AGI Brain: A Learning and Decision Making Framework for Artificial General Intelligence Systems based on Modern Control Theory

Mohammadreza Alidoust [1[0000-0003-3712-0535]]

m.alidoust@hotmail.com

Abstract. In this paper a unified learning and decision making framework for artificial general intelligence (AGI) based on modern control theory is presented. The framework, called AGI Brain, considers intelligence as a form of optimality and tries to duplicate intelligence using a unified strategy. AGI Brain benefits from powerful modelling capability of state-space representation, as well as ultimate learning ability of the neural networks. The model emulates three learning stages of human being for learning its surrounding world. The model was tested on three different continuous and hybrid (continuous and discrete) Action/State/Output/Reward (ASOR) space scenarios in deterministic single-agent/multi-agent worlds. Successful simulation results demonstrate the multi-purpose applicability of AGI Brain in deterministic worlds.

Keywords: Artificial General Intelligence, Modern Control Theory, Optimization, Implicit and Explicit Memory, Shared Memory, Stages of Learning, Planning, Policy, Multi-Agent, Emotions, Decision Making, Continuous and Hybrid ASOR Space

1 Introduction

In this paper, AGI Brain, a learning and decision making framework for AGI is proposed which has a unified, simple structure and tries to emulate the stages of human learning. Based on Wang's classification [1], AGI Brain looks at intelligence as a form of optimality and tries to duplicate intelligence using a unified approach by applying state-space representation (e.g. see [2]) and neural networks as its modelling technique. AGI Brain is a model based algorithm and delays its decision making stage until it built a model upon collected data from interaction with the environment. It estimates the reward value using feedforward neural networks. Like reinforcement learning (RL) [3], AGI Brain works in both continuous and discrete Action/State/Output/Reward (ASOR) spaces. But, unlike RL, AGI Brain works only in deterministic worlds with immediate rewards as yet. AGI Brain benefits from multi-agent capability. In the multi-agent case, the agents can share their experiences easily, and they can also benefit from shared memories

2 AGI Brain

2.1 The Agent and the World

Consider an intelligent agent ω living in the world γ which also contains the object ψ. The agent ω benefits from an artificial brain Γ which controls the behavior of ω for achieving its goals.
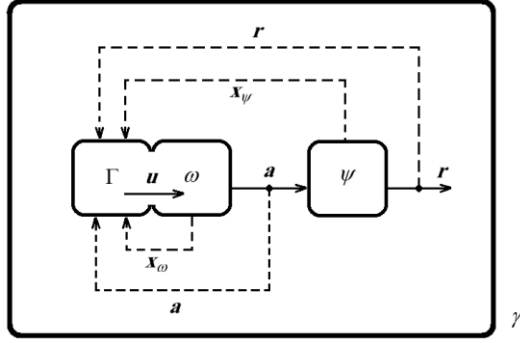
Fig. 1. The world γ consisting of the artificial agent ω and the object ψ.

At every time step n, the artificial brain Γ produces commands $\boldsymbol{u}$ (e.g. hormones or neural signals) which change the states of ω's body, i.e. $\boldsymbol{x}_{\omega}$, which then leads to performing action $\boldsymbol{a}$ on the object ψ. This action changes ψ's states $\boldsymbol{x}_{\psi}$, which consequently leads to ψ's response $\boldsymbol{r}$. Like a natural brain, the Γ can observe these values by its sensors.

We model the agent ω by its input $\boldsymbol{u}$, its states $\boldsymbol{x}_{\omega}$ and its output, i.e. action $\boldsymbol{a}$;

$$\omega: \begin{cases} \boldsymbol{x}_{\omega}(n+1) = \boldsymbol{f}_{\omega}(\boldsymbol{x}_{\omega}(n), \boldsymbol{u}(n)) \\ \boldsymbol{a}(n) = \boldsymbol{g}_{\omega}(\boldsymbol{x}_{\omega}(n), \boldsymbol{u}(n)) \end{cases} \tag{1}$$

And, the object ψ by its input $\boldsymbol{a}$, its states $\boldsymbol{x}_{\psi}$ and its output, i.e. response $\boldsymbol{r}$;

$$\psi: \begin{cases} \boldsymbol{x}_{\psi}(n+1) = \boldsymbol{f}_{\psi}(\boldsymbol{x}_{\psi}(n), \boldsymbol{a}(n)) \\ \boldsymbol{r}(n) = \boldsymbol{g}_{\psi}(\boldsymbol{x}_{\psi}(n), \boldsymbol{a}(n)) \end{cases} \tag{2}$$

The functions $\boldsymbol{f}$ and $\boldsymbol{g}$ are columns and they are, in general, complex nonlinear functions of the states and the inputs.

Please note that bold letters represent vectors or matrices. For simplicity, here we assume that both of ω and ψ change simultaneously.

From the Γ's viewpoint, the ω's body is an actuator for performing the Γ's commands, so, the world γ can be modeled by its input $\boldsymbol{u}$, its states $\boldsymbol{x} = \boldsymbol{x}_\gamma$ (vector of all states contained in the world γ, i.e. combination of $\boldsymbol{x}_\omega$ and $\boldsymbol{x}_\psi$), as well as its outputs $\boldsymbol{y} = \boldsymbol{y}_\gamma$ (vector of all outputs contained in the world γ, i.e. combination of $\boldsymbol{a}$ and $\boldsymbol{r}$).

$$\boldsymbol{x} = \boldsymbol{x}_\gamma = \begin{bmatrix} \boldsymbol{x}_\omega \\ \cdots \\ \boldsymbol{x}_\psi \end{bmatrix}, \quad \boldsymbol{y} = \boldsymbol{y}_\gamma = \begin{bmatrix} \boldsymbol{a} \\ \cdots \\ \boldsymbol{r} \end{bmatrix} \tag{3}$$

Thus, the discrete time state-space representation of the world γ will be as follows;

$$\gamma : \quad \begin{cases} \boldsymbol{x}(n+1) = \boldsymbol{f}(\boldsymbol{x}(n), \boldsymbol{u}(n)) \\ \boldsymbol{y}(n) = \boldsymbol{g}(\boldsymbol{x}(n), \boldsymbol{u}(n)) \end{cases} \tag{4}$$

And the continuous time state-space representation of the world γ will be as follows;

$$\gamma : \quad \begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \\ \boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \end{cases} \tag{5}$$

2.2 Inside the Artificial Brain

As mentioned in the previous section, in AGI Brain we look at intelligence as a form of optimality. This optimality should be observable and also measurable during the artificial life of an intelligent agent (or a swarm of them). Therefore, an optimality criterion must be defined first. We define the artificial life of an artificial intelligent agent ω living in the world γ as completing a task Q (or a series of tasks) on the object(s) ψ (or on its own body). Completion of task Q requires maximization of reward function R as the task completion criterion, which is a linear combination of objectives $\boldsymbol{J}$ and their corresponding importance coefficients $\boldsymbol{P}$ called personality, as well as an artificial brain Γ which is responsible for this optimization procedure.

For the reward function R we have;

$$R = \boldsymbol{P}^T \boldsymbol{J} \tag{6}$$

Thus, completion of task Q will be as follows;

$$Q : \quad Max \left[R = \boldsymbol{P}^T \boldsymbol{J} \right] \tag{7}$$

The aim of the artificial brain Γ is to complete Q by calculating and then producing the optimal control signal $\boldsymbol{u}^*$ such that the reward function R is maximized with respect to the state equations that govern the world γ, i.e. equation (4) (or (5)). So, in discrete time case, the brain Γ has to solve the following optimization problem at every time step n ;

$$\boldsymbol{u}^{*}(n) = \underset{\boldsymbol{u} \in \aleph}{Arg Max} \left[R = \boldsymbol{P}^{T} \boldsymbol{J} \right]$$

$$s\,t.$$

$$\begin{cases} \boldsymbol{x}(n+1) = \boldsymbol{f}(\boldsymbol{x}(n), \boldsymbol{u}(n)) \\ \boldsymbol{y}(n) = \boldsymbol{g}(\boldsymbol{x}(n), \boldsymbol{u}(n)) \end{cases}$$

(8)

Where $\aleph$ is the set of all possible alternatives. The optimal command $\boldsymbol{u}^{*}$ is the decision made by the Γ and will be transmitted to the agent's body to produce a desired action. Please note that the same problem applies to the continuous time case.

Remark 1: Objectives. The vector of objectives $\boldsymbol{J}$ is a vector that contains the objectives of the agent ω and relates the components of the world γ to the agent's goals. In general, it is a function of time, input, states and outputs as well as the desired states $\boldsymbol{x}_d$ and desired outputs $\boldsymbol{y}_d$ of the world γ, i.e. $\boldsymbol{J} = \boldsymbol{J}(n, \boldsymbol{u}, \boldsymbol{x}, \boldsymbol{x}_d, \boldsymbol{y}, \boldsymbol{y}_d)$.

Remark 2: Personality. Personality vector $\boldsymbol{P}$ is a dynamical vector which regulates the behavior of the agent and plays as the necessary condition for attaining an objective. Each element of the dynamical vector $\boldsymbol{P}$, i.e. p_k, is the coefficient of a corresponding objective j_k at the current time. In general, $\boldsymbol{P}$ is a function of the current states and outputs of the world γ, i.e. $\boldsymbol{P} = \boldsymbol{P}(\boldsymbol{x}_0, \boldsymbol{y}_0)$, and based on the current situations alters the importance of each objective to be achieved, in order to regulate the behavior of the agent ω. For instance, $\boldsymbol{P}$ activates fight-or-flight behavior by deactivating other objectives when ω senses a predator approaching. Also it leads to more diversity in multi-agent environments.

2.3 Learning and Decision Making

Equation (8) has two parts; the reward function and the state equations that govern the world γ. For solving equation (8), the artificial brain Γ has to search its available alternative set $\aleph$. For each alternative $\boldsymbol{u}^k \in \aleph$, the Γ must solve the state equations first and then evaluate the reward function. The reward function is pre-defined by the designer, but since most environments are unknown, the state equations of such environments are not available. Therefore, the agent does not know what the consequences of performing a command $\boldsymbol{u}^k \in \aleph$ are, how it changes the world and how the world may seem after performing each command.

In this major case, the agent has to build a model of the world which enables the agent to estimate the consequences of performing each alternative $\boldsymbol{u}^k \in \aleph$ on the world during the decision making stage. So, by using an estimator, the state equations of equation (8) turn into the following estimation problem:

$$\left\langle \begin{matrix} \hat{\boldsymbol{x}}(n+1) \\ \hat{\boldsymbol{y}}(n+1) \end{matrix} \right\rangle \xleftarrow{\;Estimator\;} \left\langle \begin{matrix} \boldsymbol{x}(n) \\ \boldsymbol{y}(n) \\ \boldsymbol{u}(n) \end{matrix} \right\rangle$$

(9)

Where $\hat{x}(n+1)$ and $\hat{y}(n+1)$ are the estimated states and outputs of the world γ after performing command $u(n)$ on the γ with initial conditions $x(n)$ and $y(n)$.

For building such an estimator the agent needs a set of collected data from the world, i.e. observations during its learning stages (which will be described in the following paragraph), as well as a memory to learn those collected data. In AGI Brain, due to the high power of function approximation of neural networks, the agent is equipped with two neural network memories: explicit memory (EM) and implicit memory (IM). The EM is used as the estimator model in equation (9) and the IM is used for direct production of the optimal command u^* without solving equation (8).

According to [4], there are three stages of human skill learning; 1) Cognitive: in which movements are slow, inconsistent, and inefficient, 2) Associative: in which movements are more fluid, reliable, and efficient, 3) Autonomous: in which movements are accurate, consistent, and efficient. Regarding the accuracy of the produced signals and with some connivance in the meanings, we implemented the above learning stages in our model as infancy, decision making and expert respectively. During these three learning stages, the artificial brain Γ produces three differently-produced control signals, and stores the consequences of them in its memories, EM and IM.

Infancy Stage. In the infancy stage, the agent tries to collect data from its surrounding world by interacting with it. During this stage, at every time step n, the artificial brain Γ exerts randomly-generated commands u to the world with initial conditions $x(n)$ and $y(n)$, and then observes the results $x(n+1)$ and $y(n+1)$.

Each observation vector o at time step n has the following form;

$$o^n = \left[x(n), y(n), u(n), x(n+1), y(n+1)\right]^{\mathrm{T}} \tag{10}$$

The agent stores these data in its memories EM and IM. For the EM, each observation is split into vectors $I_{EM}^n = \left[x(n), y(n), u(n)\right]^{\mathrm{T}}$ and $T_{EM}^n = \left[x(n+1), y(n+1)\right]^{\mathrm{T}}$ which are trained to the EM as its inputs and targets respectively. And, for the IM, each observation is split into vectors $I_{IM}^n = \left[x(n), y(n), x(n+1), y(n+1)\right]^{\mathrm{T}}$ and $T_{IM}^n = \left[u(n)\right]$ which are trained to the IM as its inputs and targets respectively. During this stage, these observations will not be used for producing the next commands.

Decision Making Stage. Using EM as the estimator in equation (9) and substituting equation (9) with state equations of equation (8), the decision making problem of equation (8) turns into the following equation;

$$u^*(n) = \underset{u \in \aleph}{A\,rgMax} \left[R = P^T J\right]$$

$$s\,t. \tag{11}$$

$$\left\langle \begin{matrix} \hat{x}(n+1) \\ \hat{y}(n+1) \end{matrix} \right\rangle \xleftarrow{\quad EM \quad} \left\langle \begin{matrix} x(n) \\ y(n) \\ u(n) \end{matrix} \right\rangle$$

For solving equation (11), at every time step n, the artificial brain Γ searches $\aleph$ in three stages: 1) Estimation: using its EM, the Γ estimates how the world γ would seem after executing each alternative $u^k \in \aleph$, i.e. estimating the states $\hat{x}(n+1)$ and the output $\hat{y}(n+1)$ of the world γ for each alternative $u^k \in \aleph$, given the current states $x(n)$ and outputs $y(n)$, 2) Computation: computing the influence of the estimations on reward function $R = P^T J$ with respect to $J = J(n, u, \hat{x}, x_d, \hat{y}, y_d)$ and $P = P(x_0, y_0)$, 3) Comparison: selecting the alternative which maximizes the reward function R the most as the optimal decision u^*. The learning process of the agent is also continued in this stage by learning observations with its IM and EM at certain time steps.

Planning. Planning happens when the agent cannot solve a problem at once (because of its limitations) and has to solve it by making decisions in a series of consecutive steps. During this process, the agent estimates the total value of the reward function over different possible series of consecutive inputs, i.e. $U = \{u(n_1), u(n_2), ..., u(n_f)\} \in \aleph$, that here we call them policies. Using its EM, the Γ estimates the consequences of the first member of each randomly-generated policy on current states and outputs of the γ, and then the consequence of the second input on the changed γ, and so on. By computing and then comparing the overall reward of all the policies, the Γ selects the optimal policy U^* which has the maximum reward value over the time horizon n_f. Thus, planning can be considered as an extension of decision making and we have;

$$U^* = \left\{ u(n) \,\middle|\, \underset{u \in \aleph}{ArgMax} \sum_{n=n_1}^{n_f} \left[R = P^T J \right] \right\}$$

s.t. (12)

$$\left\langle \begin{array}{c} \hat{x}(n+1) \\ \hat{y}(n+1) \end{array} \right\rangle \xleftarrow{\quad EM \quad} \left\langle \begin{array}{c} x(n) \\ y(n) \\ u(n) \end{array} \right\rangle$$

Expert Stage. In the expert stage, the IM takes over the decision making unit. Given the initial conditions x_0 and y_0 as well as the desired states x_d and outputs y_d, the IM produces the command u which transits the current states and outputs to the desired states and outputs. In this stage the commands are produced faster and they are more accurate and efficient. The learning process of the agent is also continued in this stage by learning observations with its IM and EM at certain time steps.

Role of Emotions (Stress). The role of stress as a key factor in decision making has been implemented in our model as the exploration/exploitation ratio regulator. It empowers the agent to a further exploration of the world during its second and third stage of learning. Here, we define stress as the distance between the agent's current state/output and its desired state/output, that produces stress signal s which changes the probability of selecting the optimal decision u^* (or optimal policy U^*) as follows;

$$p^{u=u^*} = \frac{1}{1+e^s}$$

(13)

Data Sharing and Shared Memories. In the multi-agent scenarios, the agents can share their experience in the form of observation vectors of equation (10). This helps the agents to benefit from the experiences of the other agents and make better decisions.

At a higher level, all of the agents can benefit from one shared explicit memory (SEM) and one shared implicit memory (SIM) which are trained with the observations of all of the agents. For training SEM and SIM, a matrix of observations O^n is formed from the observation vector o^n of each agent.
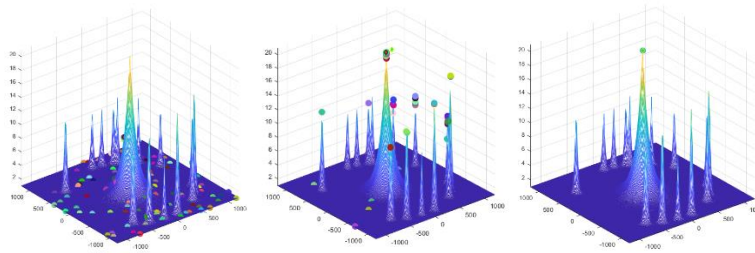
3   Simulation

3.1   Continuous ASOR Space

Function Optimization. Assume a world γ_1 which contains a function f as the object ψ_1 which is going to be optimized by a group of agents $\omega_1^1, \omega_1^2, ..., \omega_1^N$, who have these objectives: 1) j_1: finding the maxima of the function, 2) j_2: social behavior by moving close to the group, and 3) j_3: following the leader, i.e. the agent whose function value is more than the other agents. The agents have different personalities. Coefficient p_1 is positive but p_2 and p_3 are small normally distributed random numbers, so that some agents may try to get far from the group and/or the leader. Agents with negative value of p_3 get nervous when they are far from the leader and may not execute their optimal decision. They select their next moves during their three stages of learning using their SEM and SIM.

$$\gamma_1: \quad y = f(x_1, x_2) = 20\exp(-0.01\sqrt{x_1^2 + x_2^2}) + \sum_{i=1}^{20} 10\exp(-0.05\sqrt{(x_1 - x_{1i})^2 + (x_2 - x_{2i})^2})$$

(14)

Where x_i and y_i are random numbers that satisfy the equation: $x_{1i}^2 + x_{2i}^2 = 1000^2$. The simulation results are depicted in Fig. 2.
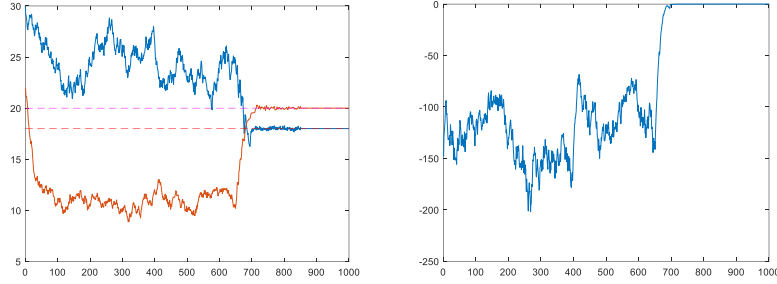

Fig. 2. Simulation Results of AGI Brain on γ_1: Small spheres represent agents' positions at: Left) the infancy stage, Center) At epoch 35 of the decision making stage some agents reached local and global maxima, Right) Expert stage: Starting from any initial position, they all reached the global maxima using SIM.

Tracking Control of a MIMO Nonlinear System. Assume a world γ_2 which contains a continuous stirred tank reactor (CSTR) as the object ψ_2, which is a multi-input multi-output (MIMO) plant described by the following state-space equations:

$$\gamma_2 : \begin{cases} \dfrac{dh(t)}{dt} = w_1(t) + w_2(t) - 0.2\sqrt{h(t)} \\ \dfrac{dC_b(t)}{dt} = (C_{b1} - C_b(t))\dfrac{w_1(t)}{h(t)} + (C_{b2} - C_b(t))\dfrac{w_2(t)}{h(t)} - \dfrac{k_1 C_b(t)}{(1 + k_2 C_b(t))^2} \end{cases} \tag{15}$$

Where $h(t)$ is the liquid level, $C_b(t)$ is concentration of the output product, $w_1(t)$ and $w_2(t)$ are input flow rates, $C_{b1} = 24.9$ and $C_{b2} = 0.1$ are input concentrations, and $k_1 = k_1 = 1$ are constants associated with the rate of consumption. The single agent ω_2 has to control the liquid level and product concentration on a predefined reference $y_d = [18, 20]^{\mathrm{T}}$ by its available actions $u(n) = [w_1(n), w_2(n)]^{\mathrm{T}}$. Fig. 3 illustrates the results of simulation.

Fig. 3. Simulation results of AGI Brain on γ_2: Left) Plant outputs during: Infancy stage $(0 \le n \le 650)$, decision making stage $(651 \le n \le 850)$, and expert stage $(851 \le n \le 1000)$. Blue line: liquid level, red line: liquid concentration, dashed red line: level reference, and dashed magenta line: concentration reference, Right) Total reward value.

3.2 Hybrid ASOR Space

Animats. The term animats coined by S.W. Wilson refers to artificial animals which interact with real or artificial ecosystems and have the following key properties: 1) autonomy, 2) generality, and 3) adequacy. They exist in a sea of sensory signals, capable of actions, act both externally and internally, and their certain signal or absence of them have special status [5].

Assume a world γ_3 which is an artificial ecosystem that contains a cattle of grazing animats $\omega_3^1, \omega_3^2, ..., \omega_3^N$, pastures $\psi_3^{ps1}, \psi_3^{ps2}, ..., \psi_3^{psM}$, and a predator ψ_3^{Pr}. The animats have the following objectives: 1) Staying alive by a) j_1: grazing for increasing their energy, b) not getting hunted by the predator by j_2: moving as far as possible from it, and j_3: by moving close to the cattle, 2) j_4: Reproduction by mating, and 3) j_5: Searching for new pastures ψ_3^{psl}. The animats have the following discrete actions set: $\aleph = \{\text{up}, \text{down}, \text{right}, \text{left}, \text{eat}, \text{mate}\}$.

The rules of the world γ_3 are as follows: The animats can move by performing u_1=up, u_2=down , u_3=right and u_4=left . They can only increase their energy by performing action u_5 = eat when they are near a pasturage, otherwise they lose one unit of energy. They can only reproduce by performing action u_6 = mate when they are mature and they have enough energy. The animats get hunted and die by the predator ψ_3^2 when they are very close to it. Also, they die whether their energy level is equal to zero, or they get old. Corresponding elements of the dynamic personality vector $\boldsymbol{P}(\boldsymbol{x}_0,\boldsymbol{y}_0)$ will change when different situations occur, e.g. when the animat ω_3^k observes the predator near to it, $p_2(\boldsymbol{x}_0,\boldsymbol{y}_0)$ and $p_3(\boldsymbol{x}_0,\boldsymbol{y}_0)$ are increased while other elements are decreased to zero. They select their next actions during their three levels of learning using their SEM and SIM. Fig. 4 illustrates snap shots of simulation.
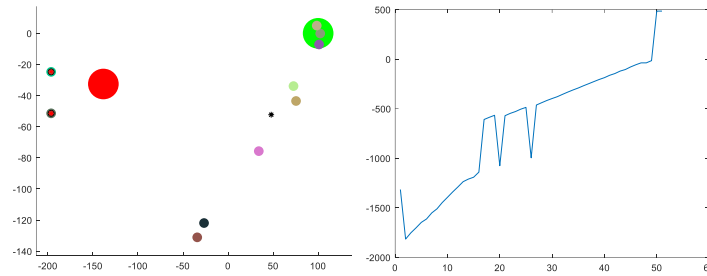
Fig. 4. Simulation snapshots of AGI Brain on γ_3: Left) Snapshot of the agents' positions during decision making stage. The big green circle represent a pasturage, small circles represent the agents moving towards the pasturage, big red circle represent the predator, and small circles with red star represent hunted agents, Right) Reward value of agent ω_3^k during decision making stage.

4　Discussion and Future works

In this paper, the AGI Brain as a learning and decision making framework was introduced. The model utilizes optimization as its theory and state-space representation as its technique. It implements the three levels of human learning using its IM and EM. Also it incorporates planning as well as some mental factors of the human intelligence like emotions and different personalities. AGI Brain was tested on two category of problems with continuous and hybrid ASOR deterministic spaces.

As can be seen in the simulation section, the model returned successful results in the above-mentioned scenarios. Utilizing different personalities helped in different behavior as well as exploration/exploitation diversity. For instance, in the first scenario, agents with different personalities found other local maxima of the function. Also, implementation of the role of stress resulted in more exploration, e.g. in the second scenario. Additionally, the model benefits from ultimate learning power of neural networks. The learning problem of neural networks when dealing with discrete (categorical) ASORs was solved by incorporating pattern recognition neural networks, and high precision learning achieved, e.g. in the third scenario. In multi-agent scenarios, the agents utilize SEM and SIM where they can share their experiences with other agents.

Despite the successful results of AGI Brain on current scenarios, still more developments, implementations and tests on other scenarios must be performed, in order to guarantee the multi-purpose applicability of AGI Brain. Currently, the model is parameter-dependent, e.g. the time horizon of planning. In order to determining the *sufficient* amount of a parameter for successfully accomplishing a policy on a desired task, the model should also learn these parameters during its learning stages, preferably during its infancy stage. The model works only in deterministic environments as yet. For stochastic environments, the IM and EM must be empowered with memories which are able to correctly estimate, for example, the results of an action in a stochastic world. For large alternatives sets $\aleph$, the performance speed of the model can be improved by searching over smaller random subsets of $\aleph$ (whose elements are randomly selected from $\aleph$), instead of searching over $\aleph$. The size of those subsets of $\aleph$ with respect to the size of $\aleph$ determines a trade-off between speed and accuracy.

References

[1] P. Wang, "Artificial General Intelligence, A gentle introduction," [Online]. Available: https://sites.google.com/site/narswang/home/agi-introduction. [Accessed January 2019].

[2] P. Paraskevopoulos, Modern Control Engineering, Taylor & Francis Group, 2002.

[3] R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction, 2nd ed., MIT Press, 2018.

[4] G. Wulf, Attention and Motor Skills Learning, Human Kinetics, 2007.

[5] C. Strannegard, N. Svangard, D. Lindstrom, J. Bach and B. Steunebrick, "Learning and Decision Making in Artificial Animals," *Journal of Artificial General Intelligence,* vol. 9, no. 1, pp. 55-82, 2018.