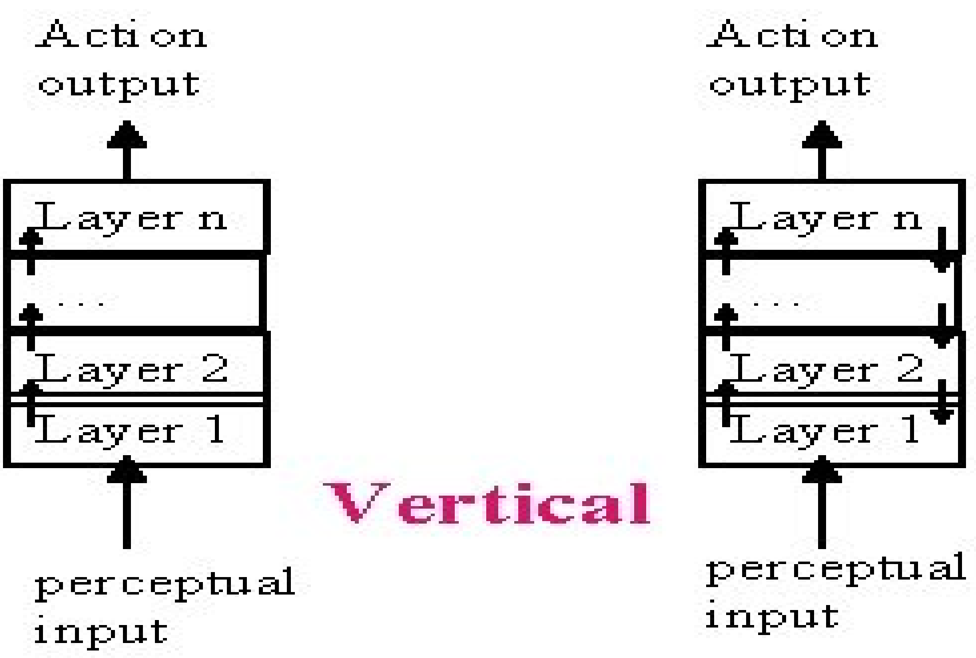


Figure (2.6): Vertical architecture

Table (2.1): Agents type with their decision

Agent type	How take decision	Example
Purely interactive agent	current state	Thermostat
Utility-based agents	Utility	route recommendation system(RRS)
Goal based	Research, planning, and	Procedural Reasoning System(PRS)

agents	perception	
Rational agent	Reactive, Proactive and social behavior	BDI agent (Believe, desired, and intention)
Hybrid agents	reactive and proactive behavior	Cooperative Intelligent Real-Time Control Architecture (CIRCA),

2.6. Environment characteristic

The real world around us has a lot of constants and variables that we deal with on a daily basis; this is triggered to design an environment with specifications that mimic our real world and has the properties of the environment. The environment has to design to embed the agent and give the domain to simulate active of the agent in the environment.

Russel and Norvig have put a description of the agent's environment[44]. The below paragraphs are the description of the characteristics of the environment.

2.6.1. Fully observable vs. partially observable

If the agent is able at every moment through its sensors to access the entire state of the world, then the environment task is described as the full observation [34]. When the sensors are able to detect all aspects related to the filtering of action, the environment task is fully observed. The agent in the fully observable environment does not need history to keep the state of the world in order to track it.

In the state that the sensors suffer from deficiencies for design or external reasons, the data is incomplete about the world, this environment is unobservable, like the driver of the autonomous car cannot predict what other drivers are thinking, but despite this situation, which seems hopeless, however, the goals of the agent achievability remain possible.

2.6.2. Single agent vs. multi-agent

The distinction between single-agent and multi-agent environments may seem simple enough [35]. For instance, a crossword puzzle game is an environment with a single agent responsible for finding the solution, while chess is a two-agent environment. In a single agent environment, there is one agent operating whereas in multi-agent environments there are many agents that interact with each other, but at times objects or entities that we would not normally consider as agents may have to be modeled as such. Nature may be modeled as an agent usually any entity/object that affects or influences the behavior of the agent under consideration needs to be regarded as an agent.

2.6.3. Deterministic vs. stochastic

When the agent can depend on the current state to know the next state of the environment in which it is embedded, and the action is carried out by the agent then we can describe this environment as deterministic, otherwise, it is described as stochastic. This is our talk when there is one agent. In cases characterized by more than one agent, it is deterministic when each agent can predict the actions of other agents. Thus, the partial observation environment is classified as a stochastic environment [36]. Most aspects in the real world are difficult to fully observe, which is why it is treated as a stochastic environment. For example, a self-driving car has a stochastic world, so the agent cannot fully predict the traffic, in addition to design events such as engine failure without warning. From the above, we come out with a conclusion: the uncertain environment is one that is not fully observable (nondeterministic). When we have no certainty about the results, the stochastic environment has no probabilities associated with the results.

2.6.4. Episodic vs. sequential

In the episodic state of the world, the experience of the agent is partitioned into molecular episodes. The agent is given data and then executions one action, that is in every episode. Critically, the following episode doesn't rely upon the activities in past episodes are taken. The episodic is classing for several tasks.

For example, the agent makes a decision regarding the current part without being indifferent to the previous decisions in determining the defective parts on the production line, in addition to that, if the next part is defective and not affected by the current decision [37]. On the other hand, in a sequential world, future decisions are affected by the current decision. Autonomous cars and chess are both sequential worlds, in which long-range convoys can be caused by actions of short-range. The sequential world is much more complex than the episodic world because the responsibility of the agent in the sequential world is to think ahead.

2.6.5. Static vs. Dynamic

In a static world, the agent does not need to look at the world without interruption while deciding on a specific action, in addition to the fact that the passage of time does not pose a concern to the agent, so it is easy to deal with a static environment. When an agent is in deliberation and changes in the environment may occur, that environment is termed the dynamic of that agent. The dynamic environment continuously asks the agent a question, what do you intend to do, if a decision has not been made yet, this is a decision to do nothing. We have another name, which is semi-dynamic, this name refers to the environment that changes due to the action of the agent [46]. If the environment itself does not change with the passage of time but the agent's performance score does, then we say the environment is semi-dynamic. Taxi driving is clearly dynamic: the other cars and the taxi itself keep moving while the driving algorithm dithers about what to do next. Chess, when played with a clock, is semi-dynamic. Crossword puzzles are static.

2.6.6. Discrete vs. continuous

The discrete/continuous differentiation applies to the condition of the world, to the manner in which time is dealt with, and to the agent's perceptions and activities, for example, a limited number of distinct states in the chess world (barring the clock). Also, a discrete set of perceptions and actions are in the chess world. While there is state and time-continuous problem in self-taxi driving: the taxi's position and speed and other cars move through a scope of continuous values and do so easily with time. Self-Taxi-driving activities are likewise ceaseless (continuous) and the input by a set of cameras is discrete.

2.6.7. Known vs. unknown

Strictly speaking, this description is not directed at the environment per se, but rather at the agent's state of knowledge about its environment. All results are known for all actions, this is in the known environment.

In an unknown environment, the agent must learn how that environment works in order to be able to make the appropriate decision. Note, the difference between the full observation and the partial observation in the environment is not the same as the difference between the known and the unknown environment. For example, in solitaire games, we have knowledge of the rules of the game, but there is no vision for the cards that have not been received, it is a known environment and partial observation environment. And an example reflects the other picture when the environment is unknown and fully observed, when watching a new game video, we have a complete vision of the game but we do not know how to use the buttons.

As one might expect, the hardest case is partially observable, multi-agent, stochastic, sequential, dynamic, continuous, and unknown. Taxi driving is hard in all these senses, except that for the most part the driver's environment is known. Driving a rented car in a new country with unfamiliar geography and traffic laws is a lot more exciting. Table 2.2; lists the properties of a number of familiar environments. We have not included a "known/unknown" column because, as explained earlier, this is not strictly a property of the environment. For some environments, such as chess and poker, it is quite easy to supply the agent with full knowledge of the rules, but it is nonetheless interesting to consider how an agent might learn to play these games without such knowledge.

As expected, the most difficult case is (sequential, stochastic, multi-agent, partially observable, dynamic, continuous, and unknown). The interesting question about how the agent learns and provides it with knowledge and that is under the heading reinforcement.

Table (2.2): Examples of Environments and Their Characteristics[44].

Task Environment	*Observable*	*Agents*	*Deterministic*	*Episodic*	*Static*	*Discrete*
Crossword puzzle	Fully	Single	Deterministic	Sequential	Static	Discrete
Chess with a clock	Fully	Multi	Deterministic	Sequential	Semi	Discrete
Poker	Partially	Multi	Stochastic	Sequential	Static	Discrete
Backgammon	Fully	Multi	Stochastic	Sequential	Static	Discrete
Taxi driving	Partially	Multi	Stochastic	Sequential	Dynamic	Continuous

Medical diagnosis	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Image analysis	Fully	Single	Deterministic	Episodic	Semi	Continuous
Part-picking robot	Partially	Single	Stochastic	Episodic	Dynamic	Continuous
Refinery controller	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Interactive English tutor	Partially	Multi	Stochastic	Sequential	Dynamic	discrete

2.7. Beliefs Desires Intentions Model BDI

One of the most notable models for deliberative agent architecture is the Beliefs, Desires, and Intentions (BDI). The BDI model was first proposed by Bratman [47] as a design for deliberative software agents. This model came forth from the philosophy of practical reasoning. The theory has been used for the logical modeling of agent-based systems. Practical reasoning has also been applied to implementing agent-based systems. It is important to distinguish practical reasoning from theoretical reasoning. Theoretical reasoning is directed towards beliefs. For instance; if we believe that all women are beautiful, and we believe that Saja is a woman, then we will usually conclude that Saja is beautiful. The process of concluding that Saja is beautiful is theoretical reasoning since it affects only our beliefs about the world. While the practical reasoning is directed towards action, for instance, the process of deciding to catch a taxi instead of the bus.

The BDI architecture originates in folk psychology and practical reasoning. Folk psychology [48] is distinct from the scientific psychology that one might be more accustomed to:

Definition. "Folk psychology is the common-sense conceptual framework that human beings employ to understand, predict and explain the behavior of other humans and higher animals."

Folk psychology uses terms from the family of mentalistic concepts like beliefs, desires, knowledge, fear, and so on. In a way, it is not concerned with the lower level of the actual information representation and processing. An important feature of folk psychology, and its parent folk science, is that it works. It delivers a framework people use in their everyday lives. This does not only refer to inter-human relations, but to the interaction with other objects in the world as well.

2.8. Uncertainty and Rational decision

Decision-making in uncertainty, the basis of modern decision-making theory in uncertainty dates back to the eighteenth century, the principle of temporary utility maximization by Daniel Bernouilli [46] , while the first derivation of the representation of potential benefit dates back to Frank Ramsey [47]. Savage is from his which modern decision theory descends[49].

What Savage said has become a standard in decision theory, under conditions of uncertainty and what the agent faces. Three vocabularies have been distinguished by Savage: states, consequences, and actions, states are the capture of all states of the world that affect the consequences of actions, consequences are features, or in other words what the decision-maker is concerned with the features of the world, and the last singular is the link between states and consequences it is actions, a tool for creating dissimilar consequences in different situations, according to Savage, is a function from cases (states) to consequences.

There is a difference between certainty and uncertainty to describe the agent's cognitive state of his world. It is important to differentiate between a situation with known or given consequences, and a situation with unknown consequences in which the decision-maker is reliant on self-evaluations. The agent is unsure of the consequences in two places. First, he is unsure of the true merits of particular consequences. The example of owning a modern Porsche in the lottery does not guarantee that you know the speed of the car. Second, the agent is not sure of the value of the consequence, not because of its actual unknown features, but because of uncertainty whether the features have value or how valuable they are.

The factual reality of the decision problem is that in many places we are uncertain about the relationship between consequences, actions, and situations. For example, taking an umbrella will keep us from getting wet when it rains, but maybe the umbrella has holes that we don't know about, or there may be winds that affect the level of protection from getting wet. the using the term uncertainty is refer to one of the two meanings, the first being uncertainty about the consequences, the second being uncertainty about the options available, or both as third meaning.

Preferences with the ability of the agent to make a decision, these preferences are the dividing line between the possible outcomes of various plans. The outcome is a state that has several factors such as the time for the arrival of the agent. Preferences and outcomes fall within the heading of the utility theory; each state has a benefit and the role of the agent in choosing and preferring the most beneficial.

Decision theory will combine probability with utility to come up with a rational decision[49]:

Decision theory = Probability theory + Utility theory……..   (2.1)

An agent is referred to as rational when he chooses the action with the highest utility expected, after accounting for all possible outcomes.

2.9. Meta-Level Reasoning

The basic insight behind normative control reasoning is that computations are action. The utility of a computational action must be derived from its effect on the agent's choice of action in the real world [48]. Meta-level reasoning (meta-reasoning) means reasoning about reasoning [48-51]. Meta-reasoning can be viewed as an extension and revision of information value theory to cover computations in resource-bounded agents. When the base level problem solver operates using prior value estimates for the real world actions, the effects of computations can be assessed by using prior statistical knowledge of the distribution of the new value after the computation in question [52]. Meta-level reasoning is distinguished from its counterpart object-level reasoning. Object-level reasoning is deliberation about external entities, e.g., considering which action to take, where meta-level reasoning is deliberation about internal entities, e.g., deciding whether it is worth deliberating about a specific action. If the universe of discourse is a game of chess, object-level reasoning might for example be concerned with which opening move to make and meta-level reasoning with deciding whether it is worth deliberating about which opening move to make. Russell gives the following definition of meta-reasoning [53]:

Definition: *Meta-reasoning is any computational process that is concerned with the execution of other computational processes within the same agent."*

Meta-reasoning serves two important purposes in an intelligent agent [54][55]. Firstly, it gives the agent control over its (object-level) deliberation. Secondly, it increases the flexibility of the agent in the way it enables the agent to recover from errors in its object-level deliberation.

An important meta-level reasoning is a commitment strategy, which is an important aspect of agents is achieving a rational balance between deciding what to do (deliberation) and deciding how to do it (means-ends reasoning) [56]. This is simply because real agents are resource-bounded and the task-environment of real agents is always real-time. This means that instead of deliberating indefinitely and making optimal choices, the agent can only deliberate for a limited period of time and must settle with satisfying decisions. In the literature, this rational balance is also known as deliberation control [53]; inference control [44]; or deliberation and action trade-off [55]. In the BDI literature, the rational balance has been treated as intention commitment [56] or intention reconsideration [57].

2.10. Bold and Cautious agent

In our experiments in the next chapter, we explain the relationship between the rate of world change and effectiveness, commitment, and planning time. Effectiveness is in an inverse relationship with the rate of world change. The higher the rate of world change with the less effective. To solve the problem, a reconsideration strategy came, the agent if it reconsidered his plans frequently in proportion to with the increase in the rate of world change, the effectiveness of the agent becomes higher than the agent who commitments to his plans, with this adopted strategy of reconsideration the results may be excellent, but when the cost factor is founded, there will be a change in the scene. An agent used reconsideration in its current plan after a number of steps or more. Two agents will go in our project, the bold and the cautious agent. The agent decides which hole to visit considering the distances and the no. of the holes.

The bold agent, which never pauses to reconsider intentions until its plan is completed [58-60], see Algorithm (2.1).

Algorithm	Bold Agent
Input	B: =Bo; /* Bo are initial beliefs */ I: = Io; /* Io are initial intentions */
output	π:= plan (B,I)

```
while true do
        get next perception p;
                B: = brf(B,p);
                D: = options (B,I);
                I: = filter (B,D,I);
                π:= plan ( B,I);
        while not(empty(π)or
    succeeded(I,B) or impossible (I,B) do
        α: = hd (π);
        execute (α);
        π: = tail (π);
        get next percept p;
        B: = brf (B,P);
        if not sound (π,I,B) then
                π: = plan (B,I)
        end-if
    end-while
end-while
```

Algorithm (2.1): Bold Agent Algorithm

A cautious agent, which stops to reconsider intentions before performing any action [51], See Algorithm (2.2).

Algorithm	Cautious Agent
Input	B: =Bo; /* Bo are initial beliefs */ I: = Io; /* Io are initial intentions */
output	π:= plan (B,I)

```
while true do
        get next perception p;
                B: = brf(B,p);
                D: = options (B,I);
```

```
                I: = filter (B,D,I);
                π:= plan ( B,I);
        while not(empty(π)or
    succeeded(I,B) or impossible (I,B) do
        α: = hd (π);
        execute (α);
        π: = tail (π);
        get next percept p;
        B: = brf (B,P);
        D: = options (B,I);
        I: = filter (B,D,I);

        if not sound (π,I,B) then
                π: = plan (B,I)
        end-if
    end-while
end-while
```

Algorithm (2.2): Cautious Agent Algorithm

2.11. Tileworld Testbed

Pollack and Ringuette report on the Tileworld test bed in [12-14], an unexpected dynamic system that provides controlled, observable, and traceable experiences built for adaptive agent architecture. This environment is an incubator for the agent, tiles, obstacles, and holes in its two-dimensional grid profile.

A subjective expected utility (SEU) proceeds, that represents the evaluation method for agent behavior in the Tileworld testbed in different levels of dynamism and commitment. When addressing the option to fill a hole gives an estimate by means of calculations in which the time available to fill the hole and the distance between the agent and the hole, the distance between the agent and the tiles, and the size of the hole, this metric determines the success of the strategy of the agent by filling more holes:

$$SEU = \frac{\text{Score (h)}}{\text{distance(a,h)}+tileavail(h)} \quad \dots (2.2)$$

The distance between one side and another on the network is represented by the distance (x,y) function, and the compensation for y is for the location of the agent (a) and instead of x for the location of the hole(h), while the tileavail(h) function is the distance between a tile and its hole:

$$tileavail(h) = \sum_{i=1}^{n} 2 * \text{distance}(\text{ti}, \text{ h}) \ldots (2.3)$$

In equation (2.3), the calculation of the distance is multiplied by2, because of the agent's round trip, the distance to the tile, and then it pushes the tile another distance into the hole. SEU does not take into account the time remaining to close the hole.

Table (2.3): SEU Comparing SEU Evaluator with Simple, and Human[12].

Agent	SEU	SEU/serial	Simple	Simple/serial	Human
normal speed	396	353	347	291	468
10x faster	256	234	183	152	3

The differences here are quite apparent. In the normal speed environment, the human subject performed best. This resulted from his having more sophisticated planning capabilities than the robot agent. But in a faster environment, the human's planning "tricks" were insufficient, and he could not keep up with the pace of change. Also, from the above experiment, we can see in a normal speed environment, the human subject performs best, which represents full scorers. So, full scores in Tileworld at each trial regardless of speed are 468 scores.

Note the below Table (2.4), that have results of SEU evaluator in normal speed and 10 faster with different levels of threshold that represent commitment levels.

Table (2.4): The Experiment by Tileworld testbed [12].

Threshold	-100	-75	-50	-25	0	25	50	75	100
Normal speed	396	413	393	409	404	398	388	381	371
10x faster	256	265	264	241	251	233	255	251	266

Each trial doing by Tileworld testbed must have specific characteristics, if the experimenter does 30 trials, it needs to set all characteristics, can constant some of them as the threshold on -100, and do 30 trials with objects (holes, obstacles, tiles, …) have various values are determined range of it, where the simulator can select a random value within a range.

Table 2.5: thirteenth of trials in Tileworld testbed

	Threshold (-100)				
	Object	Lifetime	Score	Size	Scores
Trial 1	Holes	[20, 50]	[80, 100]	[40, 50]	398
	Obstacles	[20, 40]		[35, 60]	
	Tiles	[20, 50]		[60, 70]	
	Tiles store	[40, 70]		[80, 90]	
Trial 2	Holes	[30, 50]	[100, 120]	[10, 50]	390
	Obstacles	[30, 40]		[20, 60]	
	Tiles	[20, 60]		[70, 90]	
	Tiles store	[30, 70]		[40, 100]	
…….	….	….	….	….	……
	….	….		….	
	….	….		….	
	….	….		….	
Trial 30	….	….	….	….	…….
	….	….		….	
	….	….		….	
	….	….		….	
Average					396

Table (2.5) is a background for each number in Table (2.4) but they are approximate values; the purpose of including them is clarification.

Therefore, this study focuses on proposing a testbed for intelligent agent architecture that can address the evaluation process issues associated with the proposal platform. As explained in the next chapter and chapter four presents the experimental results of the proposal platform

Chapter 3- The Proposed Model

(Gridworld Testbed)

Chapter 3

The Proposed Model (Gridworld Testbed)

3.1. Introduction

An important meta-level reasoning is a commitment strategy, which is an important aspect of agents is achieving a rational balance between deciding what to do (deliberation) and deciding how to do it (means-ends reasoning). This is simply because real agents are resource-bounded and the task-environment of real agents is always real-time. Also, deliberation is a computationally expensive operation, so a BDI agent should only deliberate when necessary this requires suitable intention reconsideration strategy. This means that instead of deliberating indefinitely and making optimal choices, the agent can only deliberate for a limited period of time and must settle with satisfying decisions.

The proposed model, we call it Gridworld, cares with reconsideration strategy for each agent and it calculates the effectiveness for each agent depending on its own reconsideration strategy.

The design of a Gridworld testbed consists of a simulated robot agent and simulated environment which is both dynamic and unpredictable. Both the agent and the environment are highly parameterized, enabling one to control certain characteristics of each. We can thus experimentally investigate the behavior of various meta-level reasoning strategies in different environments tuning environmental parameters. Grildworld testbed enables us to control the dynamic while giving an idea of the observation cost that the agent takes and there is a threshold that the agent does not cross, that threshold is among the accounts of the agent that gives it a balance between deliberation and execution. The time limits of the testbed environment force the agent to estimate the time to reconsider its environment with variable goals. All of the above are in the field of real-time search and our research has a close relationship to real-time search that integrates planning with execution. It is necessary to clarify the relationship between the agent and its environment, and the requirements of meta-reasoning, as the agent who has belief, desire and intention (BDI)

We will investigate the policy of reconsideration strategy for an agent in an environment with the following factors:

1. Dynamism: the percentage of change in the environment, and it is independent of the agent's activities.
2. Accessibility: The extent to which the agent has reached the state of the environment.

3.2. Gridworld Testbed

Gridworld: a grid network like a chessboard with agents and holes, the agent has the ability to move vertically and horizontally, if the hole is visited by the agent, the holes disappear. Visiting a hole means that the agent gets a score, the agent's goal is to get the most points by visiting the holes before closing.

There is a dynamic simulation in the GW based on parameters that generate a random state that changes over time.

The parameters determine the appearance and disappearance of the holes while the agent moves towards the holes. The adoption of a dynamic simulation system is to study and reconsider planning. See Figure 3.1 that gives an idea for GW.

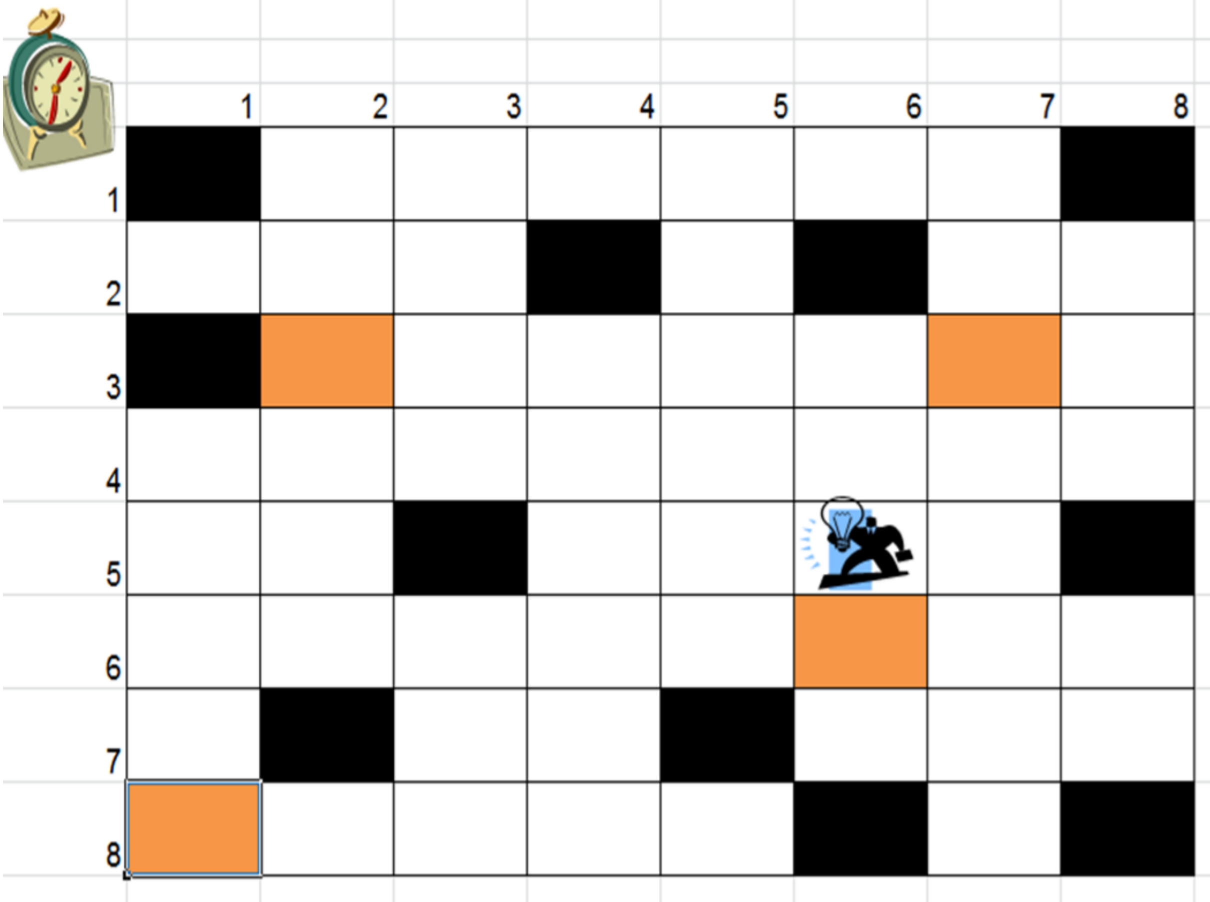


Figure (3.1): Gridworld (GW)

An agent can be likened to a postman who delivers messages in different locations, where the locations are the holes, the grid is the roads, and the time-vary simulates real-time. The agent's task is to perform the necessary efficient actions after reasoning, but reasoning imposes a cost on the agent related to changes in the world over time that cannot be predictable. Figure (3.2) shows the GW structure and Algorithm (3.1) shows the GW algorithm.

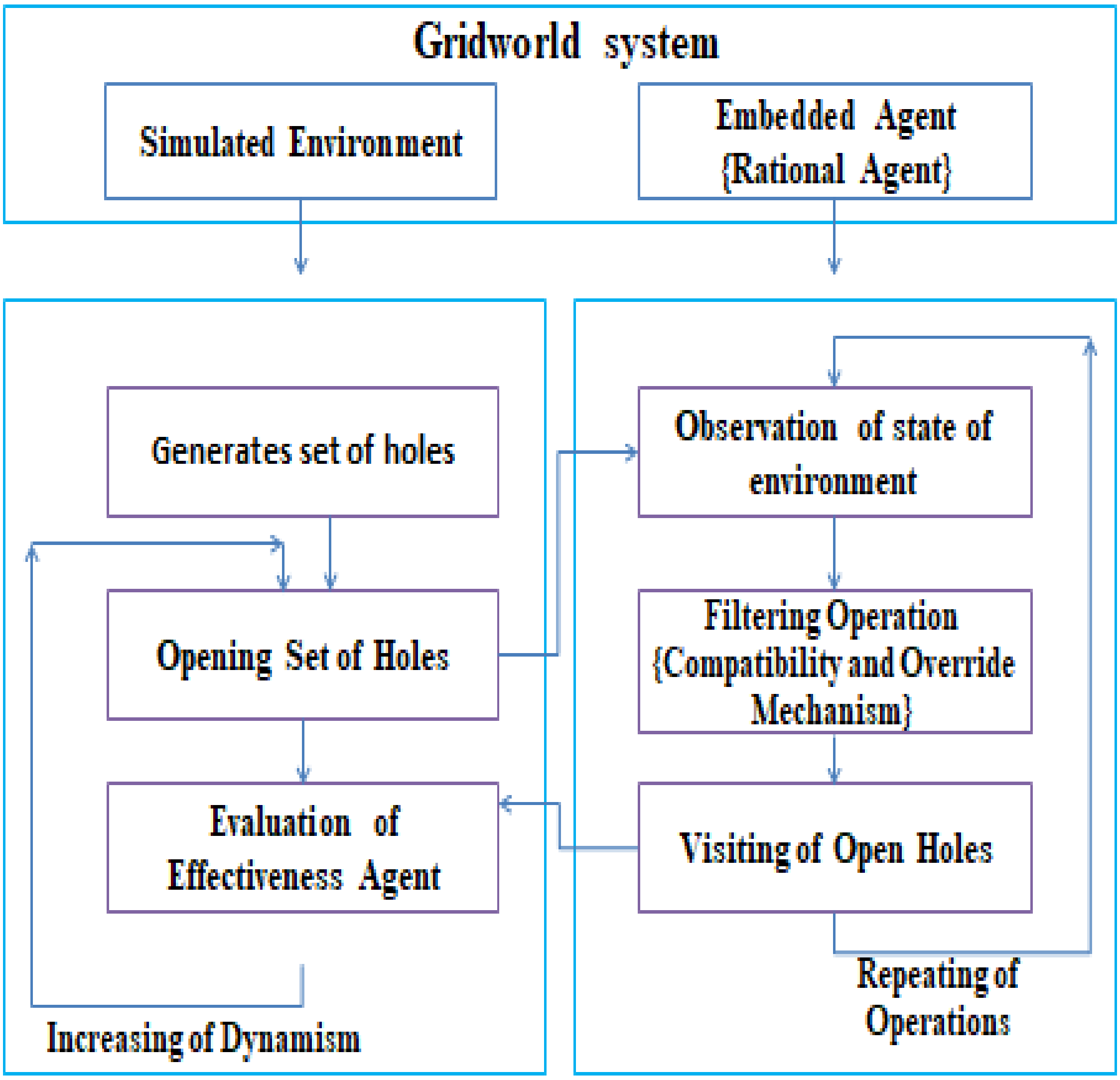


Figure (3.2): Gridworld Structure

Algorithm:	Simulated Environment and Embedded Agent
Input:	Current dynamism and maximum dynamism; level of commitment; value of holes and speed of agent (optionally)
Output:	Effectiveness of Agent

```
1:   while loop
2:       Generate multi random constants used as a lifetime of holes
3:       Generate multi random constant used as a gestation time
4:       Appear and disappear holes
5:       Agent
6:         While
7:           Observe GD
8:           If plan (i) has
               Goal && Threshold
9:               Visiting of Opening holes
10:              Go-to step 8
11:          Else
12:            Go-to step 8
13:        End while loop
14:      Calculate effectiveness
15:  End while loop
```

Algorithm (3.1): Gridworld (GW) Algorithm

Monitoring the changing environment, monitoring current plans, and making comparisons according to benefit and cost are tasks imposed on a rational agent in its decisions, must there is a tradeoff. Time pressure is one of the challenges that the agent faces when competing goals.

Revealing the agent feature necessitates providing a dynamic environment that is gradual in its complexity, so we set control for the dynamism and the level of gradual complexity, controlling the appearance and disappearance of holes, as for the agent, it may be fully or partially observing for its environment, depending on current state circumstances (rate of dynamism, beliefs, …) that simulates the real world with changing its goals over time, and it may be adopting a strategy to adhere to its plans or reconsider them.

See Figure (3.3) is a snapshot of the GW system before process of running to begin testing.

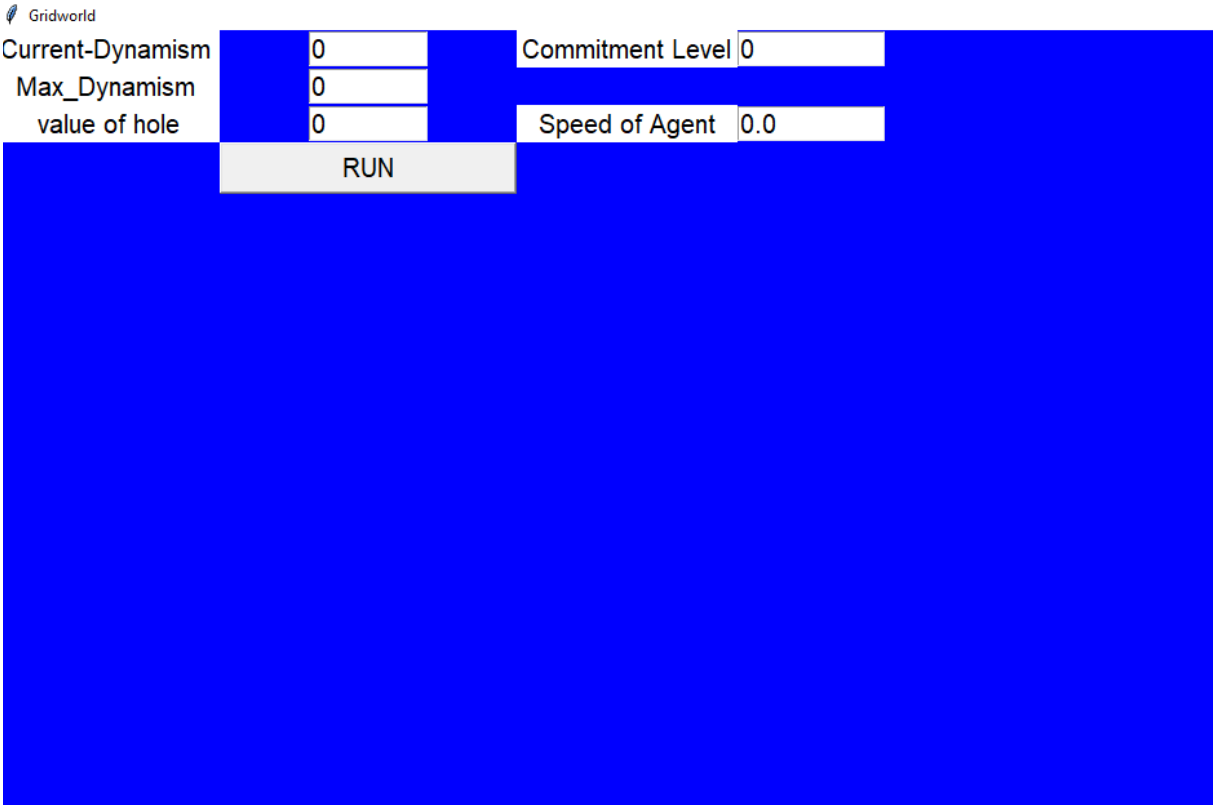


Figure (3.3): Interface of GridWorld Testbed

Figure (3.3) showed the interface of GW before running the test process. We find that there are a number of parameters that the experimenter has to enter into the system and then press the running button so that the window appears in a new form containing the results, as in Figure (3.4). The task entrusted to the experimenter is to enter a primitive value of the dynamic at the location indicated by the current dynamism, for example, starts with current dynamism 1 or 5 and so on, provided that the value is an integer. Entering the highest dynamic value that the system reaches in the testing process is in the location indicated by the Max-Dynamism, this value must be greater or equal to the primitive value, in addition to this value being an integer. The primitive value with the operation of the system begins to increase gradually to increase the dynamism of the system (the appearance and disappearance of holes) until it reaches the highest value entered by the experimenter.

In the interface, the value of the holes, entering this value is optional and takes values from 1 to 5 according to the design of the system (it can be changed), if the experimenter enters the value number 1 this means that all the holes have one value. But if the experimenter enters the numbers 2, 3, 4, or 5, the values of the holes will be subject to a uniform distribution, where each hole has the same opportunity to take a value from those values. That is useful when agents frequently store values for holes and have a history to review them before starting to visit them.

The level of commitment as it appears in the interface takes a value from nine values confined between (0 and 16), the experimenter is the one who determines this value by entering it according to the experiments that he performs, but the system does not have the control to enter this value or change it, knowing that this value is An image of a threshold will be explained in the next chapter.

The speed of agent is one of the optional parameters in the system that enables the experimenter to increase or decrease the agent's speed of movement according to his vision. A given value is greater than one means an increase in speed, while less than one means a decrease in speed. In system adopted the compatibility between the speed of the agent and the system because of our interest in meta-reasoning and its impact on the effectiveness of the agent.

Figure (3.4): Interface of GW testbed after running

The new in the interface after the process of running, by pressing the button of agent for putting the agent in the experiment and according to the parameters that have been set, that the test results begin to appear, and periodically. We will start describing those results in sequence from the top to bottom:

The cost of reconsideration (Cost Recon.), here shows the cost of reconsideration that the agent may need with the speed of change of the environment and this result is greatly affected by the parameter of the level of commitment.

The cost of transition (Cost Tran.) is the cost of the agent's movement from one row to another, and it is affected by the presence of the largest number of holes in the row among the rest of the rows, as this row is preferred by the agent.

The cost of observation (Cost Obser.), when the dynamics of the system change and the speed of change in the environment increases, the cost of observing the environment with its changes increases on the agent's shoulders, for example when dynamic 15 the cost of observation is not as in dynamic 20, here the lost opportunities abound rapidly changing environment.

The effectiveness, after the agent finishes his trial, the result will appear with his efficiency in counting the number of holes visited divided by the total number of holes.

3.3. GridWorld Characteristics

- GridWorld testbed has a dimension (row [0, 10] and column [0, 10]), changing of dimensions of GridWorld testbed are by change of holes' appear, where the agent may be observed matrix in a certain location at a certain time have 2 × 2.
- Hole scores are the same value for all cells, can be changed the value of holes by another parameter in other experiments.
- Hole life expectance(L): is a hole age before it disappears, it's chosen from a uniform distribution for [1, 3.03165×10^{-12}]
- Hole gestation time (g): here the interval is [10, 1.81899×10^{-11}], chosen by uniform distribution, is the interval between two successive appearances of the hole.
- Changing time: no. of changing the GW in a certain time [1.5, 9.42439×10^{11}], chosen by a uniform distribution.
- Dynamism(γ): changes in the interval [1, 40], can increase the dynamism in other experiments.
- No. of holes that are appeared in dynamism [1, 40] have a range [160, 8.79609×10^{13}]
- Reconsideration rate or Commitment level [0 - 16]: extreme bold agent represented by 16 and extreme cautious agent extreme represented by 0.
- Agent speed is optional for future trials.

Before we go deeper with the previous parameters and functions, the next paragraph will review the using of uniform distribution, for more understanding of the change that happened in our simulated environment.

We briefly review the concept of uniform distribution and its connection to our GW testbed. A uniform distribution describes a structure of probability distribution. Every possible consequence has an equal likelihood of accruing. The probability is a constant value since each variable has equal chances of being the consequence. In light of the type of possible outcomes, the types of uniform distribution are discrete and continuous[59].

As seen, the state of the hole in GW is continuo uniform distribution because its change in either appear or disappear in the interval, example in dynamism (γ) the lifetime of the hole (L) is variable randomly between the interval [x, y], between x and y infinite number, if x = 1 and y = 2 the L maybe 1 or 1.1 or 1.01 …infinite number. While the value of hole (score) is discrete uniform distribution if the experimenter did the choice of a multiplicity of values for the holes, then the value of hole has fixed value in the interval [1, 2, 3], the hole may have 1 or 2 or 3 scores only of them, in equal chance(probability) with other holes.

The hole value is subject to a discrete uniform distribution, it opens the way for our testbed in the future to test agents in reinforcement learning scope. Reinforcement learning is the agent that starts its movement at random, since it does not have predictions about its environment, during the iteration process the learning takes place slowly, and depends on history, resulting in the prediction of reward concepts [62-64].

3.4. Architecture and Strategy of GW Agent Embedded

The general agent architecture is based on Bratman [65], which asserts that the situated agent in a dynamic environment makes use of alternative plans for the current situation. The work of the situated agent is centered on the principle of means-ends-reasoning, the agent develops its current plans, the agent's plans are limited by the available options, and those options are filtered to achieve the goal.

Also, the terms ‘Agent’ and ‘environment’ are coupled so that one cannot be defined without the other. In fact, the distinction between an agent and its environment is not always clear, and it is sometimes difficult to draw a line between them [66].

The rational agent does not always commit to the current plan, the plan needs to be reconsidered based on environmental change, and the reconsidered plans have policies and are not continuous in all circumstances. In other words, the current state determines whether to reconsider or not. Reconsideration has a cost; the agent observes possible options and filters them to choose the best one. Observation consumes time and filtering as well.

The main structure of the reconsideration process is responding to options and is the filtering process, the filtering process imposes its first condition that the new choice is compatible with the current plans, and its second condition is that it suspends the current plans after comparing them with the possible options [61].

The suspension process is expressed as the override mechanism, the override mechanism is used by the rational agent without excessive and moderately, reconsideration of the plan in every change and for every instant (unexpected) event, the agent gets out of the moderate or possible calculations, at the same time insufficiently sensitive to the environment results in lack of response to obvious deviations.

The deliberative process is the make decision factory, the filtering mechanism exports the decision problem framed within specific options, the actions toward the agent's updated intentions is the deliberative process. Now we can conclude that we have a process of deliberations that determines the options that are obligatory to follow based on the updated intentions. As for means-ends-reasoning, it is the mechanism for determining how to achieve the goals, in other words exporting options to the deliberation process to be a subject for deliberation and make a decision.

In our model is an agent that works in two cycles: the act cycle and the reasoning cycle after observing dynamic environmental. The act cycle implements the formulated and approved plans, and the reasoning cycle monitors the environment and its changes and presents the best options to be alternative plans. The reasoning cycle has a computations process, in other words, there is cost time. The agent can observe its current position, the entire world map, and the locations of its objects.

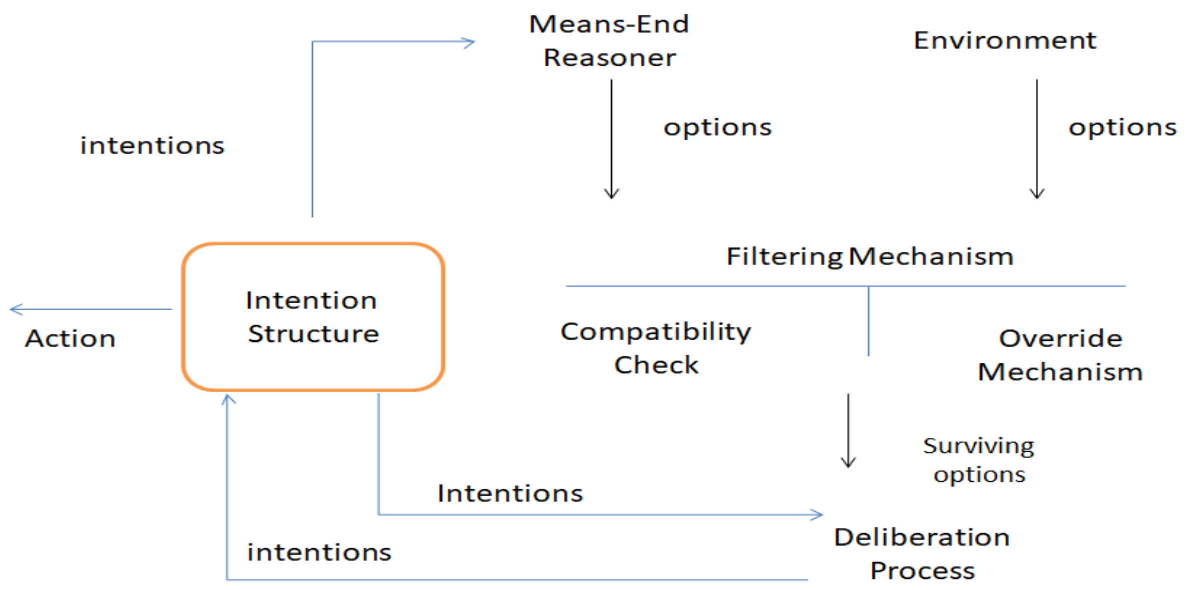


Figure (3.5): Structure Reconsideration of Agent

The reasons for the new options to be considered by the agent are by two reasons, the first: the agent observes changes when holes appear, second: means-end reasoner offers options such as changing the agent's location to visit more holes, see Figure 3.5. We can see the similarity with IRMA structure(Intelligent Resource-bounded Machine) [65].

Options are subject to filter, the filtering mechanism offers the option compatible with the agent's current plan. For example, to fill the holes in row R1 at the current time, and offering the option to fill the holes in row R2 will be incompatible with the current intention at the current time here will depend on the override mechanism.

To apply the overriding strategy, setting a threshold level on the constant t, a decrease in its value means the new position outnumbers the old position in the number of open holes and this motivates the agent to consider its plans.

However, if the number of holes (M) in row R2 minus the number of holes (N) in row R1 the result maybe trigger the override mechanism after comparing with t, then deliberations (reconsideration) will occur in the next step, as in equation (3.1).

$$M - N\ (\geq)\ t \longrightarrow \text{DELIBERATIONS} \qquad \ldots (3.1)$$

It is important to introduce some terms from Bratman: If the agent is very sensitive to changes in the environment and is willing to reconsider its plans in response to a set of events, it is called a cautious agent, while the agent who commitments to his plan, and it turning a blind eye during executed its plan, it is called a bold agent for more details see section (2.10).

The Gridworld agent, when the value of the threshold decreases, becomes more cautious, the options issued by the filtering mechanism (surviving options) are subject to deliberation.

Assuming that option M survives. It is necessary to have a parameter that determines whether this intention will be adopted to fill row R2 or to keep the current plan in filling row R1.

In other words, there is a need to estimate the results of the options surviving the filtering process, whether its adoption achieves the desired results of the agent (high scores) or not. When the threshold is negative for the override mechanism, the new options will pass to the deliberation mechanism to present the best option; the option to be approved by the agent, a function that guesses the results of that option exists within the deliberation structure.

The next chapter will introduce our experimental results and compare them with previous works.

Chapter 4- Experimental Results & Evaluation

Chapter 4

Experimental Results & Evaluation

4.1. Introduction

Based on what we suggested in the previous chapter of presenting a testbed with specifications and usage mechanisms that were clarified in detail. In this chapter, we present an experiment that evaluates reconsideration strategies and the degree of commitment to immediate plans by agents included within the system and gives clear results on the agent's interaction with the dynamic environment at several levels of commitment (different strategies).

In addition to reviewing the results, they are compared with previous experiences, to note the reliability of the results of our proposal outperformed almost the results of other proposals.

4.2. Experiment of Commitment

The experiment is defined by a set of experimental conditions and the number of trials to be conducted for each condition. A trial is a single run of the GW system. A trial is defined by its duration and its experimental condition characteristics. Although, it is possible to interrupt a trial and change the environmental characteristics.

Our experiment used a design with two factors: degree of commitment for which we had 9 levels, and degree of dynamism, for which we had 9 levels.

Degree of commitment refers to the extent of the reconsideration strategy: the more committed the agent, its override threshold is high and hence less likely it is to reconsider its current intentions in light of new options. In our experiments, override threshold values ranged from -100 to 100, in increments of 25. And the agent with an override threshold of 100 was extremely bold, while the agent with an override threshold of -100 was extremely cautious. Note Table (4.1) describes the results of our experiments.

Degree of dynamism refers to the average rate of change in the environment: how frequently, on average, do exogenous events occur? A more rapidly changing environment is one with a greater degree of dynamism. Having established a baseline rate of dynamism, we used 9 levels for this factor: 1, 5, 10, 15, 20, 25, 30, 35, and 40. We do not increase the dynamism of the GW testbed because of the characteristics of pc.

The independent parameter in our experiments was effectiveness, which is a normalized measure of the agent's score: specifically, the scores that are the agent actually received divided by the total scores that were available for the agent to gain during the trial. Thus effectiveness can vary between [0, 1]; it will be less than 1 whenever the agent fails to satisfy some goals.

Figure (4.1) shows a picture of using the system and presenting the results after conducting the experiment and Table (4.1) represents the results for these experiments with levels of dynamism and commitment. For all results see Appendix 1 in this thesis.

Figure (4.1): Snapshot for experiment results in dynamism 1 and commitment 10 in GW.

Table 4.1: Agent Experiments with Multi-Dynamism and Variety of Commitments Levels in Gridworld.

Dynamism level	degree of commitment agent by its goals	override threshold	Cost of Reconsideration	Cost of Transition	Cost of Observation	Effectiveness with a Cost	Mean Effectiveness and Mid Effectiveness
1	16 (extreme bold)	100	0	0.02	0.01	0.92	0.92
	14	75	0	0.01	0.01	0.95	
	12	50	0	0.02	0.01	0.94	
	10	25	0	0.06	0.01	0.89	
	8	0	0.005	0	0.01	0.9	0.9
	6	-25	0.005	0.2	0.01	0.63	0.83
	4	-50	0.005	0.01	0.01	0.93	
	2	-75	0.005	0.06	0.01	0.89	
	0 (extreme cautious)	-100	0.005	0	0.01	0.9	
5	16 (extreme bold)	100	0	0.04	0.05	0.81	0.74
	14	75	0	0.02	0.05	0.71	
	12	50	0	0.18	0.05	0.72	
	10	25	0	0.17	0.05	0.73	
	8	0	0.025	0.08	0.05	0.79	0.79
	6	-25	0.025	0.08	0.05	0.72	0.65
	4	-50	0	0.22	0.05	0.67	
	2	-75	0.025	0.17	0.05	0.7	
	0 (extreme cautious)	-100	0.04	0.33	0.05	0.53	
10	16 (extreme bold)	100	0	0.26	0.1	0.61	0.63
	14	75	0	0.2	0.1	0.65	
	12	50	0	0.2	0.1	0.64	
	10	25	0	0.21	0.1	0.65	
	8	0	0	0.12	0.1	0.76	0.76
	6	-25	0	0.13	0.1	0.74	0.63
	4	-50	0	0.22	0.1	0.67	
	2	-75	0.012	0.18	0.1	0.6	
	0 (extreme cautious)	-100	0.06	0.24	0.1	0.53	

Dynamism level	degree of commitment agent by its goals	override threshold	Cost of Reconsideration	Cost of Transition	Cost of Observation	Effectiveness with a Cost	Mean Effectiveness and Mid Effectiveness
15	16 (extreme bold)	100	0	0.2	0.15	0.57	0.48
	14	75	0.075	0.27	0.15	0.3	
	12	50	0	0.2	0.15	0.57	
	10	25	0	0.25	0.15	0.49	
	8	0	0	0.15	0.15	0.63	0.63
	6	-25	0.075	0.19	0.15	0.46	0.38
	4	-50	0.1125	0.24	0.15	0.28	
	2	-75	0.1	0.25	0.15	0.43	
	0 (extreme cautious)	-100	0.128	0.27	0.15	0.36	
20	16 (extreme bold)	100	0	0.21	0.2	0.42	0.4
	14	75	0	0.25	0.2	0.36	
	12	50	0	0.23	0.2	0.43	
	10	25	0	0.27	0.2	0.41	
	8	0	0	0.22	0.2	0.45	0.45
	6	-25	0.15	0.31	0.2	0.39	0.31
	4	-50	0.13	0.28	0.2	0.24	
	2	-75	0.13	0.14	0.2	0.39	
	0 (extreme cautious)	-100	0.15	0.33	0.2	0.22	
25	16 (extreme bold)	100	0	0.26	0.25	0.33	0.29
	14	75	0	0.15	0.25	0.44	
	12	50	0.125	0.36	0.25	0.1	
	10	25	0	0.36	0.25	0.3	
	8	0	0	0.25	0.25	0.43	0.43
	6	-25	0.125	0.08	0.25	0.21	0.27
	4	-50	0.125	0.27	0.25	0.25	
	2	-75	0.187	0.14	0.25	0.32	
	0 (extreme cautious)	-100	0.208	0.11	0.25	0.33	

Dynamism level	degree of commitment agent by its goals	override threshold	Cost of Reconsideration	Cost of Transition	Cost of Observation	Effectiveness with a Cost	Mean Effectiveness and Mid Effectiveness
30	16 (extreme bold)	100	0	0.28	0.3	0.25	0.24
	14	75	0	0.24	0.3	0.38	
	12	50	0.15	0.2	0.3	0.2	
	10	25	0.15	0.24	0.3	0.15	
	8	0	0	0.28	0.3	0.34	0.34
	6	-25	0.2	0.17	0.3	0.18	0.13
	4	-50	0.15	0.24	0.3	0.17	
	2	-75	0.25	0.39	0.3	0.18	
	0 (extreme cautious)	-100	0.26	0.26	0.3	0.01	
35	16 (extreme bold)	100	0	0.26	0.35	0.27	0.23
	14	75	0	0.19	0.35	0.44	
	12	50	0.175	0.29	0.35	0.02	
	10	25	0.175	0.11	0.35	0.21	
	8	0	0	0.29	0.35	0.3	0.3
	6	-25	0.175	0.21	0.35	0.12	0.08
	4	-50	0.23	0.23	0.35	0.09	
	2	-75	0.28	0.28	0.35	0.07	
	0 (extreme cautious)	-100	0.3	0.3	0.35	0.04	
40	16 (extreme bold)	100	0	0.29	0.4	0.2	0.21
	14	75	0	0.12	0.4	0.3	
	12	50	0.01	0.09	0.4	0.22	
	10	25	0.2	0.36	0.4	0.14	
	8	0	0.26	0.39	0.4	0.27	0.27
	6	-25	0.3	0.12	0.4	0.02	0.05
	4	-50	0	0.44	0.4	0.09	
	2	-75	0.33	0.17	0.4	0.04	
	0 (extreme cautious)	-100	0.32	0.29	0.4	0.08	

When examine the information in Table (4.1), can conclude the following notes and results:

1. In dynamism phase number one, the agent was tested within nine levels of commitments, where each level represents a specific commitment to

the agent with its plan. The nine levels range from the highest commitment, which is represented by the bold agent, to descend to lower and lower until the commitment level reaches the lowest levels and is represented by the cautious agent. This multi-commitment test process is repeated at the other phases of the dynamism.

2. The highest of commitment, the highest of the value of the threshold t (positive value), in other words, the processes of deliberation and the presentation of options decreases. While the lowest value of commitment towards the lowest value of threshold t (negative value), thus the more need to reconsider, filtering, deliberation, and the best options.
3. It is very clear that the need for reconsideration increases and appears in the table in the stages of low commitment with an increase of dynamism. While in the early stages where commitment is high with low dynamism, we find reconsideration value is zero.
4. The average value of each dynamism phase in which there is a state of moderation between high commitment and low commitment, so the agent's performance is at the best possible compared to the strict values at both ends of the same dynamic phase. This is similar to the EBDI agent (Extensible, Believe, Desire, and Intention), which has a trade-off reconsideration strategy [67].
5. The average performance rate of the bold agent and the cautious agent was calculated to facilitate the comparison process with the results. In the last column. Three values: The first value is the effectiveness rate of the bold agent taken from four previous cells of the previous column (effectiveness). The second value is the value of the agent's effectiveness in the case of moderation, meaning that the review is carried out only in case of necessity. And the third value is the rate of

effectiveness of the cautious agent taken from the previous four cells of the previous column (effectiveness).

6. It is clear, in a manner appropriate to the results of previous experiments of former scientists in test environments, that the more cautious agent suffers from inefficiency for two main reasons: The first reason is that it works in the process of reconsideration extravagantly, thus placing more costs on it. And the second reason is that returning to the reconsideration process frequently misses the agent many opportunities to acquire points.

The emergence of these results demonstrates the success of GW testbed for investigation of behaviors of the agent embedded in a dynamic environment if it compared with previous researches as what we mentioned. We believe some noise in the results comes from the vagaries of the machine and software system. We will need to execute more trial sets to decrease that and need a model to smooth the noise. And this is one of the main reasons behind did not execute the testbed in more than 40 dynamism. During doing multi-testing in setting similar parameters we note some results from one group have little difference.

Furthermore, Table 4.2 is a simplification of Table 4.1: it shows the performance of the most committed, the least committed, and not extremely agents in our experiments. As we mentioned above, the most committed agent had a threshold of 100, which is equal to the maximum full score associated with a hole in this experiment. The most committed agent is thus extremely unlikely to reconsider its current intention in light of deliberation options. The least committed agent had a negative threshold, and thus frequently deliberated about options.

Table (4.2): A Brief of Comparing between three types of agent's behavior

Dynamism	Bold Agent	Cautious Agent	Agent(not extreme)
1	0.92	0.83	0.9
5	0.74	0.65	0.79
10	0.63	0.63	0.76
15	0.48	0.38	0.63
20	0.4	0.31	0.45
25	0.29	0.27	0.43
30	0.24	0.13	0.34
35	0.23	0.08	0.3
40	0.21	0.05	0.27

Table 4.2 gives the results in a summary form that makes it easy for the reader to notice the results we have presented, where the cautious agent's efficiency declines with increasing dynamism for the reasons we are mentioned, while the agent who is not extreme in his commitment for plans outperforms both agents in all dynamism phases by trade-off commitment.

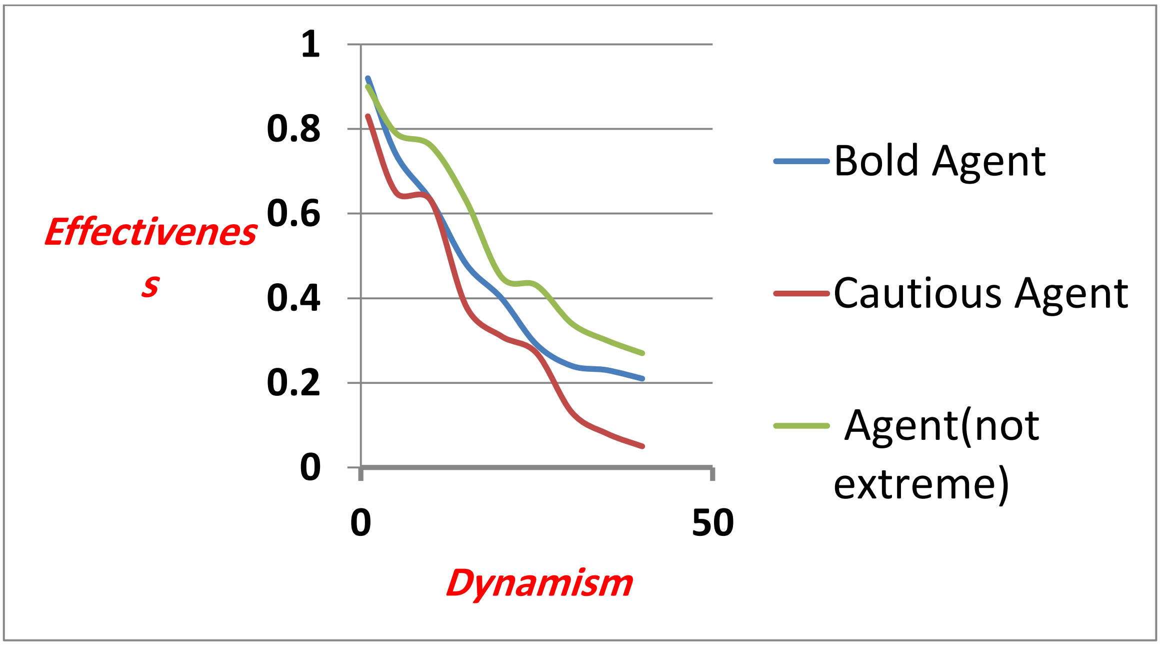


Figure (4.2): Plot the Effectiveness of Three types of Agents

Figure (4.2) plots the degree of dynamism on the x-axis, but it shows the difference between the most committed agent's effectiveness, the least committed agent's effectiveness, and the not extreme agent, all these on the y-axis.

One can see from this graph that the value of commitment is a function of the degree of dynamism in the environment. As dynamism increases, the marginal value of commitment first increases, in slower worlds, there are fewer options presented to the agent, and, hence, fewer opportunities for filtering to result in savings in reasoning cost. Moreover, the advantages of reducing reasoning are minimal, since there is generally enough time to deal with options. As the world becomes more dynamic, there are more options for consideration, and the penalty for extra reasoning increases because there is less time to respond to those options.

It is noticed with the high dynamics that the agent's attempts increase, so it cannot bear to waste the opportunities that it pursues in its environment while it is busy implementing its plan. Excessive counting of attempts in and of itself wastes opportunities with an increase in cost.

4.3. The Limits of Commitment

The experiment just described demonstrates that, in a wide range of GW testbed, agents perform better when they filter from consideration actions that are incompatible with their existing intentions.

In fact, it is surprising how to evaluate effective filtering proved to be in this experiment. Essentially, what the experiment shows is that at least under the range of experimental conditions studied, the agent that performed the best was always the agent who reconsidered its current goal only if necessary, we describe this agent as not extreme or having a trade-off mechanism.

Extremely bold behavior of this type is similar to what we might expect from a traditional AI system, which handles goals sequentially, always completing one task before turning to the next. Yet in principle, it seems unlikely that extreme boldness will generally lead to satisfactory performance. It seems natural to think that extreme commitment to existing goals would not be a good strategy because the high payoff opportunities if they are to be successfully acted on, require a quick response.

4.4. Performance Measurements

Pollack and Ringuette report on the TILEWORLD test bed[12], unexpected dynamic system that provides controlled, observable, and traceable experiences built for adaptive agent architecture. This environment is an incubator for the agent, tiles, obstacles, and holes in its two-dimensional grid profile.

A subjective expected utility (SEU) proceeds that represents evaluation method for agent behavior in Tileworld testbed in different levels of dynamism and commitment, for more details see section (2.10) in this thesis.

What we offer is the evaluator effectiveness agent (EEA), it is simple in calculation and high accuracy in the outputs in evaluating the efficiency of the agent that has a perceptual structure and the most important pillar is to reconsider the current plans.

$$EEA(agent) = \frac{1}{k}\sum_{n=1}^{k}\left(\frac{\text{Agent scores}}{total\ scores}\right) \quad \ldots (4.1)$$

Where score- agent refers to scores that are obtained by the agent during its trial, total scores are all holes are appearing. And the summation is used to cover a certain period of time.

A cost is the summation of cost transition, cost observation, and cost reconsideration of an agent, it can be subtracted from EEA (agent).

We neglected some of the vocabularies in our modulation on the (SEU) in calculating the effectiveness of agent, the tile is neglected and the equation for calculating the distance between the agent and the tile has no longer value, as for the holes, they are similar in their values (variable value of holes is optional), the agent’s trouble in the dynamic represented by the appearance and disappearance of holes with variable time periods in addition to the changing time period between two successive appearances of the hole.

Table (4.3) shows the comparison between the effectiveness of agent behavior in the Gridworld testbed and the Tileworld testbed. Where the effectiveness for Tileworld agent in Table (4.3) is calculated by dividing the scores already gained by the agent overall scores available in each trial, as we mentioned in section (2.11) in chapter 2. The full scores in the Tileworld testbed are 468 scores at any time regardless of speed. So, for our comparison, we need to find the effectiveness for scores in Table (2.4) to make a comparison with the Gridworld testbed by dividing gained scores in each trail over full scores (468).

. While, the effectiveness of the Gridworld testbed was gathered from Table (4.1) for dynamism 1 that represents normal speed, and dynamism 10 represent 10 faster normal speed, under different level of commitment strategies from -100 to 100.

Table (4.3): Effectiveness Comparison between GW and Tileworld Experiments.

Threshold	-100	-75	-50	-25	0	25	50	75	100
Normal speed (Agent Tileworld)	0.84	0.88	0.83	0.87	0.86	0.84	0.82	0.81	0.79
Normal speed (Agent GW)	0.90	0.89	.0.93	0.63	0.90	0.89	0.94	0.95	0.92
10x faster (Agent Tileworld)	0.54	0.56	0.51	0.51	0.53	0.49	0.54	0.53	0.56
10x faster (Agent GW)	0.53	0.60	0.67	0.74	0.76	0.65	0.64	0.65	0.61

As seen in Table (4.3) that represents the comparison our proposed system Grildworld and Tileworld in normal speed and 10 faster under different levels of commitment, the Girdworld outperforms Tileworld in most of the time.

To sum up the results of our experiments in Table (4.1), and the comparison process with Tileworld, we see that filtering is harmful at slow speeds, and even at high speeds does not give a net benefit. Our hypothesis is that the time cost of the meta-level reasoning of Grilworld agent is not very high, and consequently, it is usually worth taking the time to engage in extra deliberation about new opportunities. For these, filtering may be much more valuable; the EEA-estimator is efficient enough that it can itself be used as the filter override mechanism for the more complex deliberation components.

Chapter 5- Conclusions & Future Works

Chapter 5

Conclusions & Future Works

5.1. Introduction

The design of a Gridworld testbed is an environment for an agent has a strategy of combining reactive and planning. The testbed environment that we designed enables us to control the dynamic while giving an idea of the observation cost that the agent takes and there is a threshold that the agent does not cross, that threshold is among the accounts of the agent that gives it a balance between deliberation and execution. The time limits of the testbed environment force the agent to estimate the time to reconsider its environment with variable goals. All of the above are in the field of real-time search and our research has a close relationship to real-time search that integrates planning with execution. It is necessary to clarify the relationship between the agent and its environment, and the requirements of meta-reasoning, as the agent who has belief, desire and intention (BDI), the decision is made by the agent through two activities, the first of which is deliberation (setting the intentions that must be achieved), and the second is means-ends-reasoning (how to reach those intentions) in other words, how is the intentions achieved. Deliberation is a costly process, so the BDI chooses it only when necessary, which requires the adoption of a policy to reconsider intentions appropriately.

The research problem for this thesis was; to improve our understanding of the relation between agent design and environmental factors. In addition, what affects the agent's perception of its environment due to its own reason such as sensors, speed and time, or because of rapid changes around it. By this evaluating the degree of suitability of the agent to a given level of changes in the dynamic environment will show.

In other words; the research problem in this thesis was how to design and implement dynamic testbed platform in simple way for intelligent agent architecture such as situated agents, to evaluate agent's behavior under its rules for reconsideration strategy, and degree of commitment in its current plan.

The aims for this research were, to design and implement simulated environment testbed with embodied agent, which enables experimenters to test reconsideration strategies adopted by intelligent agents in dealing with unpredictable dynamic environments. This testbed can evaluate agents working with reconsideration strategies, which is, in short, a mechanism adopted by the agent that describes the degree of commitment to its plans with regard to changes in the dynamic environment.

The main objective of the thesis was to design simulation for dynamic environment consists of a grid of cells (squares) on which various objects can exist. These objects can be anyone of the following, agents, and holes. The agent (about which the experiment is based) can move up, down left or right. The agent's objective is fill the holes by reaching them. A hole has an associated point value which is awarded to the agent upon filling the hole. Once the hole is filled completely the agent gains the points. The agent knows how valuable each hole is in advance; its overall goal is to get as many points as possible. These simulations are dynamic; the environment changes continually over time. The objects appear and disappear at rates pre-determined by parameters in the simulator.

We can summarize the role of the design as follows:

1. Designing an environment with parameters that can be controlled to simulate the real environment.

2. It provides a metric of agent performance that is convenient to use, i.e. simple to calculate, yet sufficiently fine-grained to allow adequate discrimination of effectiveness.
3. A set of parameters that vary the interesting properties of the world. Ideally, the parameters would map to well-defined measurable properties of real environments.
4. Through this thesis, we stand on very important vocabularies in the composition of the intelligent agent, such as uncertainty; making decisions, deliberation, filtering, and so on.
5. The ability to generate worlds has large random numbers statistically. The random numbers control the state of the world.

In this chapter, we present the most important conclusions that clarified through the design and implementation of the Gridworld testbed, and a general description of the environment and the requirements for designing such works.

In addition, this chapter includes suggestions and possible future work for this thesis in a way that increases the efficiency of the system and expands its use for researchers.

5.2. Conclusions

The designing and implementation of Gridworld testbed for an intelligent agent led to several conclusions that we list below:

1. In this thesis we introduce an improved understanding of the relation between agent design and environmental factors. In the future, when faced with a performance domain for an agent, one should be able to draw on such an understanding to choose more wisely from the wide range of implementation possibilities available.

2. The Gridworld testbed has been demonstrated and has been shown to be an available system for evaluating agent architectures. A Gridworld testbed with a simplified that is easy to deal with, and qualified to be a platform for testers in that it does not require much effort to deal with. In addition, the Gridworld architecture is not just a simulated environment but has an embedded agent as well. Where the Girdworld testbed outperforms the Tileworld testbed most of the time.
3. The meta-level reasoning (reconsideration strategy) or (commitment strategy) is an important process that is reasoning over reasoning, with very low cost to determine wither the agent continue acting in the current plan, or going for deliberation to get alternative plan.
4. The independent parameter in our experiments was effectiveness, which is a normalized measure of the agent's score: specifically, the scores that are the agent actually received divided by the total scores that were available for agent to gain during the trail. Thus effectiveness can vary between [0, 1]; it will be less than 1 whenever the agent fails to satisfy some goals.
5. In the dynamic environments, reconsideration was still an effective meta-level control strategy: agents that always interrupted their current plans to deliberate about a new option did not perform well. However, in these environments extreme filtering is no longer optimal: instead, the agents whose performance was best were those that were sensitive not only to emergencies but also to especially promising options.
6. The Gridworld environment provides a dynamic that clearly affects the degree of commitment (the level of response to options and reconsideration of them).
7. The filtering is harmful at slow speeds, and even at high speeds does not give a net benefit. Our hypothesis is that the time cost of the meta-level reasoning of Grilworld agent is not very high, and consequently, it is usually

worth taking the time to engage in extra deliberation about new opportunities.

8. In this thesis we introduced EEA-estimator is efficient enough that it can itself be used as the filter override mechanism for the more complex deliberation components.
9. The experiments show is that at least under the range of experimental conditions studied, the agent that performed the best was always the one who reconsidered its current goal only if necessary, we describe this agent as not extreme or have a trade-off mechanism.
10. It is clear, in a manner appropriate to the results of previous experiments in test environments, that the more cautious agent suffers from inefficiency for two main reasons: The first reason is that it works in the process of reconsideration extravagantly, thus placing more costs on it. And the second reason is that returning to the reconsideration process frequently misses the agent many opportunities to acquire points.
11. The average performance rate of the bold agent and the cautious agent was calculated to facilitate the comparison process with the results. In the last column. Three values: The first value is the effectiveness rate of the bold agent taken from four previous cells of the previous column (effectiveness). The second value is the value of the agent's effectiveness in the case of moderation, meaning that the review is carried out only in case of necessity. And the third value is the rate of effectiveness of the cautious agent taken from the previous four cells of the previous column (effectiveness).
12. The highest of commitment, the highest of the value of the threshold t (positive value), in other words, the processes of deliberation and the presentation of options decreases. While the lowest value of commitment towards the lowest value of threshold t (negative value), thus the more need to reconsider, filtering, deliberation, and the best options.

13. In experiments of our proposed system in dynamism phase number one, the agent was tested within nine levels of commitments, where each level represents a specific commitment to the agent with its plan. The nine levels range from the highest commitment, which is represented by the bold agent, to descend to lower and lower until the commitment level reaches the lowest levels and is represented by the cautious agent. This multi-commitment test process is repeated at the other phases of the dynamism.

5.3. Future works

There are several proposals for future work that can be adopted to improve the system, increase its efficiency and expand its work, which we include as follows:

1. The adoption of additional design strategies within the Gridworld testbed qualifies the system to test other behaviors of the agent such as dealing with history and adding it to the awareness of a dynamic environment.

2. Adding a test mechanism for the agent's speed of movement in the presence of obstacles and override mechanisms for these obstacles to reach its goal.

3. Upload the system as a website platform.

4. Overcoming the physical challenges of a computer using a networked virtual computer for design such as system.

References

[1] T. L. Dean and M. S. Boddy, "An Analysis of Time-Dependent Planning.," in *AAAI*, 1988, vol. 88, pp. 49–54.

[2] S. J. Russell and E. H. Wefald, "Principles of metareasoning, InProceedings of the First International Conference on Principles of Knowledge Representation and Reasoning, RJ Brachman et al., eds." San Mateo, California (Morgan Kaufmann, 1989), 1989.

[3] E. J. Horvitz, "Reasoning about be-liefs and actions under computational resource con-straints," *Proc. 1987 l'l/orkshop Uncertain. Artificial Intell. Seattle, WA*, 1987.

[4] M. E. Bratman, D. J. Israel, and M. E. Pollack, "Plans and resource-bounded practical reasoning," *Comput. Intell.*, vol. 4, no. 3, pp. 349–355, 1988.

[5] S. J. Russell and E. Wefald, *Do the right thing: studies in limited rationality*. MIT press, 1991.

[6] A. B. Philips and J. L. Bresina, "Nasa tileworld manual (system version 2.2)," 1991.

[7] A. Raja and V. Lesser, "Meta-Level Control in Multi-Agent Systems."

[8] T. A. Montgomery and E. H. Durfee, "Using MICE to study intelligent dynamic coordination," in *[1990] Proceedings of the 2nd International IEEE Conference on Tools for Artificial Intelligence*, 1990, pp. 438–444.

[9] S. Yamada, "Controlling deliberation with the success probability in a dynamic environment," in *AIPS*, 1996, pp. 251–260.

[10] S. Hanks, M. E. Pollack, and P. R. Cohen, "Benchmarks, test beds, controlled experimentation, and the design of agent architectures," *AI Mag.*, vol. 14, no. 4, p. 17, 1993.

[11] T. Dean, L. P. Kaelbling, J. Kirman, and A. Nicholson, "Deliberation scheduling for time-critical sequential decision making," in *Uncertainty in Artificial Intelligence*, 1993, pp. 309–316.

[12] M. E. Pollack and M. Ringuette, "Introducing the Tileworld:

Experimentally evaluating agent architectures," in *AAAI*, 1990, vol. 90, pp. p183-189.

[13] M. Roshanzamir, M. Palhang, and A. Mirzaei, "Tasks Decomposition for Improvement of Genetic Network Programming," in *2019 9th International Conference on Computer and Knowledge Engineering (ICCKE)*, 2019, pp. 201–206.

[14] M. Alshehri, N. Reyes, and A. Barczak, "Evolving Meta-Level Reasoning with Reinforcement Learning and A* for Coordinated Multi-Agent Path-planning," in *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, 2020, pp. 1744–1746.

[15] H. Ben Ticha, N. Absi, D. Feillet, A. Quilliot, and T. Van Woensel, "A branch-and-price algorithm for the vehicle routing problem with time windows on a road network," *Networks*, vol. 73, no. 4, pp. 401–417, 2019.

[16] T. Montgomery, J. Lee, D. Musliner, E. Durfee, and D. Darmouth, "Y. So. MICE users guide," Technical Report CSE-TR-64-90, Dept. of Electrical Engineering and Computer …, 1992.

[17] S. Sankhyadhar and M. Pandey, "Test Beds for Distributed AI Research," in *Distributed Artificial Intelligence*, CRC Press, 2020, pp. 179–194.

[18] D. Hart and P. Cohen, "PHOENIX: A test bed for shared planning research," 1990.

[19] R. Sottilare, A. Graesser, X. Hu, and G. Goodwin, *Design Recommendations for Intelligent Tutoring System-Volume 5: Assessment Methods*, vol. 5. US Army Research Laboratory, 2017.

[20] D. Nguyen, S. Hanks, and C. Thomas, "The TRUCKWORLD manual," Technical Report TR 93-09-08, Dept. of Computer Science and Engineering …, 1993.

[21] S. Hanks and D. McDermott, "Modeling a dynamic and uncertain world II: action representation and plan evaluation," *J. Log. Comput.*, 1993.

[22] S. Hanks and D. McDermott, "Modeling a dynamic and uncertain world i: Symbolic and probabilistic reasoning about change," *Artif. Intell.*, vol. 66, no. 1, pp. 1–55, 1994.

[23] M. Asada, H. Kitano, I. Noda, and M. Veloso, "RoboCup: Today and tomorrow—What we have learned," *Artif. Intell.*, vol. 110, no. 2, pp. 193–214, 1999.

[24] M. Lees, "A history of the Tileworld agent testbed," *Sch. Comput. Sci. Inf. Technol. Univ. Nottingham, Nottingham*, pp. 2001–2002, 2002.

[25] G. J. Sussman, "A computer Model of Skill Acquisition. 133 p." Elsevier Science/North-Holland, Amsterdam, London, New-York, 1975.

[26] S. Minton, M. D. Johnston, A. B. Philips, and P. Laird, "Solving Large-Scale Constraint-Satisfaction and Scheduling Problems Using a Heuristic Repair Method.," in *AAAI*, 1990, vol. 90, pp. 17–24.

[27] P. Stone and M. Veloso, "Multiagent systems: A survey from a machine learning perspective," *Auton. Robots*, vol. 8, no. 3, pp. 345–383, 2000.

[28] J. M. Bradshaw, S. Dutfield, P. Benoit, and J. D. Woolley, "KAoS: Toward an industrial-strength open agent architecture," *Softw. agents*, vol. 13, pp. 375–418, 1997.

[29] C. I. Petrie, "Agent-based engineering, the web, and intelligence," *IEEE Expert*, vol. 11, no. 6, pp. 24–29, 1996.

[30] N. R. Jennings and M. Wooldridge, "Applications of intelligent agents," in *Agent technology*, Springer, 1998, pp. 3–28.

[31] Å. Rönnbom and L. Andersson, "Intelligent Agents-A New Technology for Future Distributed Sensor Systems?," 1999.

[32] P. Maes, "[a]Agents that Reduce Work and Information Overload, [o] Comm." ACM, 1994.

[33] H. S. Nwana and D. T. Ndumu, "A brief introduction to software agent technology," in *Agent technology*, Springer, 1998, pp. 29–47.

[34] T. Koda and P. Maes, "Agents with faces: The effect of personification," in *Proceedings 5th IEEE International Workshop on Robot and Human Communication. RO-MAN'96 TSUKUBA*, 1996, pp. 189–194.

[35] P. Maes, "Artificial life meets entertainment: lifelike autonomous agents," *Commun. ACM*, vol. 38, no. 11, pp. 108–114, 1995.

[36] C. G. Harrison, D. M. Chess, and A. Kershenbaum, *Mobile Agents: Are they a good idea?* IBM TJ Watson Research Center Yorktown Heights, New York, 1995.

[37] A. S. Rao and M. P. Georgeff, "BDI agents: From theory to practice.," in *Icmas*, 1995, vol. 95, pp. 312–319.

[38] E. Steegmans, D. Weyns, T. Holvoet, and Y. Berbers, "Designing roles for situated agents," *Agent-Oriented Softw. Eng. New York*, 2004.

[39] S. Franklin and A. Graesser, "Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents," in *International workshop on agent theories, architectures, and languages*, 1996, pp. 21–35.

[40] P. Koehn, “Intelligent Agents,” 2020.

[41] K. S. Løland, “Intelligent agents in computer games.” Institutt for datateknikk og informasjonsvitenskap, 2008.

[42] J. Rocha, I. Boavida-Portugal, and E. Gomes, “Introductory chapter: Multi-agent systems,” in *Multi-Agent Systems*, IntechOpen, 2017.

[43] T. N. T. Abd Rahim, M. S. T. Domingo, M. F. N. Batcha, and Z. Abd Aziz, “Automated Exam Question Set Generator Using Utility Based Agent and Learning Agent,” *Int. J. Mach. Learn. Comput.*, vol. 10, no. 1, 2020.

[44] S. J. Russell and P. Norvig, “Artificial Intelligence-A Modern Approach, Third Int. Edition.” Pearson Education, Upper Saddle River, NJ, USA, 2010.

[45] M. J. Wooldridge and N. R. Jennings, “Intelligent agents: Theory and practice,” *Knowl. Eng. Rev.*, vol. 10, no. 2, pp. 115–152, 1995.

[46] L. Sommer, “Exposition of a New Theory on the Measurement of Risk,” *Econometrica*, vol. 22, no. 1, pp. 23–36, 1954.

[47] F. P. Ramsey, “mTruth and Probabilitynin DH Mellor (ed) Philosophical Papers.” Cambridge University Press, Cambridge New York, 1926.

[48] C. L. Baker, R. Saxe, and J. B. Tenenbaum, “Action understanding as inverse planning,” *Cognition*, vol. 113, no. 3, pp. 329–349, 2009.

[49] R. Bradley, “Decision Theory: a formal philosophical introduction,” in *Introduction to Formal Philosophy*, Springer, 2018, pp. 611–655.

[50] M. Blokpoel, J. H. P. Kwisthout, T. P. Weide, and I. van Rooij, “How action understanding can be rational, Bayesian and tractable,” 2010.

[51] S. Costantini, “Meta-reasoning: a survey,” in *Computational Logic: Logic Programming and Beyond*, Springer, 2002, pp. 253–288.

[52] F. Callaway, S. Gul, P. M. Krueger, T. L. Griffiths, and F. Lieder, “Learning to select computations,” *arXiv Prepr. arXiv1711.06892*, 2017.

[53] S. T. Langlois *et al.*, “Metareasoning Structures, Problems, and Modes for Multiagent Systems: A Survey,” *IEEE Access*, vol. 8, pp. 183080–183089, 2020.

[54] S. Nayak *et al.*, “Experimental comparison of decentralized task allocation algorithms under imperfect communication,” *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 572–579, 2020.

[55] M. Ghallab, D. Nau, and P. Traverso, *Automated planning and acting*. Cambridge University Press, 2016.

[56] M. E. Bratman, *Planning, time, and self-governance: Essays in practical rationality*. Oxford University Press, 2018.

[57] M. P. Georgeff and A. L. Lansky, “Reactive reasoning and planning.,” in *AAAI*, 1987, vol. 87, pp. 677–682.

[58] J. Kirman, “Predicting real-time planner performance by domain characterization.” Citeseer, 1995.

[59] S. Kocherlakota and K. Kocherlakota, *Bivariate discrete distributions*. CRC Press, 2017.

[60] R. L. Morin, *Monte Carlo simulation in the radiological sciences*. CRC Press, 2019.

[61] M. Wooldridge, “Reasoning About Rational Agents, chapter 7.” The MIT Press, Cambridge, MA, 2000.

[62] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[63] B. O’Donoghue, I. Osband, R. Munos, and V. Mnih, “The uncertainty bellman equation and exploration,” in *International Conference on Machine Learning*, 2018, pp. 3836–3845.

[64] J. Achiam, E. Knight, and P. Abbeel, “Towards characterizing divergence in deep q-learning,” *arXiv Prepr. arXiv1903.08894*, 2019.

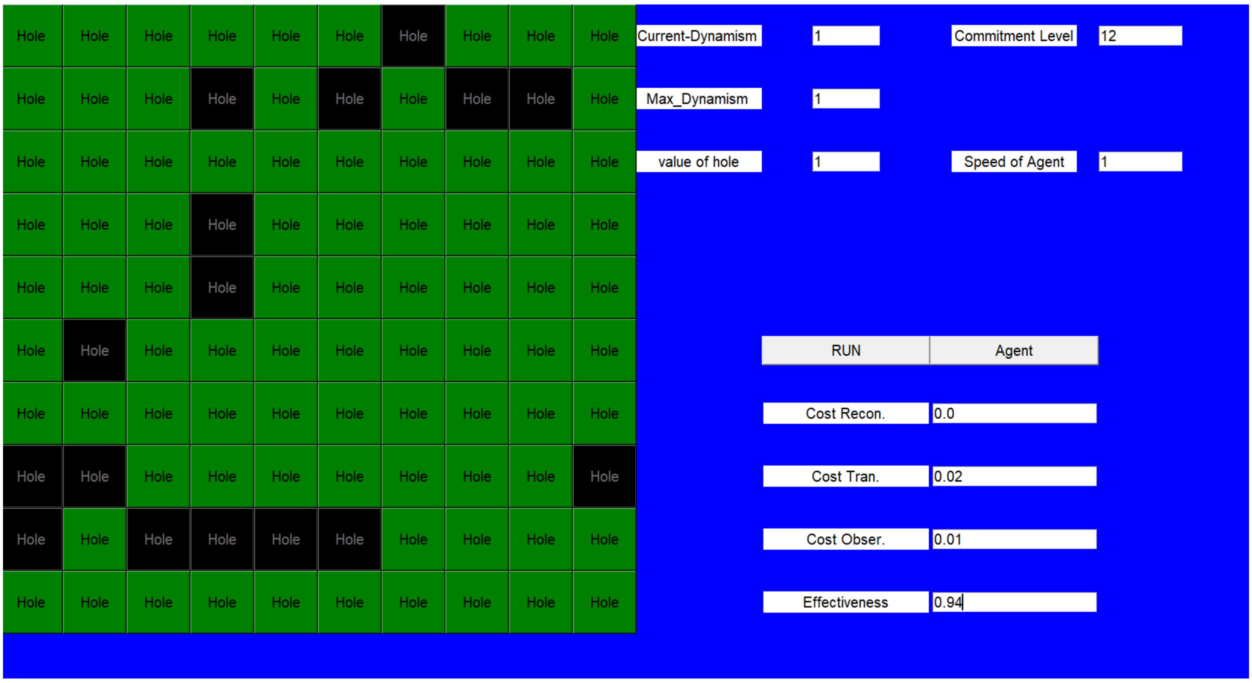
[65] M. E. Pollack, D. Joslin, A. Nunes, S. Ur, and E. Ephrati, “Experimental investigation of an agent commitment strategy,” *Pittsburgh, PA*, vol. 15260, 1994.

[66] R. S. Sutton and A. G. Barto, “Reinforcement learning: an introduction MIT Press,” *Cambridge, MA*, vol. 22447, 1998.

[67] A. Obied, “Intelligent Software Agent in E-Health System:-Review,” *J. Al-Qadisiyah Comput. Sci. Math.*, vol. 13, no. 1, p. Page-99, 2021.

Appendix

Experiments (Dynamism 1)


Current-Dynamism 1
Commitment Level 12
Max_Dynamism 1
value of hole 1
Speed of Agent 1
RUN
Agent
Cost Recon. 0.0
Cost Tran. 0.02
Cost Obser. 0.01
Effectiveness 0.94


Current-Dynamism 1
Commitment Level 10
Max_Dynamism 1
value of hole 1
Speed of Agent 1
RUN
Agent
Cost Recon. 0.0
Cost Tran. 0.06
Cost Obser. 0.01
Effectiveness 0.89


Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Current-Dynamism
1
Commitment Level
8
Max_Dynamism
1
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.005
Cost Tran.
0
Cost Obser.
0.01
Effectiveness
0.9


Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Current-Dynamism
1
Commitment Level
6
Max_Dynamism
1
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.005
Cost Tran.
0.2
Cost Obser.
0.01
Effectiveness
0.63


Current-Dynamism 1
Commitment Level 4
Max_Dynamism 1
value of hole 1
Speed of Agent 1
RUN
Agent
Cost Recon. 0.005
Cost Tran. 0.01
Cost Obser. 0.01
Effectiveness 0.93


Current-Dynamism 1
Commitment Level 2
Max_Dynamism 1
value of hole 1
Speed of Agent 1
RUN
Agent
Cost Recon. 0.005
Cost Tran. 0.06
Cost Obser. 0.01
Effectiveness 0.89

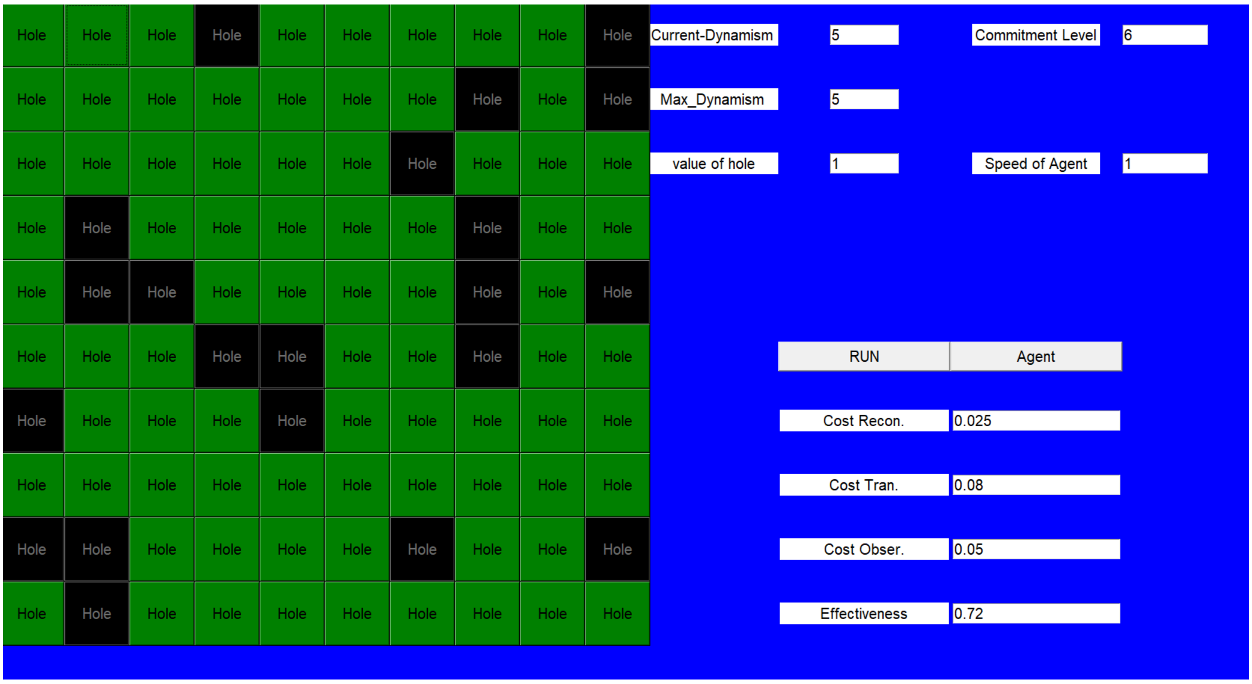
Experiments (Dynamism 5)


Hole
Current-Dynamism
5
Commitment Level
14
Max_Dynamism
5
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.02
Cost Obser.
0.05
Effectiveness
0.71


Hole
Current-Dynamism
5
Commitment Level
12
Max_Dynamism
5
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.18
Cost Obser.
0.05
Effectiveness
0.72


Current-Dynamism
5
Commitment Level
10
Max_Dynamism
5
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.17
Cost Obser.
0.05
Effectiveness
0.73


Current-Dynamism
5
Commitment Level
8
Max_Dynamism
5
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.025
Cost Tran.
0.08
Cost Obser.
0.05
Effectiveness
0.79

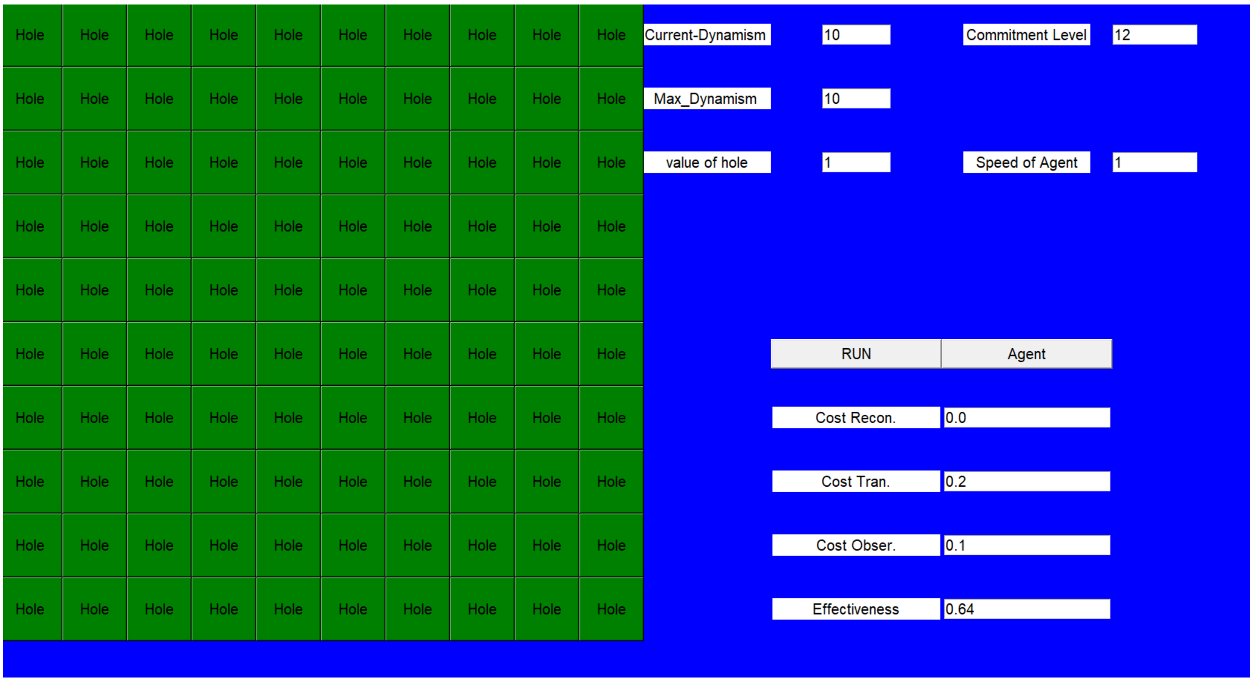

Current-Dynamism
5
Commitment Level
6
Max_Dynamism
5
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.025
Cost Tran.
0.08
Cost Obser.
0.05
Effectiveness
0.72


Current-Dynamism
5
Commitment Level
4
Max_Dynamism
5
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.22
Cost Obser.
0.05
Effectiveness
0.67


Hole
Current-Dynamism
5
Commitment Level
2
Max_Dynamism
5
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.025
Cost Tran.
0.17
Cost Obser.
0.05
Effectiveness
0.7


Hole
Current-Dynamism
5
Commitment Level
0
Max_Dynamism
5
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.04
Cost Tran.
0.33
Cost Obser.
0.05
Effectiveness
0.53

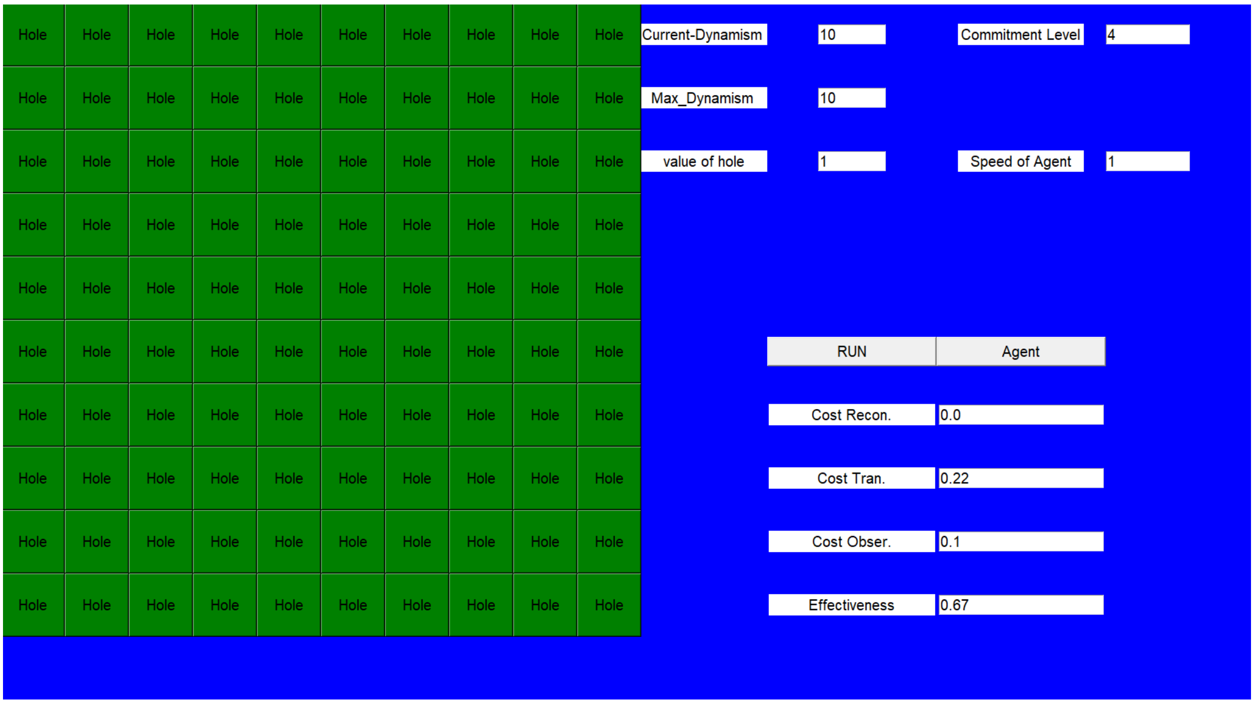
Experiments (Dynamism 10)

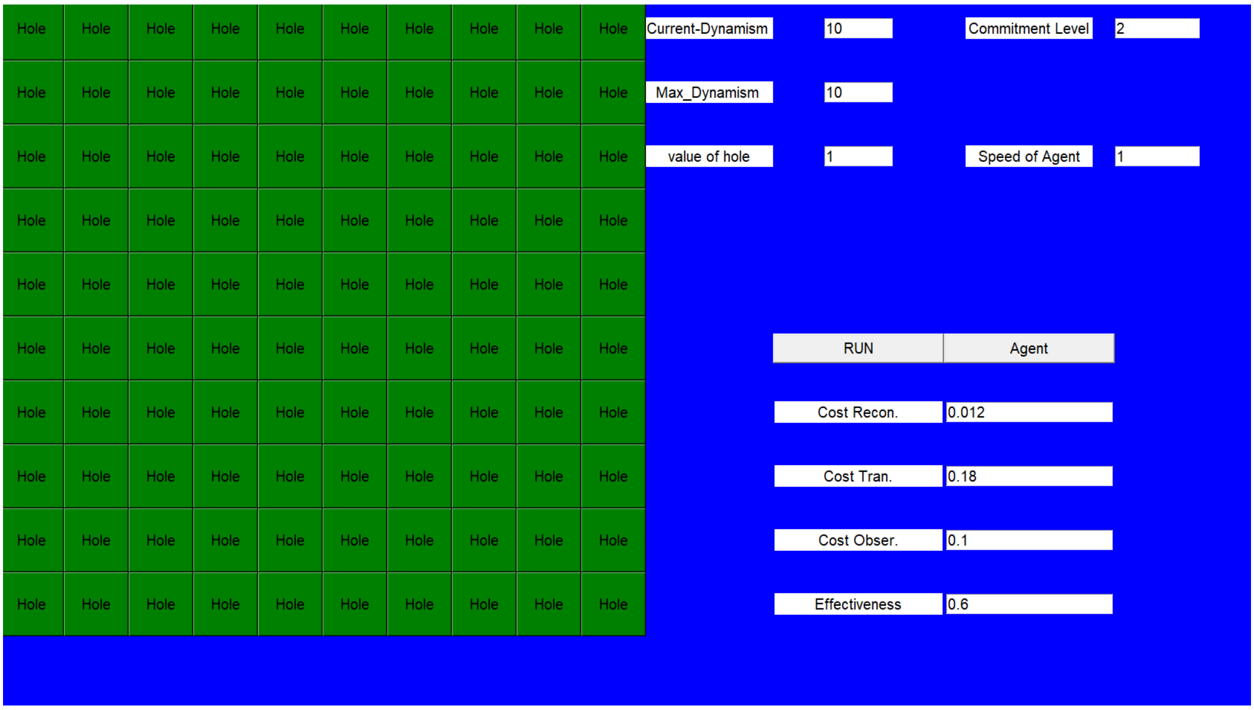

Current-Dynamism
10
Commitment Level
12
Max_Dynamism
10
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.2
Cost Obser.
0.1
Effectiveness
0.64


Current-Dynamism
10
Commitment Level
10
Max_Dynamism
10
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.21
Cost Obser.
0.1
Effectiveness
0.65

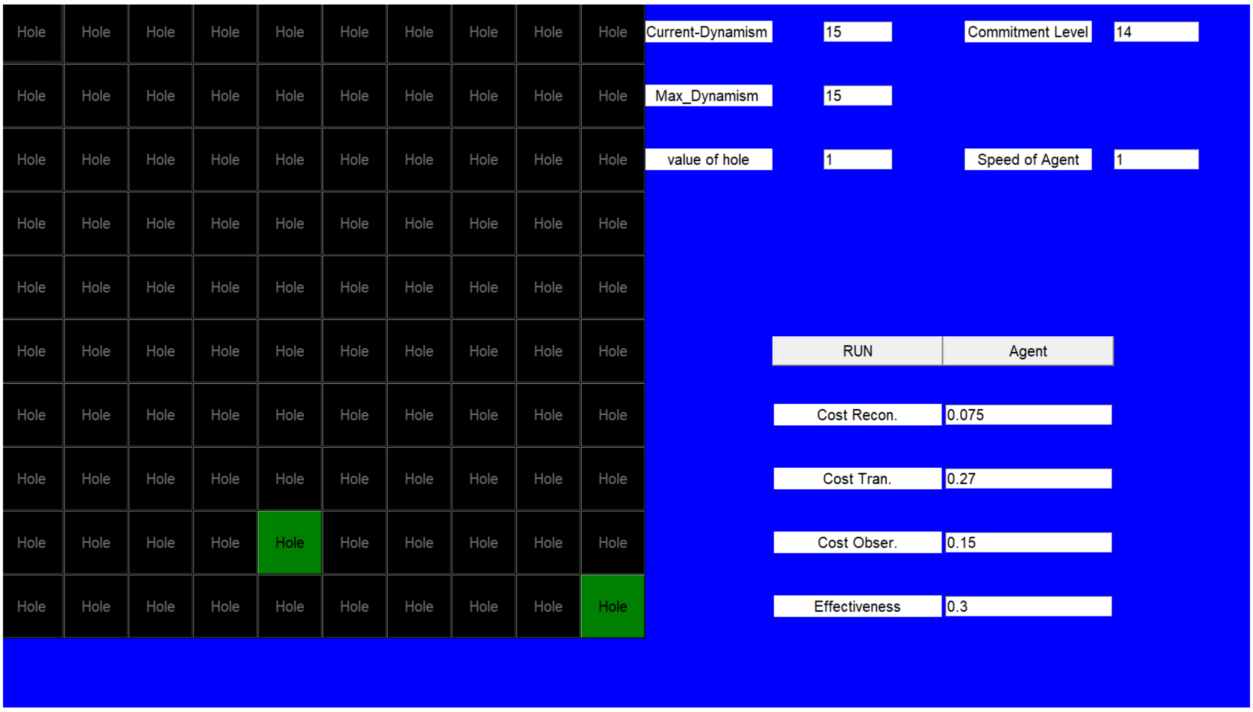

Current-Dynamism
10
Commitment Level
8
Max_Dynamism
10
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.12
Cost Obser.
0.1
Effectiveness
0.76


Current-Dynamism
10
Commitment Level
6
Max_Dynamism
10
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.13
Cost Obser.
0.1
Effectiveness
0.74


Current-Dynamism
10
Commitment Level
4
Max_Dynamism
10
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.22
Cost Obser.
0.1
Effectiveness
0.67


Current-Dynamism
10
Commitment Level
2
Max_Dynamism
10
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.012
Cost Tran.
0.18
Cost Obser.
0.1
Effectiveness
0.6

Experiments (Dynamism 15)


Hole
Current-Dynamism
15
Commitment Level
14
Max_Dynamism
15
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.075
Cost Tran.
0.27
Cost Obser.
0.15
Effectiveness
0.3


Hole
Current-Dynamism
15
Commitment Level
12
Max_Dynamism
15
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.2
Cost Obser.
0.15
Effectiveness
0.57


Current-Dynamism
15
Commitment Level
10
Max_Dynamism
15
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.25
Cost Obser.
0.15
Effectiveness
0.49


Current-Dynamism
15
Commitment Level
8
Max_Dynamism
15
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.15
Cost Obser.
0.15
Effectiveness
0.63

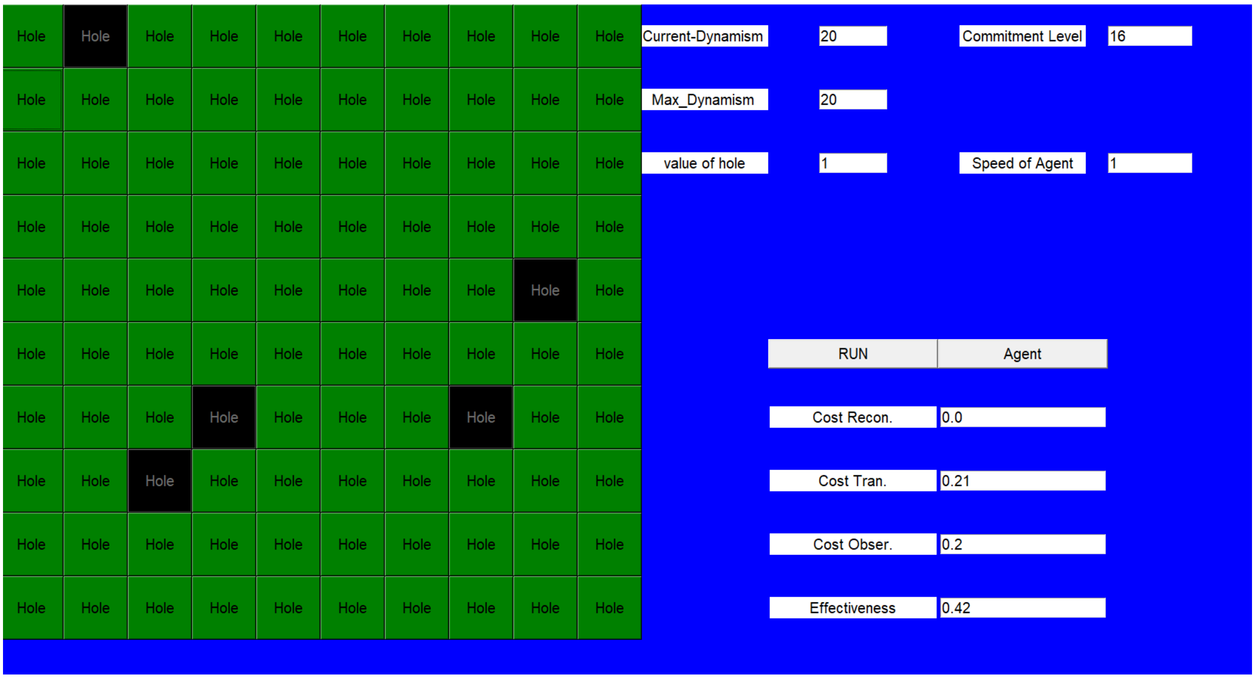

Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Current-Dynamism 15
Commitment Level 6
Max_Dynamism 15
value of hole 1
Speed of Agent 1
RUN
Agent
Cost Recon. 0.075
Cost Tran. 0.19
Cost Obser. 0.15
Effectiveness 0.46


Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Current-Dynamism 15
Commitment Level 4
Max_Dynamism 15
value of hole 1
Speed of Agent 1
RUN
Agent
Cost Recon. 0.1125
Cost Tran. 0.24
Cost Obser. 0.15
Effectiveness 0.28


Current-Dynamism
15
Commitment Level
2
Max_Dynamism
15
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.1
Cost Tran.
0.25
Cost Obser.
0.15
Effectiveness
0.43


Current-Dynamism
15
Commitment Level
0
Max_Dynamism
15
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.128
Cost Tran.
0.27
Cost Obser.
0.15
Effectiveness
0.36

Experiments (Dynamism 20)

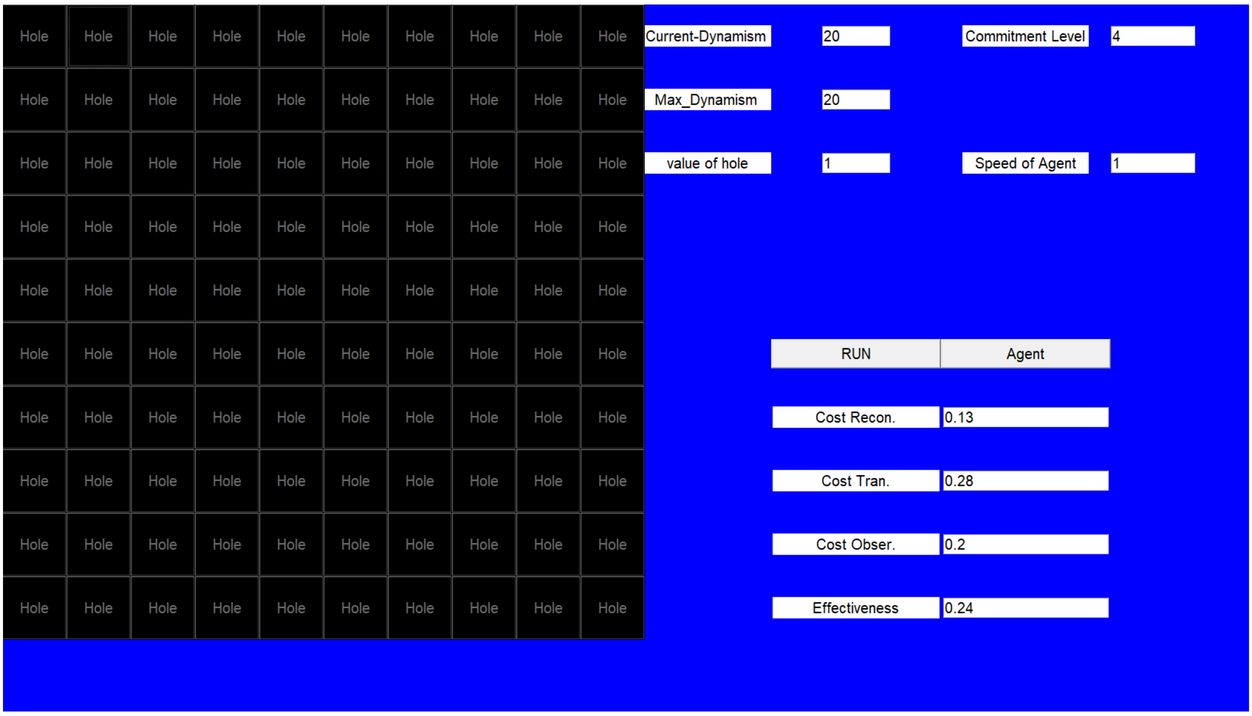


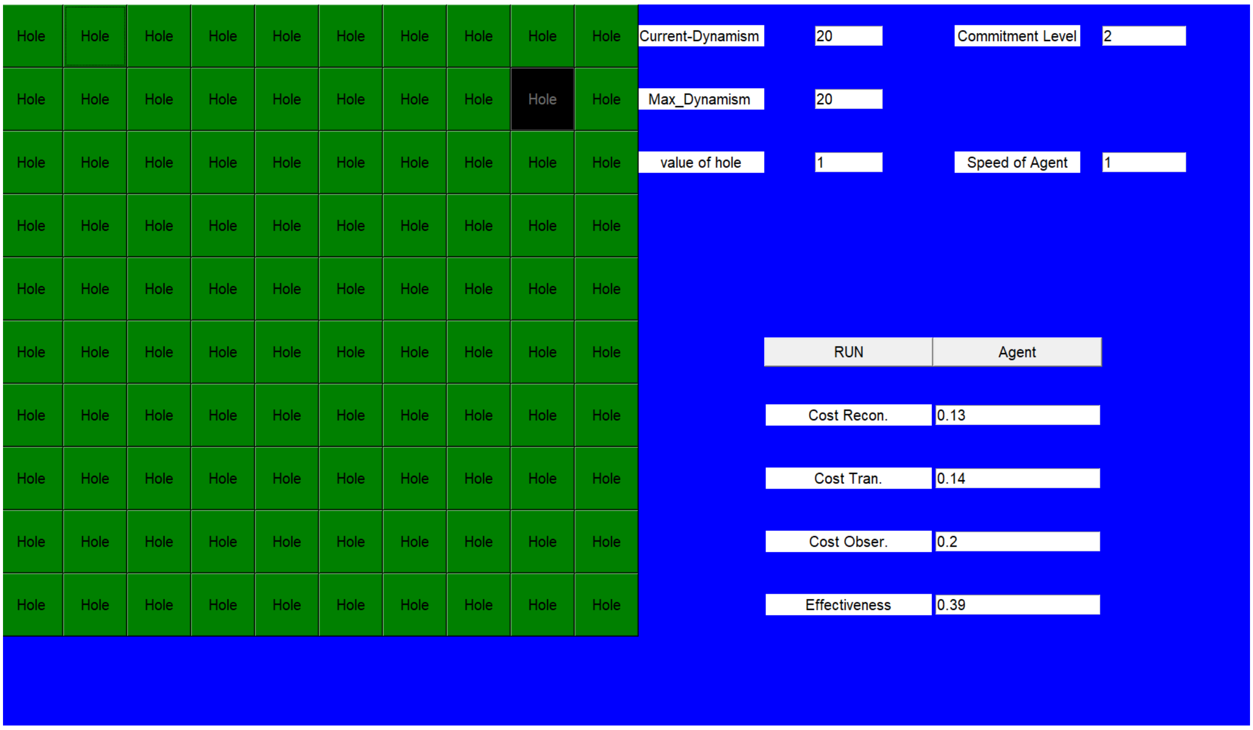

Current-Dynamism
20
Commitment Level
12
Max_Dynamism
20
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.23
Cost Obser.
0.2
Effectiveness
0.43


Current-Dynamism
20
Commitment Level
10
Max_Dynamism
20
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.27
Cost Obser.
0.2
Effectiveness
0.41

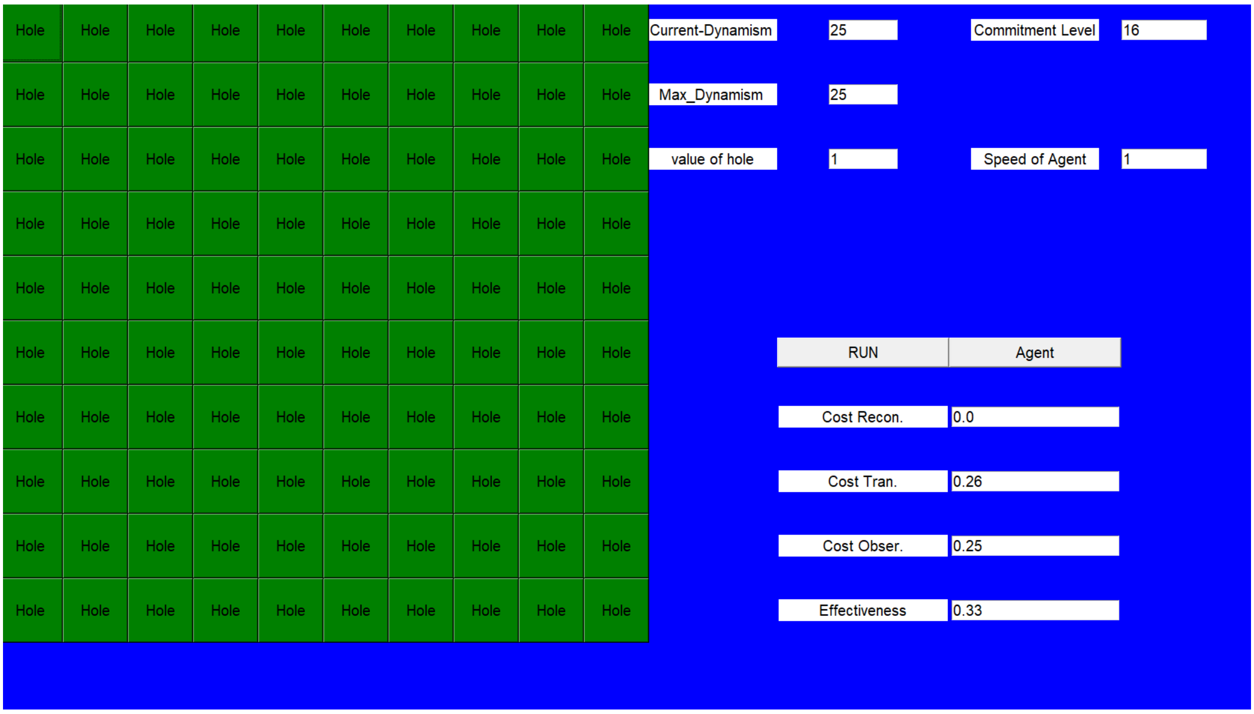

Current-Dynamism
20
Commitment Level
8
Max_Dynamism
20
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.22
Cost Obser.
0.2
Effectiveness
0.45

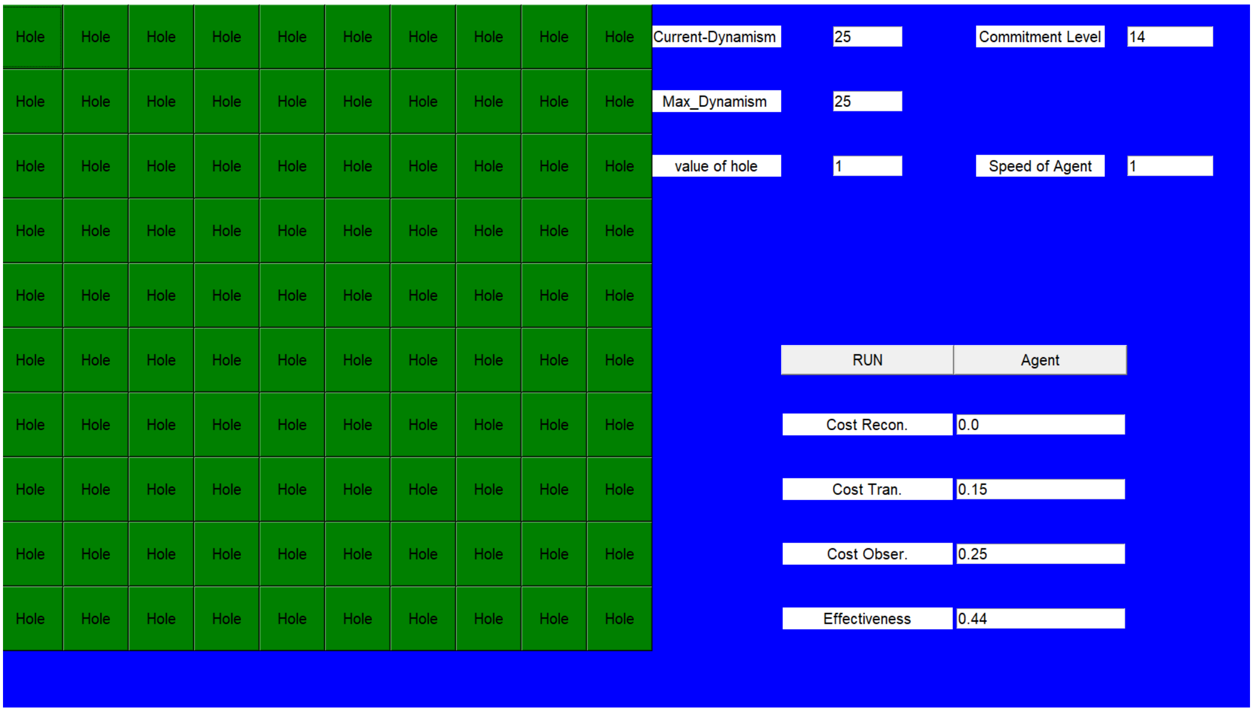

Current-Dynamism
20
Commitment Level
6
Max_Dynamism
20
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.15
Cost Tran.
0.31
Cost Obser.
0.2
Effectiveness
0.39

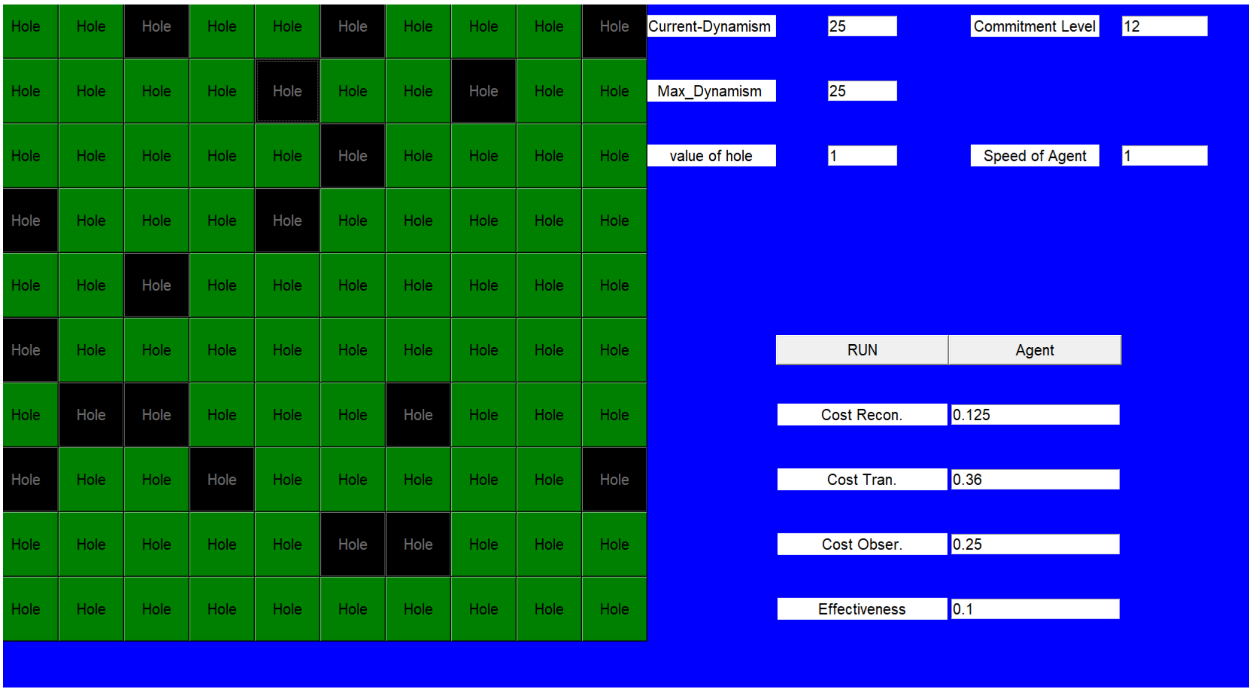

Current-Dynamism
20
Commitment Level
4
Max_Dynamism
20
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.13
Cost Tran.
0.28
Cost Obser.
0.2
Effectiveness
0.24


Current-Dynamism
20
Commitment Level
2
Max_Dynamism
20
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.13
Cost Tran.
0.14
Cost Obser.
0.2
Effectiveness
0.39

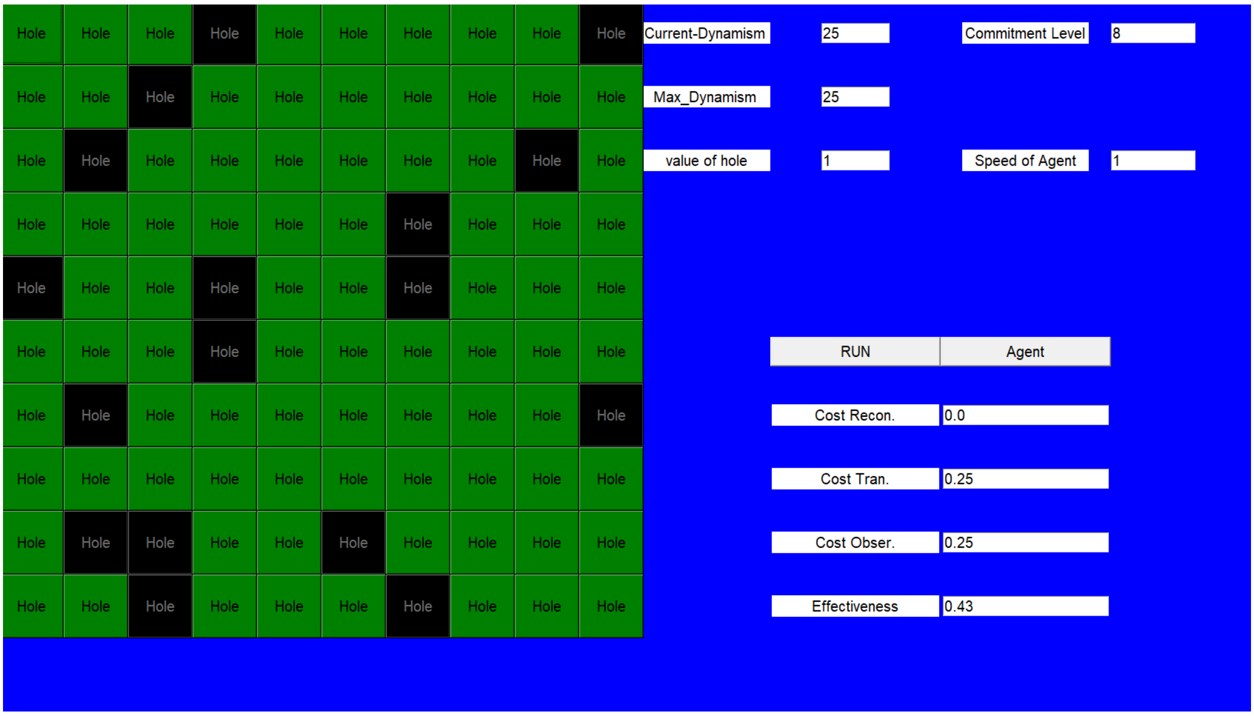
Experiments (Dynamism 25)




Current-Dynamism
25
Commitment Level
14
Max_Dynamism
25
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.15
Cost Obser.
0.25
Effectiveness
0.44


Current-Dynamism
25
Commitment Level
12
Max_Dynamism
25
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.125
Cost Tran.
0.36
Cost Obser.
0.25
Effectiveness
0.1


Current-Dynamism
25
Commitment Level
10
Max_Dynamism
25
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.36
Cost Obser.
0.25
Effectiveness
0.3


Current-Dynamism
25
Commitment Level
8
Max_Dynamism
25
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.25
Cost Obser.
0.25
Effectiveness
0.43


Hole
Current-Dynamism
25
Commitment Level
6
Max_Dynamism
25
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.125
Cost Tran.
0.08
Cost Obser.
0.25
Effectiveness
0.21


Hole
Current-Dynamism
25
Commitment Level
4
Max_Dynamism
25
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.125
Cost Tran.
0.27
Cost Obser.
0.25
Effectiveness
0.25


Current-Dynamism
25
Commitment Level
2
Max_Dynamism
25
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.187
Cost Tran.
0.14
Cost Obser.
0.25
Effectiveness
0.32


Current-Dynamism
25
Commitment Level
0
Max_Dynamism
25
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.208
Cost Tran.
0.11
Cost Obser.
0.25
Effectiveness
0.33

Experiments (Dynamism30)


Hole
Current-Dynamism
30
Commitment Level
12
Max_Dynamism
30
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.15
Cost Tran.
0.2
Cost Obser.
0.3
Effectiveness
0.2


Hole
Current-Dynamism
30
Commitment Level
10
Max_Dynamism
30
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.15
Cost Tran.
0.24
Cost Obser.
0.3
Effectiveness
0.15


Current-Dynamism
30
Commitment Level
8
Max_Dynamism
30
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.28
Cost Obser.
0.3
Effectiveness
0.34


Current-Dynamism
30
Commitment Level
6
Max_Dynamism
30
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.2
Cost Tran.
0.17
Cost Obser.
0.3
Effectiveness
0.18


Hole
Current-Dynamism
30
Commitment Level
4
Max_Dynamism
30
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.15
Cost Tran.
0.24
Cost Obser.
0.3
Effectiveness
0.17


Hole
Current-Dynamism
30
Commitment Level
2
Max_Dynamism
30
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.25
Cost Tran.
0.39
Cost Obser.
0.3
Effectiveness
0.18

Experiments (Dynamism 35)


Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Current-Dynamism 35
Commitment Level 14
Max_Dynamism 35
value of hole 1
Speed of Agent 1
RUN
Agent
Cost Recon. 0.0
Cost Tran. 0.19
Cost Obser. 0.35
Effectiveness 0.44


Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Current-Dynamism 35
Commitment Level 12
Max_Dynamism 35
value of hole 1
Speed of Agent 1
RUN
Agent
Cost Recon. 0.175
Cost Tran. 0.29
Cost Obser. 0.35
Effectiveness 0.02

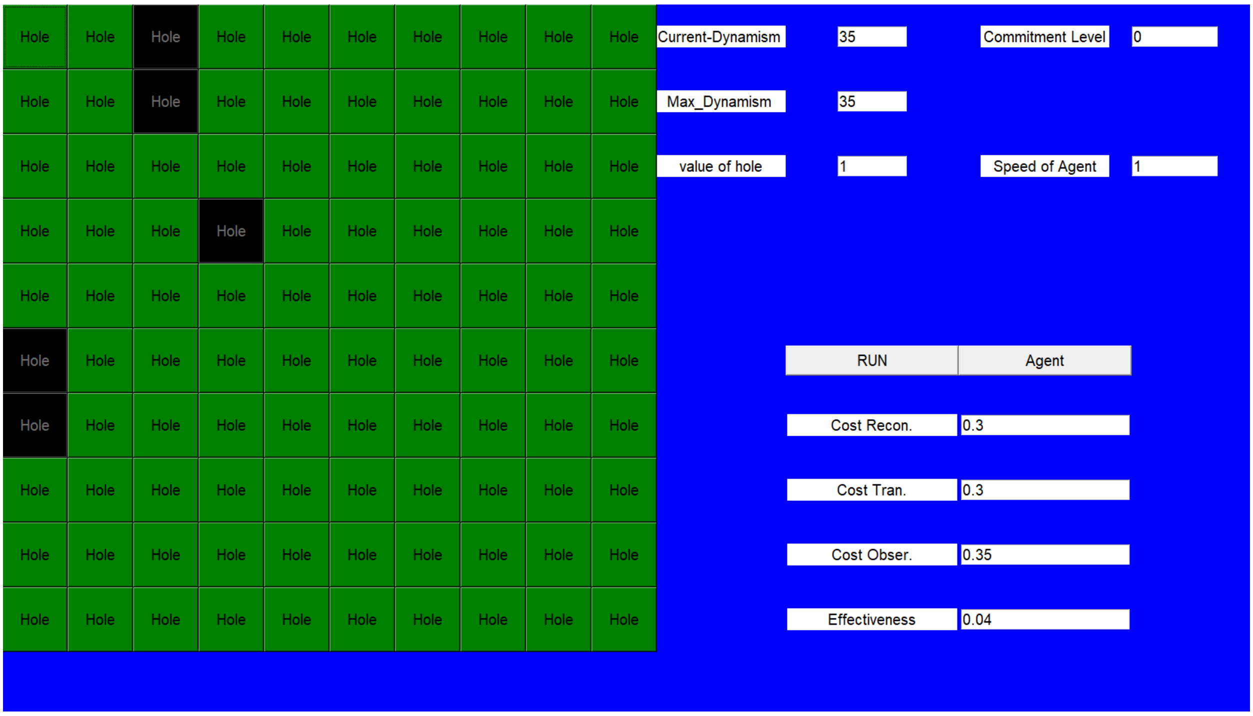

Current-Dynamism
35
Commitment Level
10
Max_Dynamism
35
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.175
Cost Tran.
0.11
Cost Obser.
0.35
Effectiveness
0.21


Current-Dynamism
35
Commitment Level
8
Max_Dynamism
35
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.29
Cost Obser.
0.35
Effectiveness
0.3


Current-Dynamism
35
Commitment Level
6
Max_Dynamism
35
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.175
Cost Tran.
0.21
Cost Obser.
0.35
Effectiveness
0.12


Current-Dynamism
35
Commitment Level
4
Max_Dynamism
35
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.23
Cost Tran.
0.23
Cost Obser.
0.35
Effectiveness
0.09


Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Current-Dynamism
35
Commitment Level
2
Max_Dynamism
35
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.28
Cost Tran.
0.28
Cost Obser.
0.35
Effectiveness
0.07


Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Hole Hole Hole Hole Hole Hole Hole Hole Hole Hole
Current-Dynamism
35
Commitment Level
0
Max_Dynamism
35
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.3
Cost Tran.
0.3
Cost Obser.
0.35
Effectiveness
0.04

Experiments (Dynamism 40)


Current-Dynamism
40
Commitment Level
12
Max_Dynamism
40
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.01
Cost Tran.
0.09
Cost Obser.
0.4
Effectiveness
0.22


Current-Dynamism
40
Commitment Level
10
Max_Dynamism
40
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.2
Cost Tran.
0.36
Cost Obser.
0.4
Effectiveness
0.14


Current-Dynamism
40
Commitment Level
8
Max_Dynamism
40
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.26
Cost Tran.
0.39
Cost Obser.
0.4
Effectiveness
0.27


Current-Dynamism
40
Commitment Level
6
Max_Dynamism
40
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.3
Cost Tran.
0.12
Cost Obser.
0.4
Effectiveness
0.02


Current-Dynamism
40
Commitment Level
4
Max_Dynamism
40
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.0
Cost Tran.
0.44
Cost Obser.
0.4
Effectiveness
0.09


Current-Dynamism
40
Commitment Level
2
Max_Dynamism
40
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.33
Cost Tran.
0.17
Cost Obser.
0.4
Effectiveness
0.04


Hole
Current-Dynamism
40
Commitment Level
0
Max_Dynamism
40
value of hole
1
Speed of Agent
1
RUN
Agent
Cost Recon.
0.32
Cost Tran.
0.29
Cost Obser.
0.4
Effectiveness
0.08

الملخص

الميزة الأكثر شهرة للآلة في عصرنا الحديث هي الاستقلالية. يتطلب هذا الاستقلال في ظروف بيئة ثابتة وديناميكية بناء نماذج، واعتماد فرضيات إحصائية، والخوض في الكثير من الاحتمالات. إن ميزة الحكم الذاتي في البيئات الثابتة هي شيء تم التغلب عليه ، لكن الكثير من الحديث عن البيئات الديناميكية ، التغير والأحداث المتعددة هو التحدي الكبير الذي يقدم العديد من الحلول.

تتضمن الدراسة المقدمة في هذه الأطروحة نظامًا للتقييم التجريبي للمنافسة على المقترحات النظرية والمعمارية ، وبصورة أدق قمنا ببناء Gridworld testbed الذي يحاكي بيئة ديناميكية حقيقية ، ويتكون النظام من ركيزة أساسية ديناميكية للبيئة و وكيل ضمني. يتم تحديد البيئة والوكيل بمستوى معين ، مما يسمح للفرد بالتحكم في خصائص كل منهما ، حتى نتمكن من التحقيق تجريبيًا في السلوكيات المختلفة لاستراتيجيات الاستدلال ضمن تفكير المستوى التلوي عن طريق تعديل المعلمات للمتغيرات البيئية وتعديل معاملات العامل المضمن. فرضيتنا هي اقتراح نظام يكتشف مدى ملاءمة استراتيجيات معينة ضمن تغييرات بيئية معينة. تم تقديم اختبار Gridworld وقد ثبت أن النظام مؤهل لتقييم معماريات الوكيل. إن اختبار Gridworld مبسط ويسهل التعامل معه ومؤهل ليكون منصة للمختبرين لأنه لا يتطلب الكثير من الجهد للتعامل معه. بالإضافة إلى ذلك ، فإن بنية Gridworld ليست مجرد بيئة محاكاة ولكنها تحتوي على عامل مضمّن أيضًا.

تم تـنـفـيذ تجارب مختلفة في ظل مستويات مختلفة من الديناميكية والالتزام وتم مقارنة نتائجنا بنتائج Tileworld تحت نفس معايير الديناميكية ومستويات الالتزام. يتفوق اختبار Girdworld على Tileworld الذي تم اختباره في معظم الأوقات.

جمهورية العراق

وزارة التعليم العالي والبحث العلمي

جامعة القادسية

كلية علوم الحاسوب وتكنولوجيا المعلومات

قسم علوم الحاسوب

تصميم و تـنفيذ منصة اختبار لمعمارية الوكيل الذكي

رسالة ماجستير

مقدمة إلى مجلس كلية علوم الحاسوب و تكنولوجيا المعلومات في جامعة القادسية كجزء من متطلبات نيل شهادة الماجستير في تخصص علوم الحاسوب

الطالب

احمد مجيد كريم البغدادي

بأشراف

أ.د علي عبيد شراد الشمري

1443 هـ

2021 م