A LITERATURE REVIEW ON NEURO-COGNITIVE LEARNING AND CONTROL

By

PATANJALIKUMAR SHASHANKKUMAR JOSHI

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

Of the Requirements

For the Degree of

MASTER OF SCIENCE IN ELCTRICAL ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2014

Acknowledgements

I am very grateful to my supervisor, Dr. Frank Lewis. He has provided excellent guidance for my research work. His vast knowledge and experience in the area of control systems has helped me understand various systems in a better way and has inspired to dive deeper research and some courses like Optimal Control, Distributed Control, Intelligent Control and Nonlinear Control. I am very happy to get an opportunity to work with him.

Moreover, I would like to thank Dr. Daniel S. Levine and his PhD and Masters students for facilitating me to understand neuroscience and psychology which were required for my research work.

Also, I am thankful to PhD students Hamidreza Modares, Bahare Khomartash, Bakur AlQaudi, Raghavendra Sriram, and Masters student Rubayiat Tousif for providing materials and guidance whenever asked for. I am also grateful to the support provided by the ONR Grant # N00014-13-1-0562. Lastly, I thank my family and friends for their encouragement and support.

November 24, 2014

Abstract

A LITERATURE REVIEW ON NEURO-COGNITIVE LEARNING AND CONTROL

Patanjalikumar Shashankkumar Joshi, M.S.

The University of Texas at Arlington, 2014

Supervising Professor: Frank L. Lewis

This thesis is an effort to provide a foundation work linking neuroscience, psychology and control theory as a part of research going on developing fast satisficing autonomous systems at the University of Texas at Arlington Research Institute (UTARI). This literature review, the compilation is aimed to facilitate information and references needed for neurocognition and control.

There is so much research going on to understand the neural mechanisms of a mammal brain, especially human brain. Although it is not fully understood, there are proposed and proven theories that address intelligence, various learning and decision making processes performed by various parts of the brain. Cerebellum is hypothesized to be responsible for supervised learning, cerebral cortex for unsupervised learning and basal ganglia for reinforcement learning with help of dopamine. Probability, the representation of data and emotions do affect the decision process. A shunting inhibitory neural network which include amygdala, orbitofrontal cortex, ventral striatum, thalamus and anterior cingulated cortex, is involved when the decision process is affected by probability and emotions. That is related with gist and verbatim, too. Cognitive abilities also make difference in decisions. It is proposed that there are multiple learning and control loops in the brain.

With aim of replicating brain-like intelligence, multiple actor-critics solve Bellman equation using approximate dynamic programming for optimal control. Multiple model based architectures for learning and control have been proposed which also find optimal control for systems. These

architectures include artificial neural network which utilize shunting inhibition and multiple model reinforcement learning.

Still, to achieve brain-like intelligence, optimality is not necessary. Satisficing decision has to only meet only some minimum acceptance; it does not have to be optimal, so it can be faster. Inclusion of satisficing in multi-player games is beneficial. Also, it is not always possible to make optimal choices due to various limitations, so with a bounded rationality, choices have to be made. Finally, the goal is to develop a framework that can learn and control various systems, fast and efficiently with limited resources and rapidly changing environment.

Table of Contents

List of Illustrations

Chapter 1

Introduction

The term "intelligent control" has been used in a variety of ways. To us, "intelligent control" should involve both intelligence and control theory. It should be based on a serious attempt to understand and replicate the phenomena that we have always called "intelligence"-i.e., the generalized, flexible, and adaptive kind of capability that we see in the human brain. There are five chapters which provide information related to the above mentioned areas. Each chapter is further divided into sections which group similar aspects.

Chapter 2 discusses some of the earliest work done to understand and model learning and decision process done in the brain. It has three sections. The first section describes the work done by Kenji Doya, W. Schultz and others that focus on process in basal ganglia and cerebral cortex. The second section talks about work done by Paul Werbos who suggested various ADP models of the brain. The third section discusses about various parts of the brain involved in the decision process.

Chapter 3 also talks about cognitive development from psychological perspective. The first section puts together the effort done by Daniel Levine to model the decision process in the brain which mainly involves orbitofrontal cortex, amygdala and relates emotion, risk and probability to decision. The second section talks about Paul Werbos newer work in ADP. The third section illustrates cognitive abilities and decision making from psychological studies which involves Piaget's theory of cognitive development.

Chapter 4 takes a look at the new, neuro-cognitively developed learning and control mechanisms. The first section discusses learning structures which include reinforcement learning, fuzzy logic and shunting inhibitory artificial neural networks. Inspired and understood through all the neuro-physiological studies, multiple actor-critic architecture is used for model prediction and control. This involves multiple model based reinforcement learning, parallel neural networks, multiple model based adaptive control and the eMOSAIC model.

Chapter 5 is about satisficing which is different from optimality. With the time and resource constraints, optimality is not always needed. Also this can result in faster decisions. The first section discusses about satisficing control theory. The second section is application of satisficing in game theory which results in satisficing games. Satisficing is like having something good or better which is not the best; it sounds more like just getting satisfied.

Chapter 6 talks about bounded rationality. The first section explains the concept of bounded rationality. It is related to psychology, economics and management. Its impact also been studied in peer-to-peer networks. The second section illustrates metacognition which is a state of 'knowing of knowing'. It is related with bounded rationality and satisficing.

This work is aimed at providing links between neuroscience, psychology and control systems. Detailed study of mechanism of computations and decisions in human brain has been presented. It is further strengthened with findings from a psychological perspective. Concepts of satisficing and bounded rationality are included. Architectures for learning and control which are inspired through, and use all these findings are presented so that an integrated compilation has been prepared on basis of which faster, more efficient decisions and control structures can be designed for various autonomous systems.

Chapter 2

Work in the 1990's on Neuro-cognition

2.1 Learning and Control in Basal Ganglia and Cerebral Cortex

This section discusses roles played by the cerebral cortex and basal ganglia in various learning processes and control. It involves work done by Kenji Doya [1, 2, and 3] and Wolfram Schultz [4-8]. Both, neurophysiology and mathematics of the brain processes are illustrated. To further explain the reward prediction during reinforcement learning, a study and its findings have been also included.

Equation Chapter 2 Section 1

2.1.1 Introduction

It was originally believed that the cerebellum and the basal ganglia were dedicated to only motor control. But more and more evidence is being found suggesting their involvement in non-motor tasks [1, 2]. By studying anatomical features of their structures Doya suggests that the cerebellum, the basal ganglia and the cerebral cortex are each specialized for a particular kind of computation. They are reciprocally connected with each other (Fig. 1) and simultaneously active [2]. A theory is proposed that the cerebellum implements 'supervised learning', the basal ganglia 'reinforcement learning' and the cerebral cortex implements 'unsupervised learning' [1].

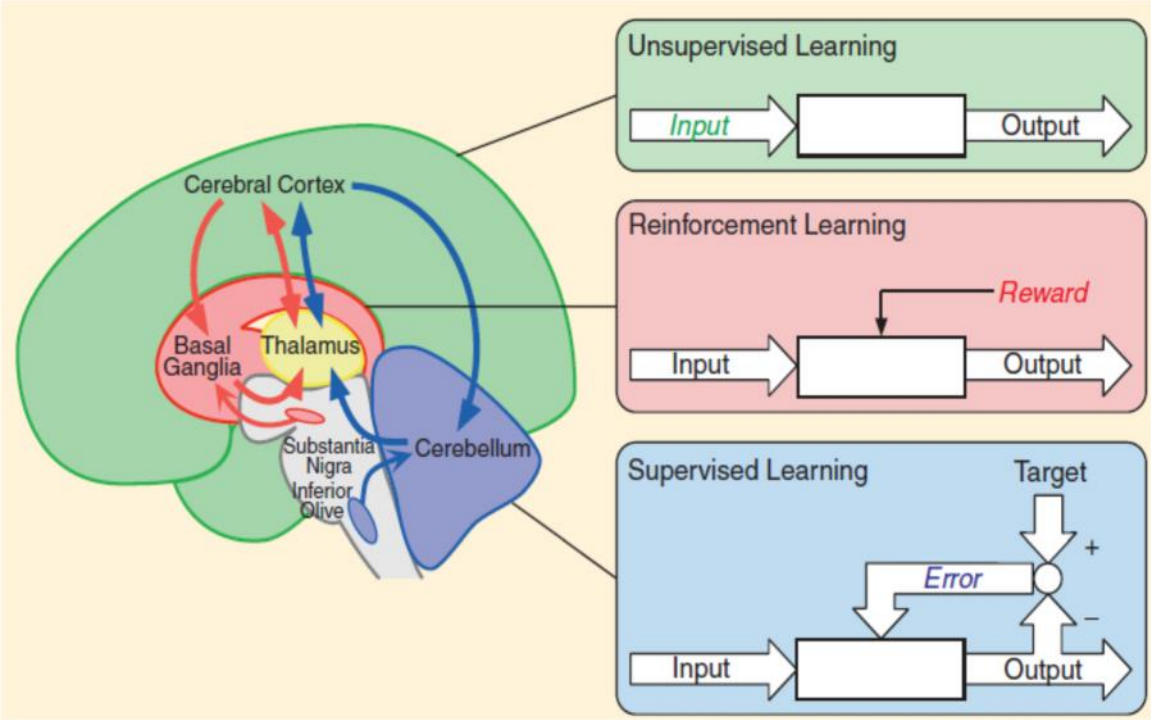

Figure 1 Interconnection of the cerebellum, the basal ganglia and the cerebral cortex [2]

2.1.2 Supervised Learning in the Cerebellum

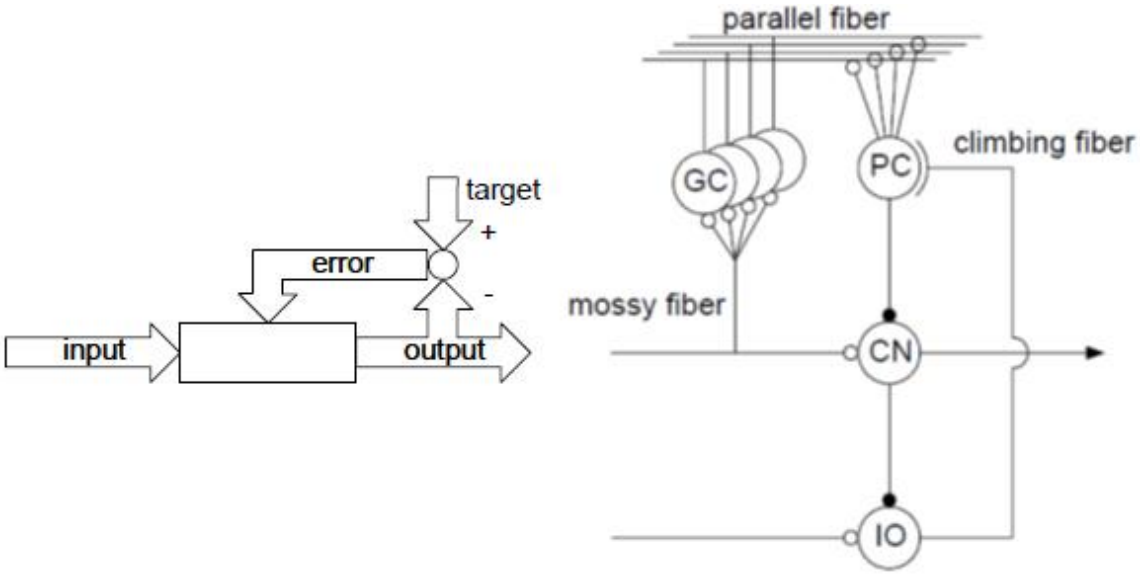

Figure 2 Cerebellar circuit for supervised learning. •-inhibitory connection, CN-Deep cerebellar nuclei, IO-Inferior olive, Empty circle-excitatory connection [1]

The cerebellum circuit for the supervised learning is shown in the figure 2. It has a nearly feed-forward structure with massive synaptic convergence of granule cell (GC) axons (parallel fibers) onto purkinje cells (PC). The purkinje cells receive inputs from both parallel fibers and the climbing fiber. The output is provided by the neurons located in the deep cerebellar nuclei. An input-output mapping is computed through the supervised learning [1].

$$\mathbf{y} = F(\mathbf{x}) \tag{2.1.1}$$

where the output $\mathbf{y} = (y_1, ..., y_m)$' and the input $\mathbf{x} = (x_1, ..., x_n)$'.

The movement related signals are encoded by the purkinje cells' simple spike responses to parallel fiber input and the errors in movement are encoded by the climbing fiber. The mapping is found from the desired output $(\hat{\mathbf{y}}(1), \hat{\mathbf{y}}(2), ...)$ in order to minimize the expected output error such as [1],

$$E_{\mathbf{x}}\left[\left\|\hat{\mathbf{y}} - \mathbf{y}\right\|^2\right] \tag{2.1.2}$$

In case of unknown distribution of the input, it can be approximated by minimizing the sum of squared errors at sample data points

$$E = \sum_t \left\|\hat{\mathbf{y}}(t) - \mathbf{y}(t)\right\|^2 = \sum_t \left\|\hat{\mathbf{y}}(t) - F(\mathbf{x}(t); \mathbf{w})\right\|^2 \tag{2.1.3}$$

under a certain constraint on the mapping F. The outputs of the granule cells are linearly combined by a purkinje cells as [1]

$$y_i(t) = \sum_{j=1}^{n} w_{ij} x_j(t), \tag{2.1.4}$$

where w_{ij} is a synaptic connection weight and it can be updated/learned by the gradient descent of the sample error [1]

$$\triangle w_{ij} \propto -\frac{\partial E}{\partial w_{ij}} = \left(\hat{y}_i(t) - y_i(t)\right) x_j(t), \tag{2.1.5}$$

That is, the parameter updates based on the correlation between the output error and the presynaptic input [1].

2.1.3 Reinforcement Learning in the Basal Ganglia

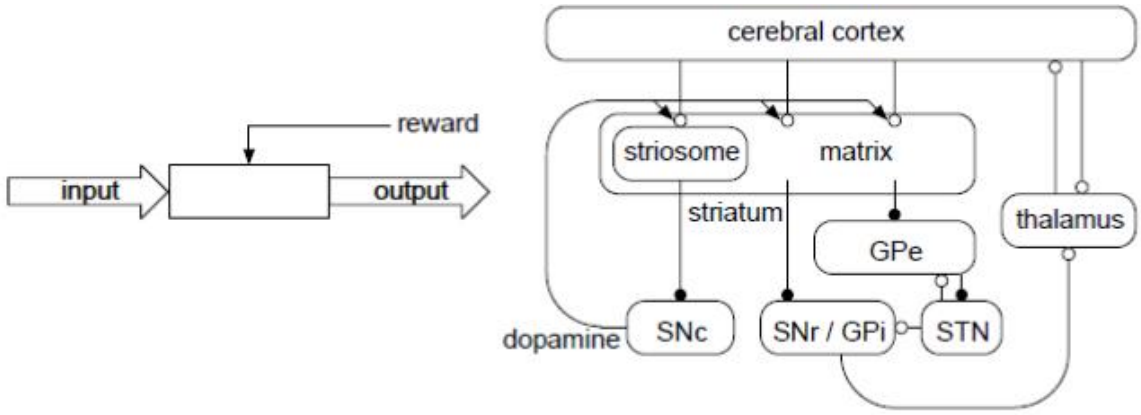

Figure 3 Neural circuit of the basal ganglia. SNc-Substantia nigra pars compacta, SNr-Substantia nigra pars reticula, GPi-Internal segment of globus pallidus, GPe-External segment of globus pallidus, STN-subthalamic nucleus, O-Excitatory connection, •-Inhibitory

The circuit of the basal ganglia implementing reinforcement learning is shown figure 3. The main input from the cerebral cortex goes to the striatum which consist of a part called striosome and a part called matrix. A learning agent takes an action u(t) ∈ R^m in response to the state x(t) ∈ R^n of the environment [1].

$$\mathbf{x}(t+1) = F(\mathbf{x}(t), \mathbf{u}(t)) \tag{2.1.6}$$

When an unexpected reward or a sensory cue signaling the delivery of a reward in near future is related with phasic increase in firing of dopamine neurons in SNc. The reward [1]

$$r(t+1) = R(\mathbf{x}(t), \mathbf{u}(t)) \tag{2.1.7}$$

The matrix compartment selects the action which maximizes the cumulative sum of rewards [1]

$$\mathbf{u}(t) = G(\mathbf{x}(t)) \tag{2.1.8}$$

The striosome plays the role of value prediction mechanism for the maximization mentioned above [1]

$$V(\mathbf{x}) = E\left[r(t+1) + ¥ r(t+2) + ª^2 r(t+3) + \ldots \right] \tag{2.1.9}$$

where a discount factor $0 \le ¥ \le 1$.

The value function can be learned by minimizing the 'temporal difference' (TD) error of the reward prediction which is encoded by the dopamine neuron activity [1]

$$¢(t) = r(t) + ¥V(\mathbf{x}(t)) - V(\mathbf{x}(t-1)) \tag{2.1.10}$$

which signals the inconsistency of the current estimate of the value function. For a value function [1],

$$V(t) = \sum_{j=1}^{n} v_j x_j(t) \tag{2.1.11}$$

the learning algorithm for the weight v_j is [1]

$$\triangle v_j \propto ¢(t) x_j(t-1) \tag{2.1.12}$$

The policy can be improved by simply taking a stochastic action [1]

$$u_i(t) = g\left(\sum_{j=1}^{n} w_{ij} x_j(t) + «_i(t) \right) \tag{2.1.13}$$

where g() is a gain function and $«_i(t)$ is a noise term. The TD error δ(t) as defined in (2.1.10) then signals the unexpected delivery of the reward r(t) or the increase in the state value V(x(t)) above expectation. The learning algorithm for the action weight is given by [1]

$$\triangle w_{ij} \propto ¢(t)\left(u_i(t-1) - \bar{u}_i\right) x_j(t-1) \tag{2.1.14}$$

where $\bar{u}_i$ is the average level of the action input. Thus, the TD error $¢(t)$ works as the main teaching signal in both learning of the value and the selection of actions [1].

2.1.3.1 Reward Prediction

Let us have more insight into how the basal ganglia predicts reward as discussed by Schultz, Tremblay and Hollerman in [7]. Rewards serve three basic objectives: (1) They serve as goals of behavior, (2) They increase the frequency and intensity of behavior to achieve goals, and (3) They prompt subjective feelings of pleasure and positive emotional states [7].

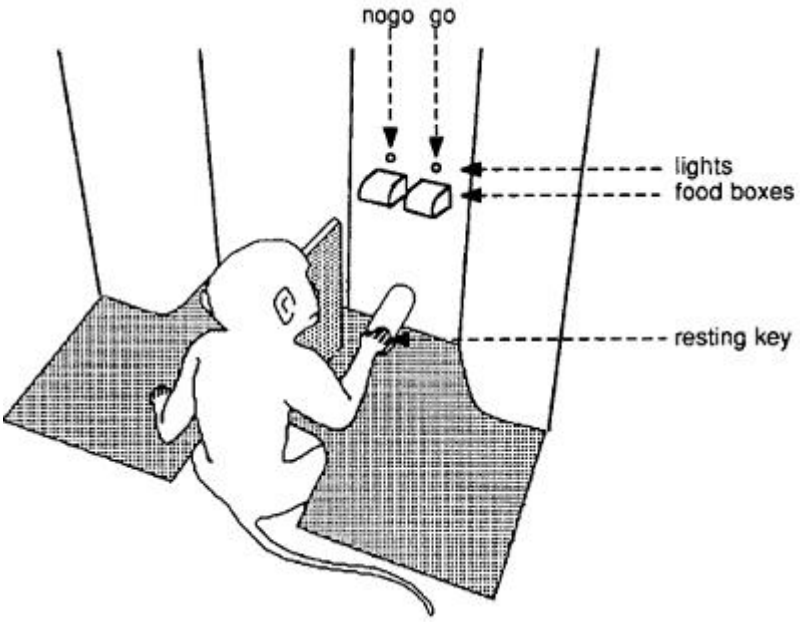

Figure 4 The monkey-reward experiment [4]

Figure 4 shows the experiment conducted by Schultz and others to determine how neurons in the basal ganglia and frontal cortex process different aspects of reward information.

2.1.3.2 Error of Prediction Reward Coded by Dopamine Neurons

If reward-predicting stimuli is absent, the dopamine neurons respond to primary food and fluid rewards. Schultz, Tremblay and Hollerman in [7] observed first response to primary reward in an experiment when a monkey touched a morsel of food which was behind a cover during self-initiated movements in the absence of phasic, reward-predicting stimuli (Fig. 4). When the monkey touched inedible objects, no response was observed (figure 5).

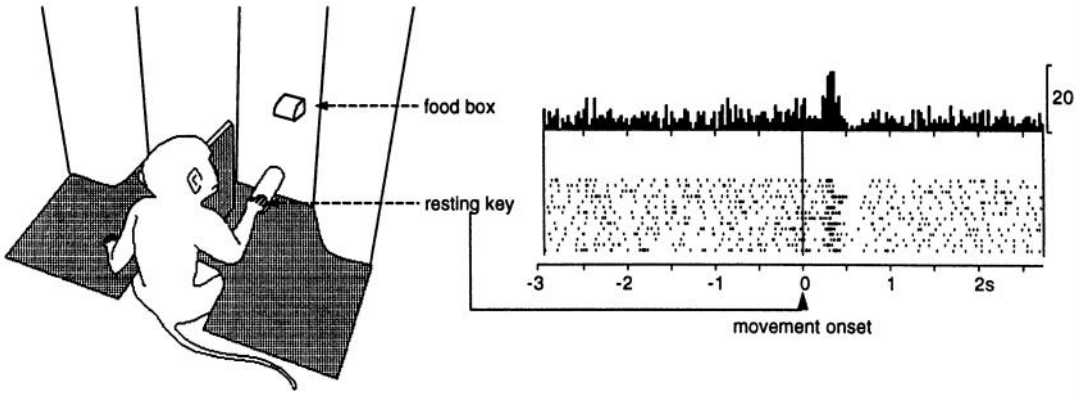

Figure 5 Response of dopamine neuron to touch of food reward [7]

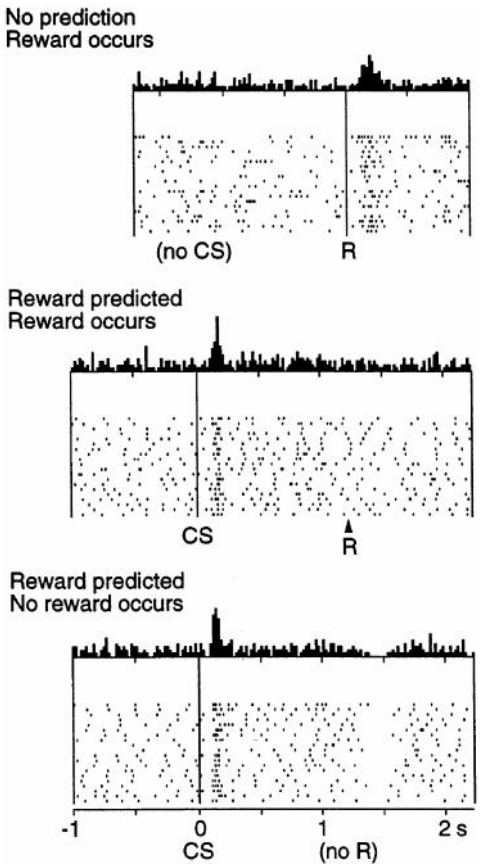

Figure 6 Error of reward prediction detected by the dopamine neurons [7]

It can be seen that the dopamine neurons are also activated when reward is presented without any stimulus during learning (Fig. 6, top). After learning the task, the dopamine response occurs after the reward-predicting conditioned stimulus and depletes after the reward (figure 6,

middle). If a fully predicted reward does not occur when it should have occurred, then at that time the activity of the dopamine neurons is depressed (figure 6, bottom). This suggests that dopamine neurons encode the error in prediction of reward. Learning slows down as the prediction error decreases and the outcome is predicted more accurately. It is suggested that the dopamine response is a scalar reinforcement signal provided simultaneously to all neurons in the striatum, although dopamine neurons cannot discriminate between different rewards [7].

2.1.3.3 Learning Changes in Reward Expectation by Striatal Neurons

Schultz and the others in [7] found that the neurons in the striatum have access to central stored representations of experiences of previous individual task events which also includes the rewards. As show in figure 7, the reward expectation-related activations did not occur with unrewarded movements; but occurred only during external reinforcements.

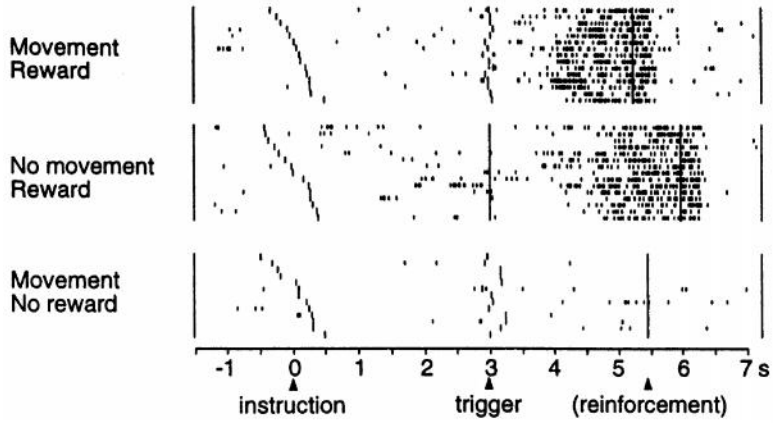

Figure 7 Reward expectation-related activity in primate putamen neuron [7]

Some of these activations were able to distinguish between different types of reward. It indicates that the striatal neurons can access previous and current expectation-related activity information and update/adapt them according to novel situation. Thus, they can validate and provide accurate information about rewards in advance. This is very much different from the activity of dopamine neurons which encode the temporal difference error between prediction and actual occurrence of the reward [7].

2.1.4 Unsupervised Learning in the Cerebral Cortex

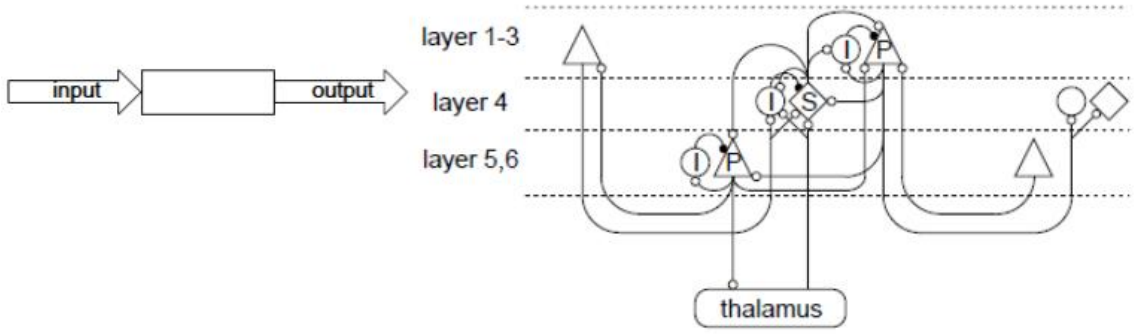

Figure 8 Unsupervised learning neural circuit of cerebral cortex. P-Pyramidal neurons, S-Spiny stellate neurons, I-Inhibitory interneurons, o-Excitatory connection, •-Inhibitory connection [1]

It can be seen in figure 11 that the cerebral cortex has a layered organization and massive recurrent connections. The cerebral cortex has different functional areas representing sensory, motor or contextual information in different modalities and frames of reference. The statistical properties of the inputs are characterized by a mapping constructed from a set of input data (x(1), x(2), …,) ∈ R^n to the output (y(1), y(2), …,) ∈ R^m. One way to maximize the mutual information between the input and the output as defined in [1],

$$H(\mathbf{x};\mathbf{y}) = H(\mathbf{x}) - H(\mathbf{x}\,|\,\mathbf{y}) \tag{2.1.15}$$

where H denotes the entropy $H(\mathbf{x}) = E\left[-\log p(\mathbf{x})\right]$. It enumerates the decrease in uncertainty about input x by knowing output y. An objective function can be used to derive an unsupervised algorithm [1]

$$E = \left\|\mathbf{x}(t) - W'\mathbf{y}(t)\right\|^2 + \sum_{i=1}^{m}\left|y_i(t)\right| \tag{2.1.16}$$

where W is the input-output weight matrix, the first term represents the input reconstruction error and the second term embodies a sparseness constraint. This yields maximization of the mutual information and reassurance of the majority of outputs to be close to zero [1].

The information coding of the neurons in the cerebral cortex can be determined by the relaxation dynamics [1]

$$\dot{\mathbf{y}} \propto -\frac{\partial E}{\partial \mathbf{y}} = W\mathbf{x} - WW'\mathbf{y} - \mathrm{sign}(\mathbf{y}) \tag{2.1.17}$$

The synapses' weights are updated by a Hebbian rule given by the gradient descent [1]

$$\Delta W \propto -\frac{\partial E}{\partial W} = \mathbf{y}(\mathbf{x} - W'\mathbf{y})' = \mathbf{y}\mathbf{x}' - \mathbf{y}\mathbf{y}'W \tag{2.1.18}$$

References

1. Kenji Doya, “What are the computations of the cerebellum, the basal ganglia and the cerebral cortex?” Neural Networks 12, 1999, pp. 961-974.

2. Kenji Doya, Hidenori Kimura, Mitsuo Kawato, “Neural Mechanisms of Learning and Control,” IEEE Control System Magazine, Vol. 21, 2001, pp. 42-54.

3. Kenji Doya, Hidenori Kimura, Aiko Miyamura, “Motor control: neural models and system theory”, International Journal of Applied Mathematics and Computer Science, Vol. 11, 2001, pp. 101-128.

4. Wolfram Schultz, Ranuifo Romo, “Role of primate basal ganglia and frontal cortex in the internal generation of movements-I. Preparatory activity in the anterior striatum”, Experimental Brain Research 91, 1992, pp. 363-384.

5. Ranuifo Romo, Eugenio Scarnati, Wolfram Schultz, “Role of primate basal ganglia and frontal cortex in the internal generation of movements-II. Movement related activity in the anterior striatum”, Experimental Brain Research 91, 1992, pp. 385-395.

6. Ranuifo Romo, Wolfram Schultz, “Role of primate basal ganglia and frontal cortex in the internal generation of movements-III. Neuronal activity in the anterior striatum”, Experimental Brain Research 91, 1992, pp. 396-407.

7. Wolfram Schultz, Leon Tremblay, Jeffrey R. Hollerman, “Reward prediction in primate basal ganglia and frontal cortex”, Neuropharmacology 37, 1998, pp. 421-429.

8. Wolfram Schultz, Leon Tremblay, Jeffrey R. Hollerman, “Reward processing in primate orbitofrontal cortex and basal ganglia”, Cerebral Cortex, 2007, pp. 272-283.

2.2 Brain like Intelligence and Approximate Dynamic Programming

In this section, I have discussed some of the initial efforts done by Paul Werbos [1-4] to design and develop brain-like intelligence. He tried to combine neuro-physiology and control system theory. In the process, he developed a concept of Approximate Dynamic Programming (ADP). Here, the first and second generation models of ADP are discussed. The third generation ADP model is discussed in section 3.2.

2.2.1. Introduction

Werbos' work explains the basic mathematical principles and their relation to the most important features of a mammal brain–how intelligence works so that a control system can be designed that can learn to perform the complex range of tasks. As even a mouse has a structure like six-layer neocortex and it shows general purpose learning abilities, understanding the mouse brain is an important step toward understanding the human mind [2]. From this, approximate dynamic programming emerges which combines control theory and neural networks.

2.2.2 from Optimality to ADP

People have tried to understand the human brain using the idea of optimization. Animal behavior is finally about choices as shown in the figure 9.

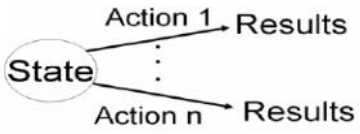

Figure 9 Animal Behavior Choice [2]

The rules of the action selection can be fixed for a simple animal; while they can be selected based on the computed outcomes for the taken actions for an advance animal. Werbos [2] defines functionality as an ability of the brain about making choices which yield better results;

while Intelligence as an ability of the brain about learning how to make better choices. But all this has to be put into mathematics to have a way to find the better result [2].

Now one may wonder that if brains are so optimal, then also humans do so many stupid things! This can be explained by Von Neumann's Cardinal Utility function which is the foundation of decision theory and dynamic programming among others. The brains are designed to learn approximate optimal policy with bounded computational resources. They never learn to play a perfect game of chess! Over the course of time, various ADP models of brain intelligence have been developed.

2.2.3 First Generation ADP Model

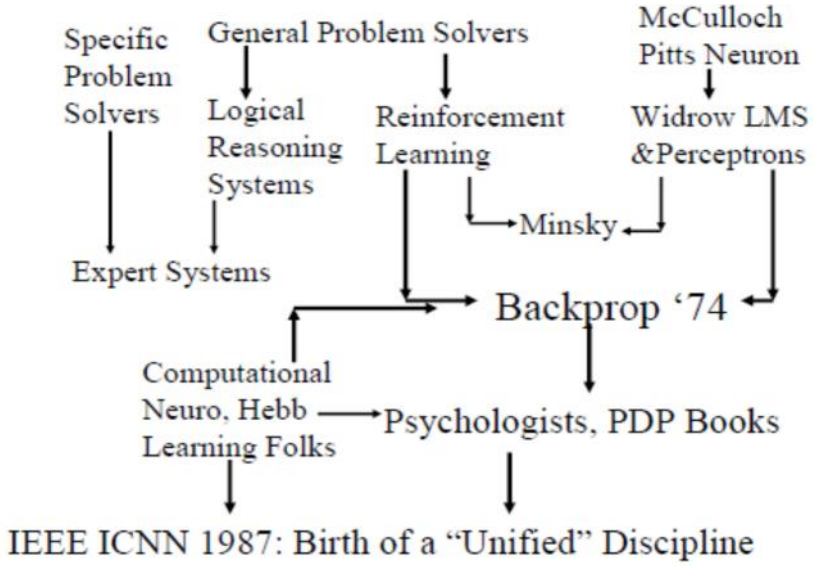

Figure 10 Origins of ANN [2]

As shown in figure 10, both back-propagation and the first ADP design originated in Werbos' work. It was proposed after many tried to develop brain like intelligence that the decision system which can learn to approximate the Bellman equation could be built (1971-72). With noise, an optimal strategy or policy of action for a general nonlinear decision problem can be computed efficiently [2].

$$J(\underline{\mathbf{x}}(t)) = \max_{\underline{\mathbf{u}}(t)} \left\langle U(\underline{\mathbf{x}}(t), \underline{\mathbf{u}}(t)) + J(\underline{\mathbf{x}}(t+1)) \right\rangle / (1+r) \qquad (2.2.1)$$

where $\underline{\mathbf{x}}(t)$ is the state of the environment at time t, $\underline{\mathbf{u}}(t)$ is the choice of actions, U is the cardinal utility function, r is the interest or discount rate, the angle brackets denote expectation value and J is the function that must be solved in order to derive an optimal strategy of action.

A system can learn to approximate this policy by using a neural network to approximate the Bellman equation as shown in figure 11.

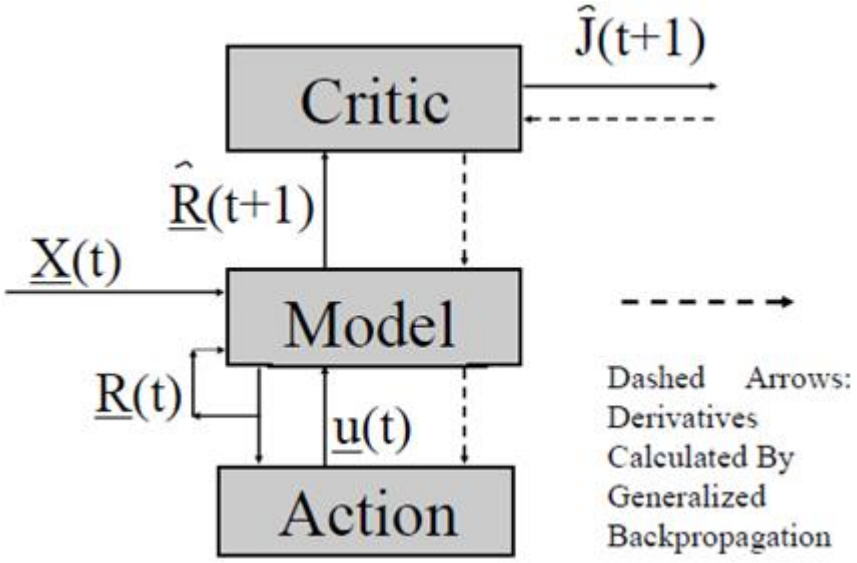

Figure 11 First possible emergent intelligence [2]

The “Action” network computes or decides the actions to be taken by the organism. The “Model” network learns both, a way to predict changes in the environment and a method to estimate the objective state of reality (R) which is a different input than the current sensory input $(\underline{\mathbf{X}})$. The “Critic” network estimates the J function, a kind of learned value function. Werbos proposed a method to adapt the Critic network called Heuristic Dynamic Programming (HDP) which was later called the Temporal Difference (TD) method. But this method learns too slowly, so he developed the core idea of Dual Heuristic Programming (DHP) [4]. He assumed a discrete time clock in the first generation model of the brain as the cerebral cortex is “modulated” (clocked) by regular timing signals. These timing signals come from bursts of the output of many types of neurons at regular time intervals with continuously varying intensity [2].

2.2.4 Second Generation ADP Model

Werbos proposed a second generation ADP model in 1987 as shown in figure 16.

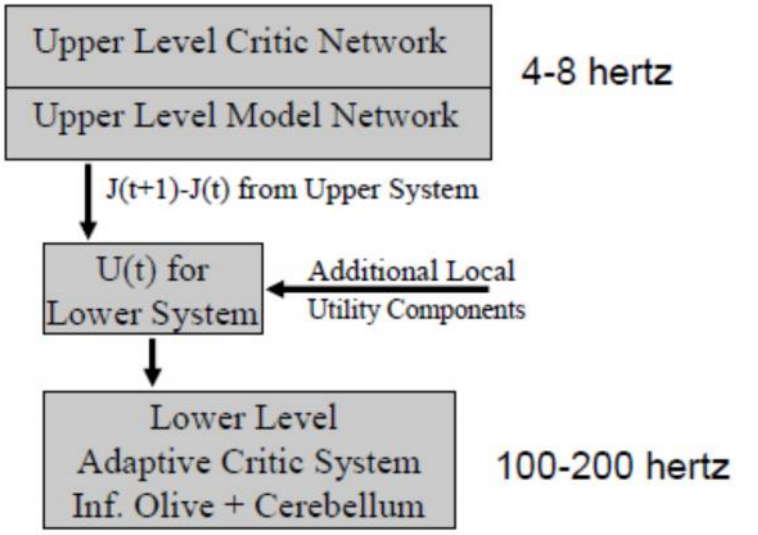

Figure 12 Second generation brain model [2]

It was motivated both by trying to understand the brain and by an engineering dilemma. The critic and actor networks required more powerful networks than feed-forward neural networks. It requires use of recurrent networks which give out the result after many cycles of inner loop computations. This low sampling rate is observed for the cerebral cortex responding to inputs from the thalamus, while muscle control is done at a much higher rate. This is similar to a master-slave system. Werbos suggested an error critic along with a fast model-free slave neural network model. The purkinje cells of the cerebellum are modeled as memory neurons in the action network which estimated the vector R. The training is done by a distributed DHP-like critic system. A strong, stable continuous time model free ADP design like the maser-slave arrangement has been formulated [2].

2.2.5 Approximate Dynamic Programming

This section will discuss about two ADP methods taken from [3], HDP and DHP to adapt critic networks. It is assumed that at each time t: (1) An estimate of the state vector $\mathbf{R}(t)$ is available, (2) the action network A computes the actions $\mathbf{y}(t) = \mathbf{A}(\mathbf{R}(t))$ and total utility $U(\mathbf{R}(\mathrm{t}), \mathbf{u}(\mathrm{t}))$ and, (3) the action u(t) is transmitted to the environment.

Also, pre-availability of a model network is assumed [3]:

$$\hat{\mathbf{R}}(t+1) = f(\mathbf{R}(t), \mathbf{u}(t)) \tag{2.2.2}$$

Only procedures to set up the inputs and the targets of a network are described here. One has freedom to choose a type of the critic network.

HDP is based on an attempt to approximate Howard's form of the Bellman equation; while DHP is based on differentiating the Bellman equation. $\mathbf{u}(t)$ is defined as a function of R which maximizes the right-hand side of the Bellman equation. With r=0, the Bellman equation becomes [3]:

$$J(\mathbf{R}(t)) = U(\mathbf{R}(t), \mathbf{u}(\mathbf{R}(t)) + \left\langle J(\mathbf{R}(t+1)) \right\rangle - U_0 \tag{2.2.3}$$

Differentiating and applying chain rule [3]:

$$\begin{aligned} {}^{\mathrm{a}}{}_i(\mathbf{R}(t)) &\triangleq \frac{\partial J(\mathbf{R}(t))}{\partial R_i(t)} = \frac{\partial U(\mathbf{R}(t), \mathbf{u}(t))}{\partial R_i(t)} \\ &+ \sum_j \frac{\partial U(\mathbf{R}, \mathbf{u})}{\partial u_j} \cdot \frac{\partial u_j(\mathbf{R}(t))}{\partial R_i(t)} + \sum_j \left\langle \frac{\partial J(\mathbf{R}(t+1))}{\partial R_j(t+1)} \cdot \frac{\partial R_j(t+1))}{\partial R_i(t)} \right\rangle \\ &+ \sum_{j,k} \left\langle \frac{\partial J(\mathbf{R}(t+1))}{\partial R_j(t+1)} \cdot \frac{\partial r_j(t+1)}{\partial u_k(t)} \cdot \frac{\partial u_k(t)}{\partial R_j(t)} \right\rangle \end{aligned} \tag{2.2.4}$$

2.2.5.1 Implementation of HDP

HDP is a procedure for adapting a network or the function $\hat{J}(\mathbf{R}(t), W)$, which attempts to approximate the function $J(\mathbf{R}(\mathrm{t}))$. The HDP can be implemented as follows [3]:

1. Obtain and store R(t) (actual or simulated) and compute $\mathbf{u}(t) = \mathbf{A}(\mathbf{R}(t))$.

2. Obtain R(t+1) by waiting until t+1 or by predicting $\mathbf{R}(t+1) = f(\mathbf{R}(t), \mathbf{u}(t))$.

3. Calculate:

$$J^*(t) = U(\mathbf{R}(t), \mathbf{u}(t)) + \hat{J}(\mathbf{R}(t+1), W) / (1+r) \tag{2.2.5}$$

5. Update W in $\hat{J}(\mathbf{R}(t), W)$ based on inputs R(t) and target $J^*(t)$ by using any real time supervised learning method.

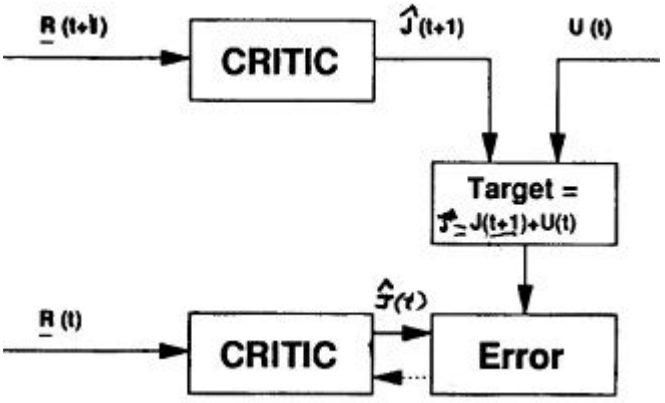

Figure 13 Adapting critic using HDP [3]

2.2.5.1 Implementation of DHP

DHP is a procedure for adapting a critic network or function $\hat{a}(\mathbf{R}(t))$ which attempts to approximate the function $a_i(t)$ defined in equation (2.2.4). Any supervised learning method can be used to adapt the critic in DHP. As use of back-propagation is not required in the supervised learning, convergence speed is not an issue in DHP. Still, the calculation of target vector a^* does use dual subroutines to back-propagate derivatives through the model network and the action network as shown in figure 14. DHP can be implemented by as follows [3]:

1. Obtain $\mathbf{R}(t), \mathbf{u}(t)$ and R(t+1) as was done with HDP.

2. Calculate:

$$\hat{a}(t+1) = \hat{a}(\mathbf{R}(t+1), \mathrm{W}) \tag{2.2.6}$$

$$\mathbf{F}\_\mathbf{u}(t) = F\_U_{\mathbf{u}}(\mathbf{R}(t), \mathbf{u}(t) + F\_f_{\mathbf{u}}(\mathbf{R}(t), \mathbf{u}(t), \hat{a}(t+1)) \quad (2.2.7)$$

$$a^*(t) = F\_f_{\mathbf{R}}(\mathbf{R}(t), \mathbf{u}(t), \hat{a}(t+1)) + F\_U_{\mathbf{R}}(\mathbf{R}(t), \mathbf{u}(t)) + F\_A_{\mathbf{R}}(\mathbf{R}(t), \mathbf{F}\_\mathbf{u}(t)) \quad (2.2.8)$$

3. Update W in $\hat{a}(\mathbf{R}(t), W)$ based on the inputs R(t) and target vector $a^*(t)$.

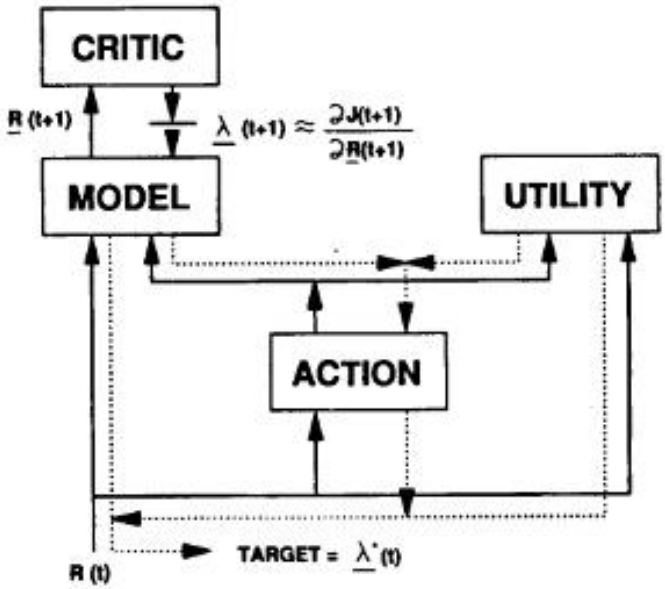

Figure 14 Adapting critic using DHP [3]

References

1. Paul Werbos, “What is Mind? What is Consciousness? How Can We Build and Understand Intelligent Systems?”, Werbos’ website.
2. Paul J. Werbos, “Using ADP to Understand and Replicate Brain Intelligence: the Next Level Design”, IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning, 2007, pp. 209-216.
3. Paul J. Werbos, “Handbook of Intelligent Control: Neural, Fuzzy and Adaptive Approaches”, VAN NOSTRAND REINHOLD, 1992, pp. 493-525.
4. Paul J. Werbos, “Neural networks and the human mind: New mathematics fits humanistic insight”, IEEE International Conference on Systems, Man, and Cybernetics, vol. 1, 1992, pp. 78-83.

2.3 Outline of Biological Functions of Various Regions in the Brain

This section talks about various parts of a human brain involved in decision process, especially the limbic system. Also, their function and location in the brain are discussed.

2.3.1 Introduction

Figure 15 shows the brain regions and their locations together. The following sections illustrate process involvement, inter-connections and locations of amygdala, orbito-frontal cortex, basal ganglia, dorsolateral prefrontal cortex, anterior cingulate cortex, hippocampus and thalamus.

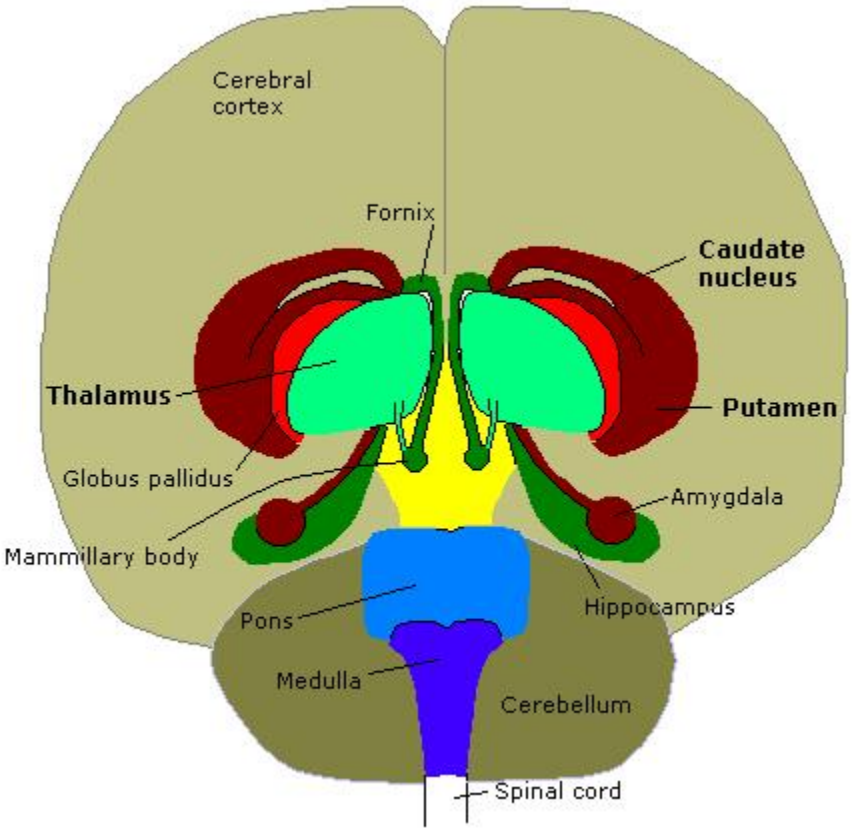

Figure 15 Human brain

2.3.2 Amygdala

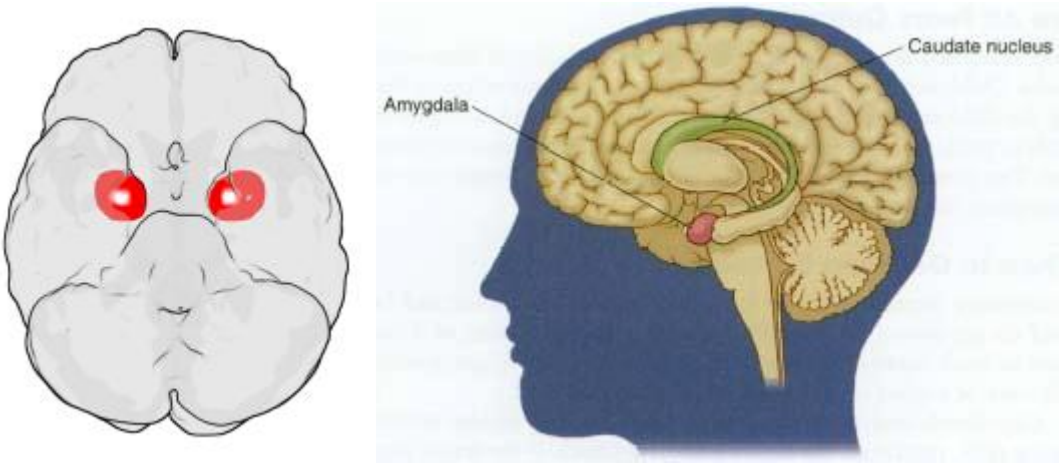

Figure 16 Location of amygdala: top view (left image), side view (Right image)

Amygdala is an almond-shaped group of nuclei which is located deep and medially within the temporal lobes of the brain. It is a part of the limbic system. It has connections with hypothalamus and dorsomedial thalamus. It is found to be primarily involved in memory processing, decision-making, and emotional reactions. The right and left amygdala perform different functions. It has been found that the right amygdala induces negative emotions, especially fear and sadness; while the left amygdala induces either pleasant (happiness) or unpleasant (fear, anxiety, sadness) emotions and is involved in the brain's reward computation. The amygdala is involved in the formation and storage of memories associated with emotions [1].

2.3.3 Orbito-Frontal Cortex

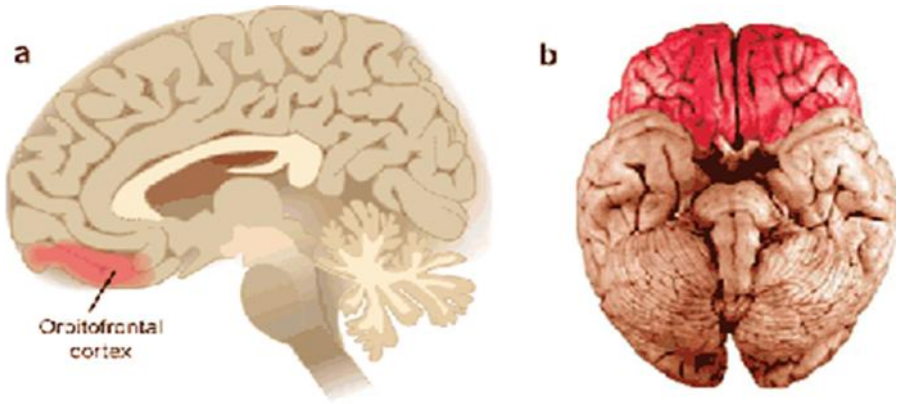

Figure 17 OFC: (a) Side view (b) Top view

The orbitofrontal cortex (OFC) is a prefrontal cortex region in the frontal lobes in the brain. It is located immediately above the eyes. It is a part of the prefrontal cortex that receives projections from the magnocellular, medial nucleus of the mediodorsal thalamus. It is involved in the cognitive processing of decision-making. Sensory cortices additionally share highly complex reciprocal connections with the orbitofrontal cortex. All sensory modalities are represented in connections with the orbitofrontal cortex, including extensive innervations from areas associated with olfaction and gustatory somatic responses. It is also connected with amygdala, hippocampus, striatum and hypothalamus. There is suggestion of a role for the orbitofrontal cortex in both inhibitory and excitatory regulation of autonomic function. The cortico-striatal networks seem to be involved in the processing of goal-directed and habitual action, cortico-limbic connection for a role in action selection, and the integration of information into behavioral output. It is involved with amygdala in representation of emotion and in decision making [1].

2.3.4 Basal Ganglia

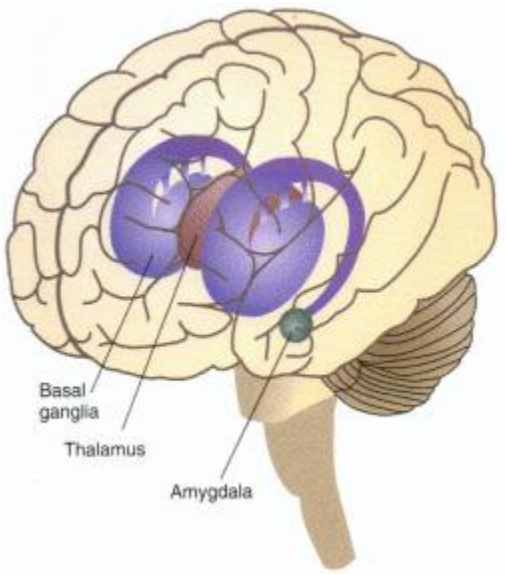

Figure 18 Location of basal ganglia in the brain

The basal ganglia comprises of multiple subcortical nuclei of varied origin in the brains of vertebrates. It is located at the base of the forebrain. It is strongly interconnected with the cerebral cortex, thalamus, and brainstem, as well as several other brain areas. The basal ganglia is associated with a variety of functions including: control of voluntary motor movements, procedural learning, routine behaviors or "habits" such as bruxism, eye movements, cognition and emotion. It has been hypothesized that the basal ganglia is also involved in action selection. It is suggested that the basal ganglia controls and regulates activities of the motor and premotor cortical areas for smooth voluntary movements. Studies show that the basal ganglia influences a number motor systems by inhibition. With signals from other parts of the brain, the basal ganglia performs switching in behavior [1].

2.3.5 Dorsolateral Prefrontal Cortex

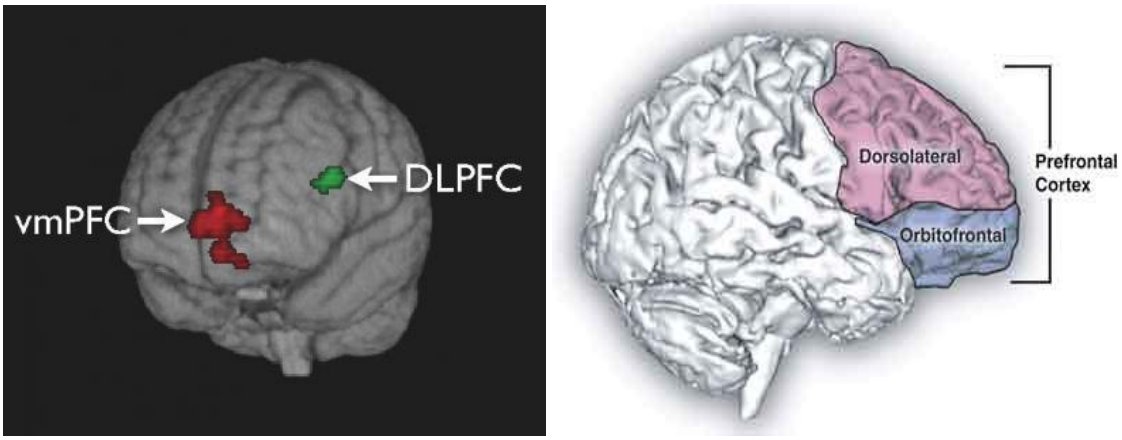

Figure 19 Location of DLPFC

The dorsolateral prefrontal cortex (DLPFC) is an area in the prefrontal cortex of the brain. The prolonged maturation of the DLPFC lasts until adulthood. It is basically a functional region which lays in the middle frontal gyrus of the brain. The DLPFC is connected to orbitofrontal cortex, thalamus, parts of the basal ganglia, the hippocampus, posterior temporal, parietal, and occipital areas. Also, the DLPFC provides methods to interact with the stimuli. It plays role in working memory, cognitive flexibility, planning, inhibition, and abstract reasoning. But, it does require assistance from other cortical and subcortical areas for complex activities like motor planning, organization and regulation [1].

2.3.6 Anterior Cingulate Cortex

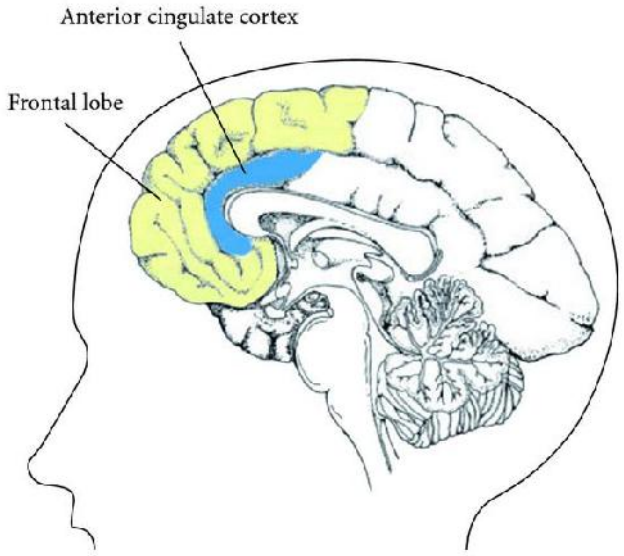

Figure 20 Location of ACC

The anterior cingulate cortex (ACC) is the frontal part of the cingulate cortex. Its shape resembles a "collar" surrounding the frontal part of the corpus callosum. It is suggested to play a role in rational cognitive functions, such as reward anticipation, decision-making, empathy, impulse control, and emotion. It can be divided anatomically based on cognitive (dorsal) and emotional (ventral) components. The dorsal part is connected with the prefrontal cortex, parietal cortex and the motor system so that it acts a center for processing top-down and bottom-up stimuli and assigning appropriate control to other areas in the brain. The ventral part is connected with amygdala, nucleus accumbens, hypothalamus, and anterior insula, so it is involved in assessing the salience of emotion and motivational information. The ACC seems to be especially involved when effort is needed in early learning and problem-solving [1].

2.3.7 Hippocampus

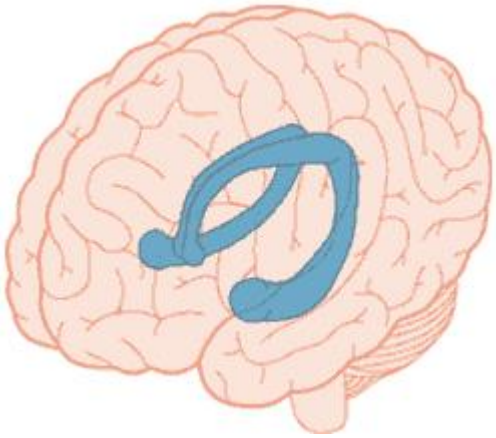

Figure 21 Location of Hippocampus

Hippocampus belongs to the limbic system. It is located under the cerebral cortex and divided in each side of the brain. It is connected with the prefrontal cortex, the septum, the hypothalamic mammillary body, and the anterior nuclear complex in the thalamus. It plays important roles in the consolidation of information from short-term memory to long-term memory and spatial navigation. A form of neural plasticity known as long-term potential occurring in the hippocampus is believed to be one of the reasons of consolidation of the memory. In many studied it has been found that a damage to the hippocampus affects memory function. Also, a perception of location in the environment is affected if the hippocampus is damaged [1].

2.3.8 Thalamus

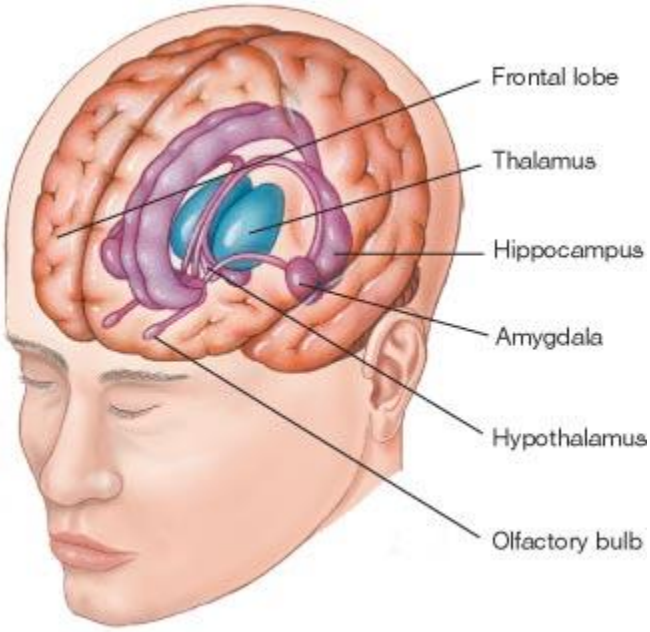

Figure 22 Location of Thalamus

Thalamus is a midline symmetrical structure consisting of two halves. It is located between the cerebral cortex and the midbrain. The thalamus is manifoldly connected to the hippocampus. Its plays a role in relay of sensory and motor signals to the cerebral cortex, and the regulation of consciousness, sleep, and alertness. A sensory tract originating in the spinal cord transmits information to the thalamus about pain, temperature, itch and crude touch. It may be thought of as a kind of switchboard of information. The thalamus is believed to process the sensory information also, and it regulates states of sleep, arousal, the level of awareness and activity [1].

References

1. Amygdala, OFC, Basal Ganglia, DLPFC, ACC and Hippocampus from Wikipedia.

2. R. Nieuwenhuys, J. Voogd, C. van Huijzen, “The Human Central Nervous System”, Forth Edition, Springer (Book), 2008.

Chapter 3

Neuro-cognitive Psychology

3.1 D. S. Levine's Work in Neuro-cognitive Psychology

In this section, Daniel Levine's work in the area of neuro-cognition is discussed. Effects of emotions, probability and risk on decision process are investigated. Also, involvement of brain regions during different types of behaviors are discussed. Then a decision process model of the brain incorporating gated dipole, adaptive resonance theory and fuzzy trace theory are illustrated.

3.1.1 Introduction

Future autonomous systems require increased speed and dynamical responsiveness of individual and of groups of coordinated multiple platforms. Due to large data availability, novel decision and control schemes are required that focus relatively on the data that are relevant for the current situation and ignore relatively unimportant details. Asymmetric human-robot systems and demands for fast response impose new requirements for fast and efficient decision, interaction and control in large distributed teams with autonomous dynamical subsystems. Streamlined and fast mechanisms that deliver prescribed aspiration levels of satisfactory results are observed in nature and in neuro-cognitive studies of human brain. Here, rigorous modeling of mechanisms for fast satisficing, risk, gist and emotional triggers based on new developments in cognitive neuroscience has been studied. They will be used to develop new structures of automatic control systems that are capable of fast satisficing, dynamic focusing of awareness and reduced response times in networked environments.

3.1.2 Emotion and Decision Making

3.1.2.1 Short-term Reactions versus Long-term Evaluations

Emotion affects decisions at various times: a guide to information, a selective attentional spotlight, a motivator of behavior and a common currency of comparing alternatives. While discussing the relationship of emotion and cognition in the human brain, ‘short-term emotional reactions’ are needed to be distinguished from ‘long-term emotional evaluations’ [1].

Short-term emotional reactions are related with changes in affective values of rewarding or punishing stimuli whether it arrives, is removed or changes in intensity; while long-term emotional evaluations are related with handling the actual affective values of the stimuli rather than changes in those values. The stimuli either keeps a constant positive or negative value over time or the value is averaged. Both systems are equally important: The short-term processing system facilitates effective adaption to sudden salient changes in the environment; while the long-term emotional processing system it provides effective sensitivity to almost constant attributes in the environment [1].

3.1.2.2 Probabilistic Choices

One aspect of human decision making is the nonlinear weighting of probabilities. It is observed that decision makers overweight low nonzero probabilities and underweight low nonzero probabilities when gambles are explicitly described A low nonzero probability of obtaining an affect-rich resource (1% probability of obtaining kiss) is more strongly over-weighted than the same low probability of obtaining an affect-poor resource (1% probability of obtaining $50) [4]. But, when decisions are made from experience (learning through feedback), the decision makers typically underweight low probabilities rather than overweighting them! [5].

3.1.2.3 Rational versus Irrational Choices

The work of Reyna and Brainerd suggested ‘fuzzy trace theory’ (FTT) that humans encode information in two different ways: ‘verbatim’ and ‘gist’ encoding. Verbatim encoding

means perception of literal meaning, facts or numerical values; while gist encodes only essential meaning, intuition. Faster and efficient decision can be achieved when unimportant, minor details are neglected and the gist of a problem is grasped along with comparison with previously encountered problems. However, sometimes gist processing using heuristics can lead to errors.is also a main source of heuristics that can sometimes lead to errors, making irrational choices instead of rational ones [3]. Methods of gist encoding are unknown and varies within individuals. Emotions do affect encoding of information. So, a typical weight probability function as shown in figure 23 was interpreted by Levine (2011) as a nonlinear average of an all-or-none step function arising from gist encoding and a linear function arising from verbatim coding [2].

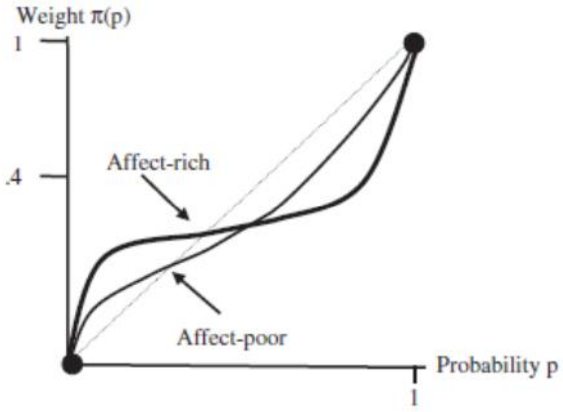

Figure 23 Typical weighing curve [4]

3.1.3 Modeling the Rules of Behavior

According to Levine [6], some behavioral patterns are based on evolution and they prevail in all humans, like the patterns of self-protection and of social bonding behaviors. Everyone has different criteria for time of engagement in a particular behavior. They are heavily affected by learning and by culture; not just by genes. So along with Eisler, Levine proposed cortical-subcortical neural pathways for three separate behavioral patterns: (1) fight-or-flight (figure 24), (2) dissociation (figure 25) and (3) tend-and-befriend (figure 26).

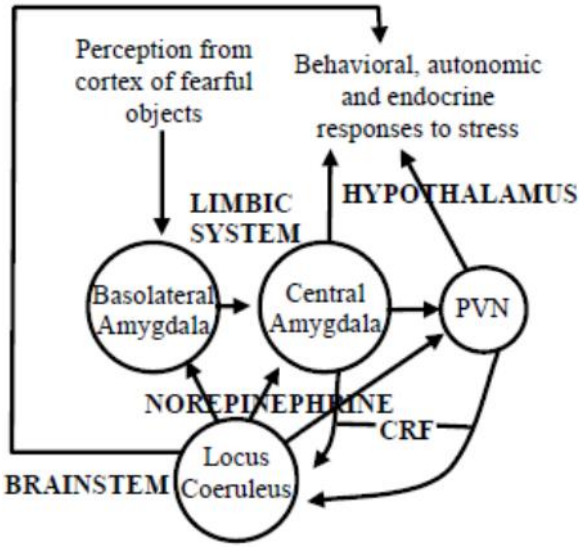

Figure 24 Fight-or-flight path. CRF is a biochemical precursor to a stress hormone [6]

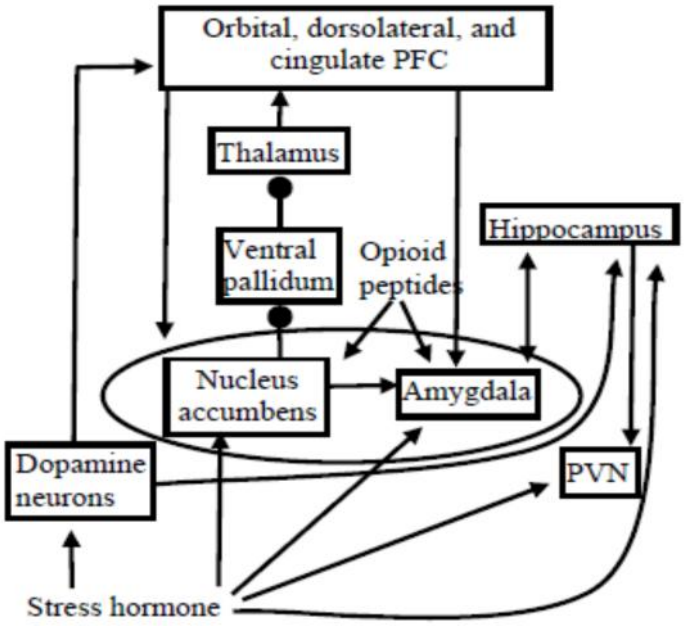

Figure 25 Dissociation pathway. Filled circles denote inhibition. PVN: paraventricular nucleus of hypothalamus [6].

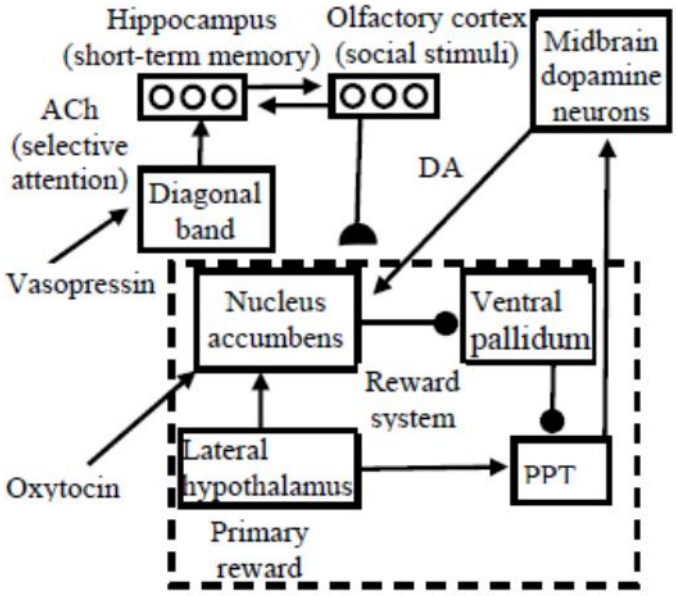

Figure 26 Tend-and-befriend pathway. ACh: acetylcholine, DA: Dopamine. Semicircles denote modifiable synapses [6].

3.1.4 The Brain Model

The process of decision making of the brain is not fully known, but brain areas such as amygdala, orbitofrontal cortex (OFC), basal ganglia, nucleus accumbens (NAcc), thalamus, anterior cingulate cortex (ACC) and dorsolateral prefrontal cortex (DLPFC) along with the neurotransmitter dopamine seems to take part in the decision process. The nervous system in the brain can implement fast and efficient behavioral plans which are flexible and responsive to unexpected changes. Also, new information is encoded without forgetting the older. It is found that there are multiple interacting modules that perform different functions like attribute selection, categorization and memory storage. The amygdala and OFC are involved in all emotional response from the most primitive to the most cognitively driven. The OFC-amygdala interaction updates reward or penalty values of a stimuli. The basal ganglia's NAcc converts affective valuations into influences on action. The five loops connecting frontal cortex, basal ganglia and thalamus have long been regarded as "gates" for control of behaviors which are excited by activation of the direct pathway and inhibited by indirect pathway. Contextual information is

provided by hippocampus [3] The ACC is activated when there is a conflict about selection of rules that would govern choices. If higher deliberation and/or low emotional influence is required, ACC activates DLPFC. DLPFC weighs task relevant attributes more heavily and decreases the irrelevant, emotional attributes. Thalamus has been found to play role in selective attention toward attributes [5]. All this has led to three frameworks which together model the decision process: (1) Gated Dipole Network (2) Adaptive Resonance Theory (ART) (3) Fuzzy Trace Theory (FTT).

3.1.4.1 Gated Dipole Network

Grossberg (1972) proposed a neural network mechanism that involves two pathways of antagonistic values as shown in figure 27. The pathways can be thought of as ‘positive’ and ‘negative’ or ‘on’ and off’. Deactivation of an input to one channel leads to transient activation of the other channel and vice versa. In the figure, J is an input, I nonspecific arousal, w1 and w2 are synapses and ‘xi’s are activity nodes. When J is on, then x5 is activated even with depleted w1. After J is shut off and w1 is depleted while w2 does not, x4 becomes more active than x3. This results in activation of x6 and inhibition of x5.By competition, x6 is activated. If no input J is present and both w1 and w2 have same potential, there is no effect [1].

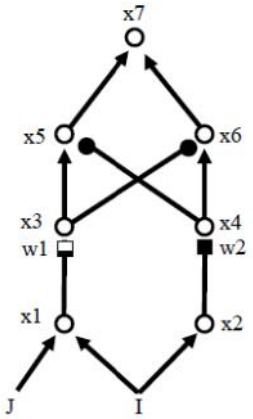

Figure 27 Schematic Gated Dipole. ‘->’ denote excitation, • denote inhibition and partially filled square denote depletion [1].

3.1.4.2 Combining Fuzzy Trace Theory and Adaptive Resonance Theory

The ART is essentially a theory of attribute selection and categorization in multilevel networks. A basic ART network comprises of two interconnected layers of nodes F_1 and F_2. A network proposed by Daniel Levine's which utilizes the ART network is shown in figure 28 with F_1 as amygdala and F_2 as OFC. F_1 and F_2, both contain fields of gated diploes. The nodes at F_1 represent different attributes of the input (gist encoding) and the nodes at F_2 represents different categories of a particular attribute node at F_1. Similar to the amygdala-OFC connections, the synaptic connections between F_1 and F_2 are bidirectional and modifiable [5]. These two layers only classify options of choices with emotional influence. Now, to make choices out of these options, this ART network is connected with another network which involves ACC for action selection, basal ganglia and thalamus for action gating and premotor cortex for execution of action. All of these parts have their local representations of actual options [4]. If there is a match between the input pattern and wining category, then corresponding action is executed with positive feedback using direct pathway. If it is a mismatch, then a "reset" is activated by the ACC and the input is classified into a new category. A parameter called 'vigilance' is used for matching [5].

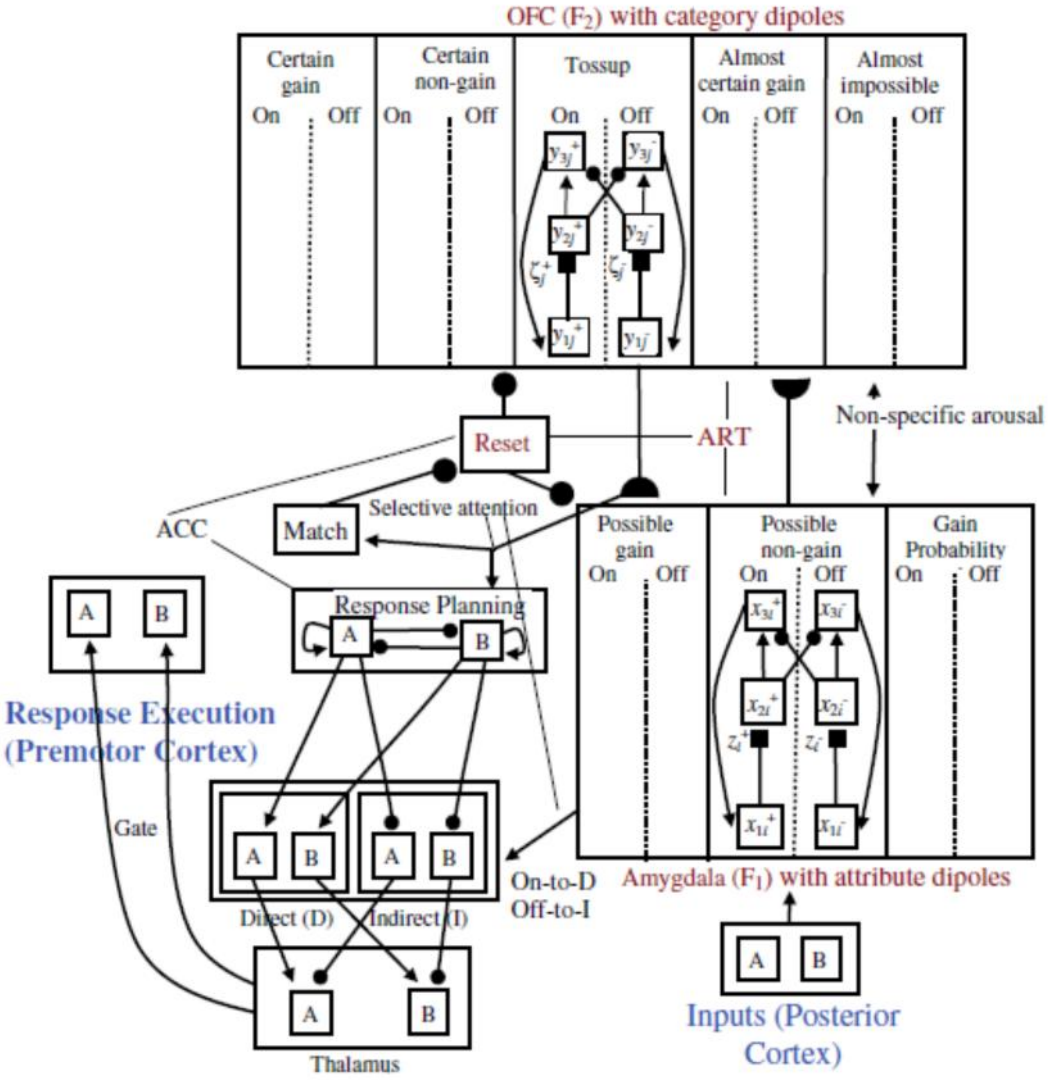

Figure 28 Daniel Levine's network. -> denote excitation, • denote inhibition and partially filled square denote depletion [4].

When input I_k has the attention, where I_1 = A and I_2 = B, the activities of the input nodes x_{1i}^{+} and x_{1i}^{-} in the i[th] amygdalar attribute dipole, i=1, 2, 3, satisfy the equations [4]

$$\frac{dx_{1i}^{+}}{dt} = -x_{1i}^{+} + J_i + bx_{3i}^{+} + \left(2 - .5k\right) \text{¡} - R \qquad (3.1.1)$$

$$\frac{dx_{1i}^{-}}{dt} = -x_{1i}^{-} + bx_{3i}^{-} + \left(2 - .5k\right) ¡ - R \tag{3.1.2}$$

where J_i denotes the i[th] attribute component of the input vector, ¡ denotes non-specific arousal with 2-.5k factor due to 1.5 times higher arousal, R denotes the activity of the reset node and b is the decay proportional activity, and output nodes x_{3i}^{+} and x_{3i}^{-}. The depletable transmitter weights z_i^{+} and z_i^{-} have the following dynamics [4]

$$\frac{dz_i^{+}}{dt} = (1 - z_i^{+}) - x_{1i}^{+} z_i^{+} \tag{3.1.3}$$

$$\frac{dz_i^{-}}{dt} = (1 - z_i^{-}) - x_{1i}^{-} z_i^{-} \tag{3.1.4}$$

The node activity equations at the next level of the attribute dipoles are [4]

$$\frac{dx_{2i}^{+}}{dt} = -x_{2i}^{+} + x_{1i}^{+} z_i^{+} \tag{3.1.5}$$

$$\frac{dx_{2i}^{-}}{dt} = -x_{2i}^{-} + x_{1i}^{-} z_i^{-} \tag{3.1.6}$$

The node activity equations at the dipole output layer are [4]

$$\frac{dx_{3i}^{+}}{dt} = -x_{3i}^{+} + (1 - x_{3i}^{+})\left(x_{2i}^{+} + \sum_{j=1}^{5} y_{3j} w_{ji}\right) - x_{3i}^{+} x_{2i}^{-} \tag{3.1.7}$$

$$\frac{dx_{3i}^{-}}{dt} = -x_{3i}^{-} + (1 - x_{3i}^{-}) x_{2i}^{-} - x_{3i}^{-} x_{2i}^{-} \tag{3.1.8}$$

where y_j denotes activity of the jth category node and w_{ji} denotes the weight of the connection between the jth y_3^{+} category node and the x_3^{+} node corresponding to the ith attribute. The F_2 layer has similar activity and transmitter weight equations for the category dipoles, with y_{1j}^{+} and y_{1j}^{-} being input nodes, ¸$_j^{+}$ and ¸$_j^{-}$ depletable transmitters, y_{2j}^{+} and y_{2j}^{-} layer-2 nodes, and y_{3j}^{+} and y_{3j}^{-} output nodes for j=1,…,5 [4].

The weights are solved for the following equation [4],

$$\frac{dw_{ji}}{dt} = \sim((y_{3j}^{+})^{2})(-w_{ji} + x_{3i}^{+}) \quad (3.1.9)$$

The reset node activity is defined by [4]

$$\frac{dR}{dt} = -R + \min_{j=1}^{5} MATCH(\mathrm{j}) \quad (3.1.10)$$

where MATCH(j) is a measure of closeness between the normalized weight and input vectors [4]:

$$MATCH(j) = \sum_{i=1}^{3} m_{ki} \left(NORMWTS_{ji} - NORMINPUT_{i} \right)^{2} \quad (3.1.11)$$

where $NORMWTS_{ji} = \dfrac{w_{ji}}{\sqrt{\sum_{m=1}^{3} \left(w_{jm} \right)^{2}}}$,

$NORMINPUT_{i} = \dfrac{I_{i}}{\sqrt{\sum_{m=1}^{3} \left(I_{m} \right)^{2}}}$ and

K: Input corresponding to the planning node activity.

3.1.4.3 Extension of the brain model

Figure 29 shows the extended neural network model of the brain that incorporates gated, dipoles, FTT and ART, with some addition. It has two layers of OFC: A superficial layer OFC1 that interacts with amygdala and, a deeper layer OFC2 that has representation of categories and receives motivational signals from medial prefrontal areas. It includes dopamine as a positive reinforcer to gate direct pathways (activation) and serotonin as a negative reinforce to gate indirect pathways (inhibition) [1].

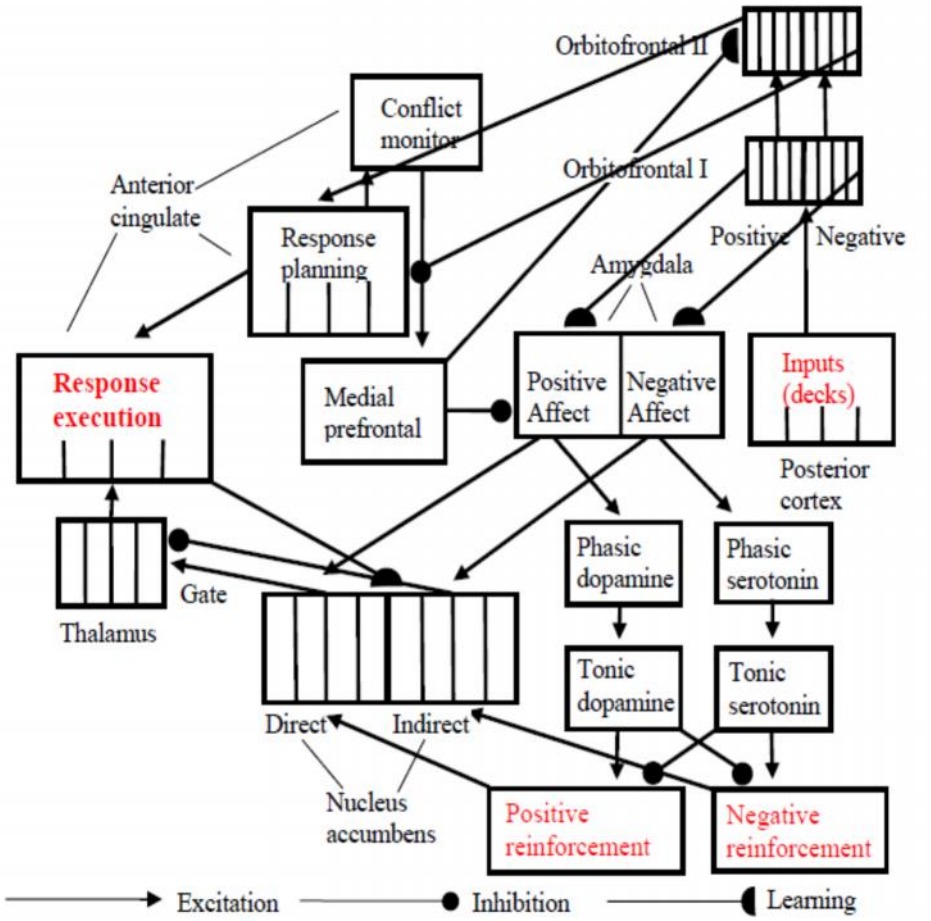

Figure 29 Neural Network Model of Brain [1]

References

1. Daniel S. Levine, “Emotion and Decision Making: Short-term Reactions versus Long-term Evaluations”, International Joint Conference on Neural Networks, 2006, pp. 195-202.

2. Daniel S. Levine, Leonid I. Perlovsky, “A Network Model of Rational versus Irrational Choices on a Probability Maximization Task”, IJCNN, 2008, pp 2820-2824.

3. Daniel S. Levine, Britain Mills, Steven Estrada, “Modeling Emotional Influences on Human Decision Making under Risk”, IJCNN Special Session, Vol. 3, 2005, pp. 1657-1662.

4. Daniel S. Levine, “Neural Dynamics of Affect, Gist, Probability, and Choice”, Cognitive System Research, Science-Direct, 2012, pp. 57-72.

5. Daniel S. Levine, P. Ramirez, “An Attentional Theory of Emotional Influences on Risky Decisions”, Progress in Brain Research, Vol. 202, Elsevier (Book), 2013.

6. Daniel S. Levine, “Modeling the Evolution of Decision Rules in the Human Brain”, International Joint Conference on Neural Networks, 2006, pp. 625-631.

7. Daniel S. Levine, “Seek Simplicity and Distrust it: Knowledge Maximization versus Effort Minimization”, Plenary talk (PPT), KIMAS, 2007.

3.2 Third Generation Brain like Intelligence and Approximate Dynamic Programming

The second section of the second chapter discussed the first and second generation models of the brain prescribed by Paul Werbos. In this section, a third generation model of the brain is discussed from his work [1]. He partly agreed with artificial intelligence researchers like Albus that the human brain has highly complex hierarchical structures to handle a high degree of complexity in space and in time, because faster learning can be achieved with modified Bellman equations which use the hierarchical partitioned state space. Though this idea of hierarchy is not supported by new biological data, there is hypotheses of some kind of specific mechanisms in three core areas: (1) a “creativity/imagination” mechanism that deals with the complex, non-convex optimization problem, (2) a mechanism to take care of equations coping with multiple time scale decisions (3) a mechanism to handle spatial complexity. So he proposed a Strawman model of the creativity mechanism in 1997 as shown in figure 30 [1].

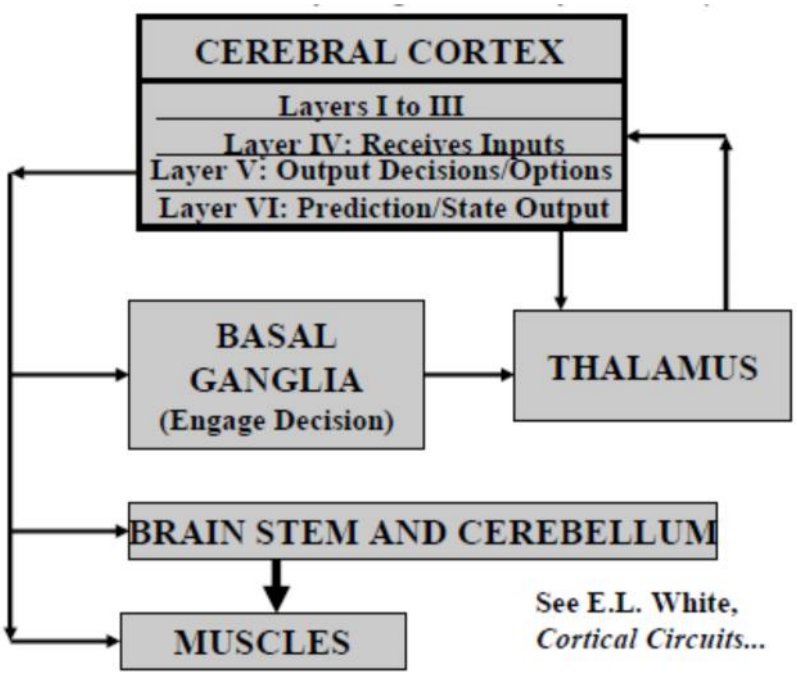

Figure 30 Third generation brain model of creativity [1]

Yet, much progress is to be made to have brain-like stochastic search. One reason for it is the association of temporal complexity with spatial complexity. Guyon at AT&T had developed

the most accurate ZIP code digit recognizer that utilized spatial symmetry with the help of modified multilayer perceptron network (MLP), though it could not segment an entire ZIP code because of failure to handle connectedness. Pang and Werbos (1994) proposed a network called "Cellular SRN'' (CSRN), integrating the key capabilities of a Simultaneous Recurrent Network and a "conformal'' network. Unlike MLP, it could learn to predict, control and navigate through far more complex planes. One of the reasons it could not be used widely is availability of fast learning tools. Then, however, Ilin, Kozma and Werbos (2008) reported a very fast learning tool [1].

In 1997 and subsequently, Werbos proposed an approach to exploit symmetry called the ObjectNet. Here, a complex input field is mapped into a network of k types of "objects" which have inner loops of k different types. This is the opposite of mapping inputs into M rectangular cells governed by a common inner loop neural network. Without any assistance from human and without using a supercomputer, a simple computer system having a simple ObjectNet, which was designed by David Fogel could achieve master class performance [1].

A brain-like ability is needed to be achieved that can learn more complex transformations than simple two-dimensional transformations. The CSRN may be exploited for this. But, the first problem here is to learn symmetries that is, to learn a family of vector maps f_a such that [1]

$$\Pr\left(\frac{f_{\sim}(\underline{\mathbf{x}}(t+1))}{f_{\sim}(\underline{\mathbf{x}}(t))}\right)=\Pr\left(\frac{\underline{\mathbf{x}}(t+1)}{\underline{\mathbf{x}}(t)}\right) \tag{3.2.1}$$

for all $\sim$ and the same conditional probability distribution Pr. This concept is called stochastic invariance. In simple words, the probability of observing $\underline{\mathbf{x}}(t+1)$ after observing $\underline{\mathbf{x}}(t)$ should be the same as the probability of observing the transformed version of $\underline{\mathbf{x}}(t+1)$ after observing the transformed version of $\underline{\mathbf{x}}(t)$. These symmetries can be exploited in one of the following ways once learned [1]:

1. "Reverberatory generalization": after observing or remembering a pair of data $\{\underline{\mathbf{x}}(t+1), \underline{\mathbf{x}}(t)\}$, also train on $\{ f_{\sim}(\underline{\mathbf{x}}(t+1)), f_{\sim}(\underline{\mathbf{x}}(t))\}$.
2. "Multiple gating": after inputting $\underline{\mathbf{x}}(t)$, pick $\sim$ so as to use $f_{\sim}$ to map $\underline{\mathbf{x}}(t)$ into some canonical form, and learn a universal predictor form canonical forms.
3. "Multimodular gating": similar to multiple gating, except that multiple parallel copies of the canonical mapping are used in parallel to process more than one subimage at a time in a powerful way.

In 1992, a new architecture called "Stochastic Encoder/Decoder Predictor" (SEDP) was proposed which extends the ObjectNet theory. SEDP directly learns condensed mappings wit symmetry relations. It can be thought as an adaptive nonlinear generalization of Kalman filtering. It still requires methods to speed up the learning process [1].

3.2.1 Stochastic Encoder/Decoder Predictor

As discussed in [2], a stochastic model can be defined by,

$$X_i(t+1) = \hat{X}_i(t+1) + \pm_i e_i(t+1) \tag{3.2.2}$$

where e_i represents random noise of unit variance and $\pm_i^2$ represents the variance of the error in predicting X_i. Yet, equation (3.2.2) is not a general stochastic model. It assumes that the matrix $\langle ee^T \rangle$ is diagonal, and the observed variables always follows a normal distribution. Then, Werbos suggested that we consider a more general model that may be written as:

$$X_i(t+1) = \pm_i^X e_i^X(t+1) + D_i(\mathbf{R}(\mathrm{t}+1), \mathrm{information(t)}) \tag{3.2.3}$$

$$\mathbf{R}_i(t+1) = \pm_i^{\mathbf{R}} e_i^{\mathbf{R}}(t+1) + P_i(\mathrm{information(t)}) \tag{3.2.4}$$

where X is observed and R is not; R is the estimate of the state vector, D stands for "Decoder" and P for "Predictor". This can produce any pattern of noise. The problem is adaption of the networks D and P when R is unknown. The classical likelihood function for this problem involves

performing a Monte Carlo integration/simulation based on equations (3.2.3) and (3.2.4), which is very inefficient. This section will discuss a more efficient design [2].

The Stochastic Encoder/Decoder Predictor design is illustrated in figure 31. It is assumed that X_i equals $\hat{X}_i$ plus some Gaussian white noise. First, information is provided from time $t-1$ into the Predictor network, and the Predictor network calculates $\hat{\mathbf{R}}(t)$. Then, the Encoder network inputs $\mathbf{X}(t)$, along with any information available from time $t-1$. The output of the Encoder network is a vector $\tilde{\mathbf{R}}$, a kind of a prediction of the true value of $\mathbf{R}$. Next, generate simulated values of $\mathbf{R}$, $\tilde{\mathbf{R}}'$ are generated by adding random numbers to each component $\tilde{R}_i$. Finally, the Decoder network generates a prediction of X from the $\tilde{\mathbf{R}}'$, along with information from time t -1. These calculations varies according to the weights of Encoder, Decoder, and Predictor networks, and according to the estimates of $\pm_i^{\mathbf{R}}$ and $\pm_i^{\mathbf{X}}$. [2]

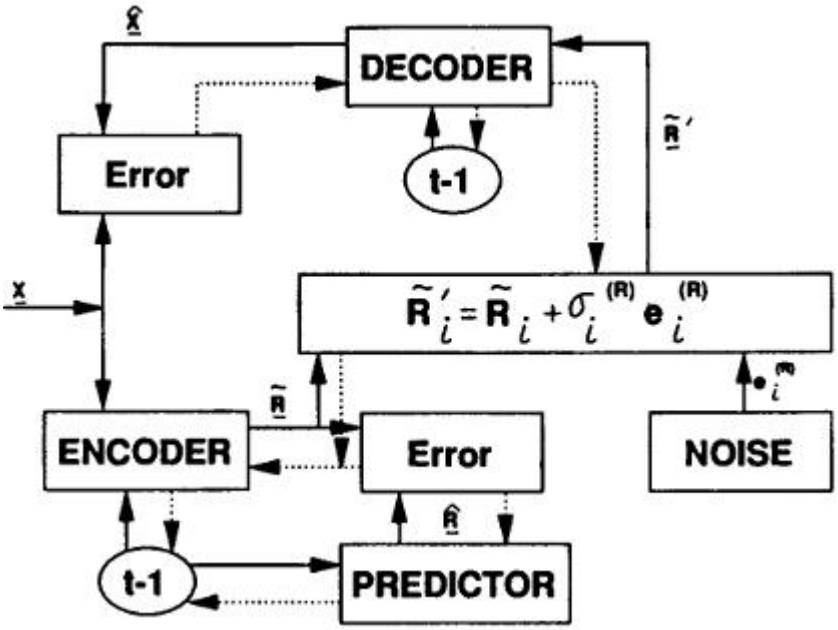

Figure 31 The SEDP [2]

3.2.1.1 Adaption of the System

It is difficult to adapt the Encoder network. All parts of the network are adapted together to minimize [2]:

$$E = \sum_i \left(X_i - \hat{X}_i(\mathbf{R}'(\mathrm{x}, \pm^{\mathbf{R}}, \mathrm{e}^{\mathbf{R}})) \right)^2 / (\pm_i^X)^2 + \sum_j \left(\hat{R}_j - \tilde{R}_j \right)^2 + \sum_i \log(\pm_i^X)^2 + \sum_j \left((\pm_j^{\mathbf{R}})^2 - \log(\pm_j^{\mathbf{R}})^2 \right) \quad (3.2.5)$$

This requires the use of backpropagation-gradient based learning to adapt the Encoder and the parameter $\pm_i^{\mathbf{R}}$. The calculated gradients feed the prediction errors back used for adaption as shown by the dashed line in figure 31. For the encoder, the relevant derivative of E with respect to $\tilde{R}_j$ is [2]:

$$F\_\tilde{\mathrm{R}}_j = 2(\tilde{\mathrm{R}}_j - \hat{R}_j) + F\_\hat{R}'_j \quad (3.2.6)$$

where the first term results by differentiating the R-prediction error in equation (3.2.5) with respect to $\tilde{R}_i$ and the second term represents the derivative of the X-prediction error which is computed by back-propagation through the Decoder network back to $\tilde{R}_j$. The Encoder network is adapted by propagating the $F\_\tilde{\mathbf{R}}$ derivatives back through the Encoder network. The resulting $\mathbf{R}$ from equations (3.2.5) and (3.2.6) is both predictable from the past and useful in reconstructing the observed variables X_i. If the Predictor network is deleted which results in kind of feature extractor, then the variance of $\mathbf{R}$ has to be minimized to prevent a kind of indirect bias or divergence from sneaking in. For a similar purpose, the parameters $\left(\pm_j^{\mathbf{R}}\right)^2$ are adapted based on [2]:

$$\frac{\partial E}{\partial \left(\pm_j^{\mathbf{R}}\right)^2} = F\_R'_j * e_j + \frac{\partial}{\partial \left(\pm_j^{\mathbf{R}}\right)^2} \left(\left(\pm_j^{\mathbf{R}}\right)^2 - \log\left(\pm_j^{\mathbf{R}}\right)^2 \right) \quad (3.2.7)$$

The networks are adapted by using some information calculated at time t-1 along with back-propagating the derivatives of equation (3.2.5) to the information producing network [2].

Figure 32 shows a block diagram of various parts of the brain which are involved in decision process. It is based on discussions of the second and third chapters.

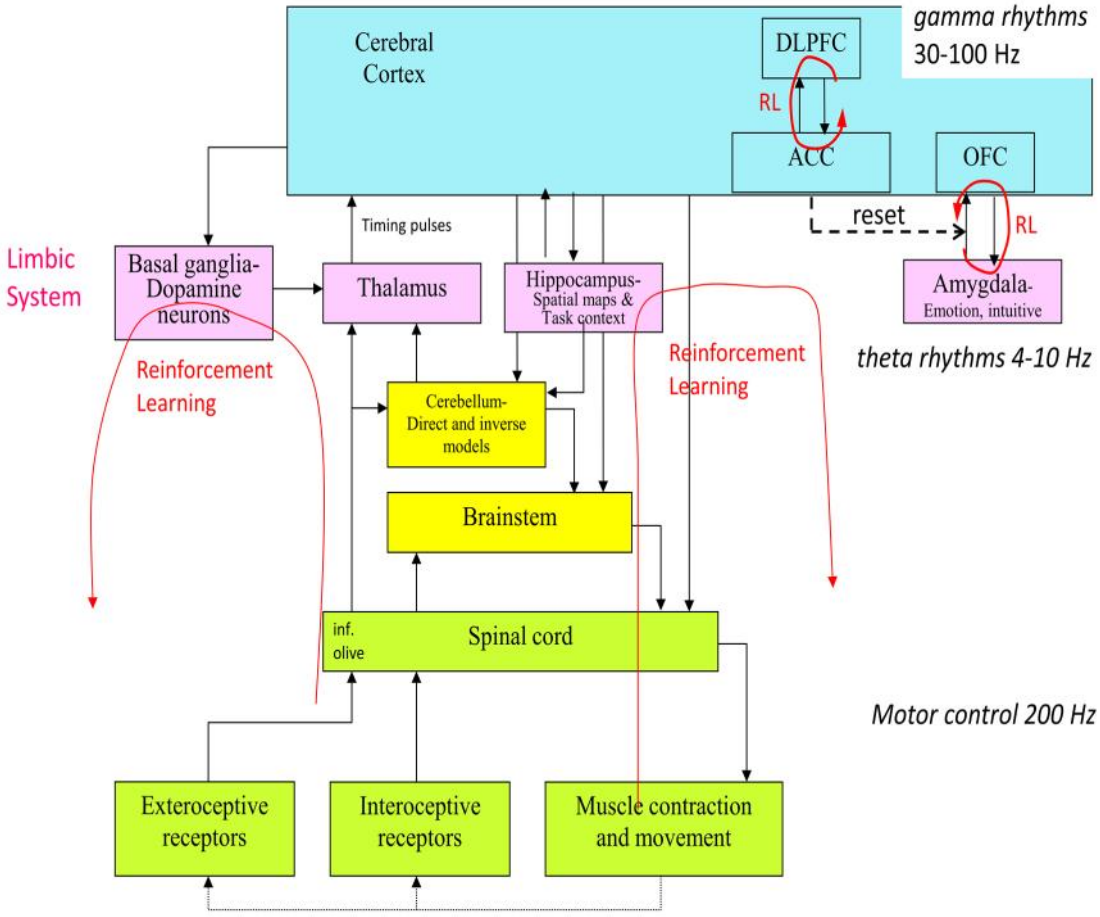

Figure 32 Summary of Human Nervous System

References

1. Paul J. Werbos, “Intelligence in the brain: A theory of how it works and how to build it”, Neural Networks 22, 2009, pp. 200-212.

2. Paul J. Werbos, “Handbook of Intelligent Control: Neural, Fuzzy and Adaptive Approaches”, VAN NOSTRAND REINHOLD, 1992, pp. 493-525.

3.3 Cognitive Development with a Psychological Perspective

3.3.1 Introduction

This section talks about studies done similar to Daniel Levine's to understand the decision process of the human brain. It includes rule process used for decision [2], types of the rules and decisions made under risk [1]. Also Piaget's theory of cognitive development is explained [3, 4].

3.3.2 Decision Making with Rules

Jansen, Duijvenvoorde and Huizenga in [2] mainly worked in order to outline the trajectory of integrative versus sequential rule use in decision making. People make decisions using basically two kinds of rules: (1) Sequential rules in which decision is made by evaluating dimensions of choices sequentially. (2) The integrative normative rule in which decisions are made by integrating the choice dimensions. In case of the sequential rule, a particular dimension is chosen and options are compared based only on this dimension. A decision is reached if options are different on this dimension; otherwise another dimension is considered. So, dimensions are processed one by one and not integrated like multiplication of two dimensions.

Jansen and the others [2] administered the Gambling Machine Task (GMT) to spanning a broad age range of people. They collected data based on the rules used for decision and number of dimensions used for the rules. Figure 33 shows results of their experiment. It can be observed that use of integrative rule decreases and use of sequential rule increases with increase in the age which contradicts proportional reasoning theory and supports fuzzy trace theory. Also, it can be observed that number of dimensions considered increase with increase in age which supports the proportional reasoning theory. Faster and less effortful decisions can be made by using the sequential rule as compared to using integrative rule, but it may sometimes lead to wrong

decisions.

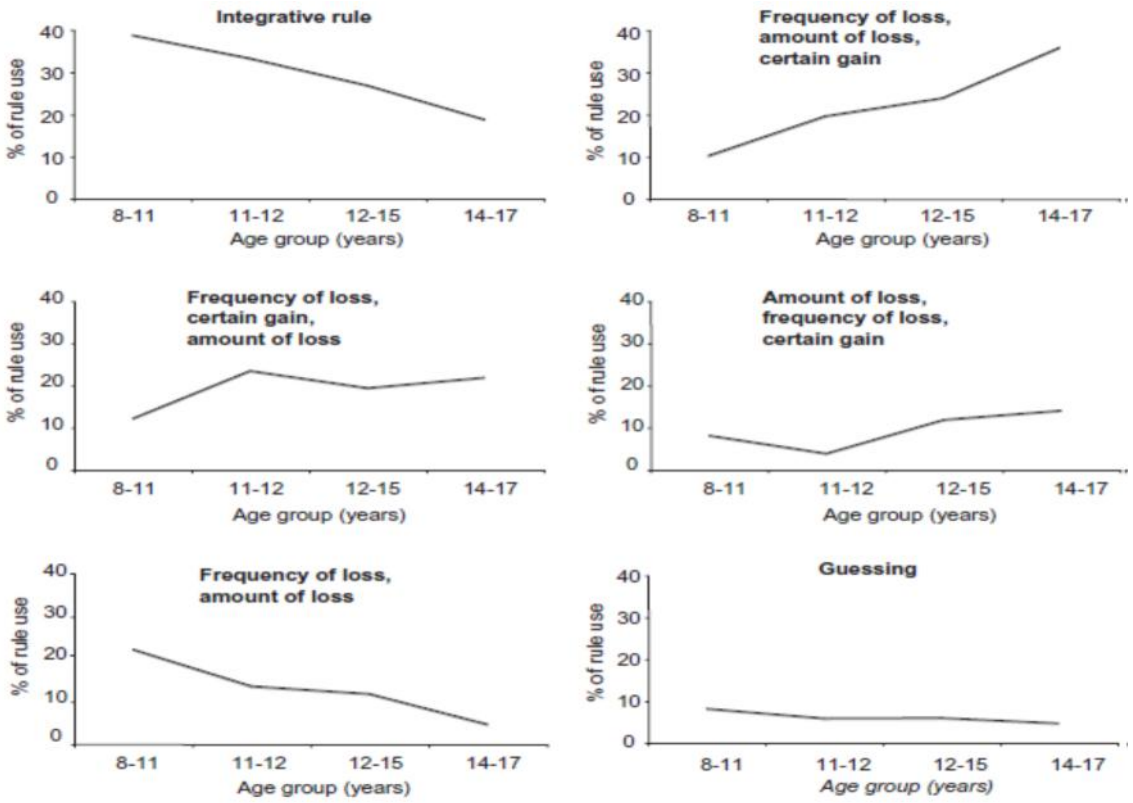

Figure 33 Specific rules used by participants as a function of age group [2]

3.3.3 Piaget's Theory of Cognitive Development

Jean Piaget's theory of cognitive development [3] is a theory about the nature of knowledge and the way humans develop their intellect. Piaget perceived cognitive development as a progression in mental processes due to biological development and experience. A developmental stage of cognition towards intelligence consists of a period of months or years. The development level and rate vary from a person to person. Though incorrect estimation of abilities of young children and older learners by Piaget has been criticized, his theory has facilitated a ways to accelerate cognitive development. There are four primary stages of development according to Piaget: sensorimotor, preoperational, concrete operational and formal operational [4].

3.3.3.1 Sensorimotor Stage

First stage in cognitive development is the sensorimotor stage. It spans from birth until the acquisition of language (till the age of 2 years). Object permanence is one of the most important aspect of this stage in which the child understands that objects can exist whether they can be seen, heard, touched or not. It has been found that children can link numbers and counting to the objects. [4] Of course, children learn to control and coordinate their senses and motor actions [3].

3.3.3.2. Preoperational Stage

The preoperational stage starts when the child begins to speak at the age of 2 and lasts until the age of 7. The child's language ability increases in this stage. The child can perceive and represent symbols related with objects. The child has a little logic and cannot perceive from more than one viewpoint [3, 4]. For example, when same amount of liquid from one container is transferred to another container with less height, the child thinks that amount of liquid has decreased in the later container [4].

3.3.3.3 Concrete Operational Stage

This third stage occurs generally between the age of 7 and 11. The child undergoes a remarkable cognitive growth. In the child, Language development and basic skill acquisition accelerate dramatically. Children at this stage can now perceive other viewpoints, can consider two or three dimensions simultaneously. They can classify objects based on a common characteristic like number, mass and weight. The classification can also be done in ascending or descending order. They are now able to think logically, can reverse an operation-can perform addition and subtraction. [3, 4].

3.3.3.4 Formal Operational Stage

This the fourth stage of cognitive development. The children at this stage are capable of hypothesizing and deduction which is needed in science and mathematics. Abstract thoughts emerge with logical use of symbols in this stage. The children can identify and analyze elements of a problem, so that they can decipher the information needed in solving a problem. Additionally, they can evaluate the adequacy of a problem solution using some criteria. Thus, the children can relate mathematical concepts to real life situations [4].

3.3.4. Cognition and Decision under Risk

Cokely and Kelley [1] investigated the relationship between cognitive abilities and superior decision making under risk. Some mechanisms try to provide this link. One way is to calculate expected value to make expected value choices (Fredrick, 2005). Working memory capacity (Stanovich and West, 2000) and numeracy – understanding of probabilities (Peters & Levin, 2008; Peters et al., 2006) may lead to individual differences in the choices in various conditions. The priority heuristic model of decision hypothesize that decisions between sure and risky options are made by considering simple reasons in a fixed order until a stopping rule is met. Then Cokely and Kelley discuss a dual process model which includes priority heuristics and a simple heuristic processes. It is assumed in this model that controlled cognition relates to more rule-based, abstract and decontextualized reasoning; while more automatic and impulsive cognition is relate to associations, personal relevance, and situational contextual information.

In the experiment, they found out that more elaborative heuristic search was performed in decision making under risk as opposed to expected value calculations which is not a computation of an answer; but a monitored and corrected output of an automatic process. The priority heuristic also was proven wrong. Variations in risky choices are linked to differences in duration and type of information search. Therefore, it is suggested that elaborative heuristic search which involves more thorough exploration and representation, is related to superior risky decision making which yields faster and correct reasons most of the time; but not always.. It is not necessary higher

performing individuals always search or reflect more. Yet, the decision process is not well understood [1]

References

1. Edward T. Cokely, Colleen M. Kelley, “Cognitive abilities and superior decision making under risk: A protocol analysis and process model evaluation”, Judgment and Decision Making, Vol. 4, No. 1, February 2009, pp. 20–33.

2. Brenda R. Jansen, Anna C. Duijvenvoorde, Hilde M. Huizenga, “Development of Decision Making: Sequential versus integrative rules”, Journal of Experimental Child Psychology 111, 2012, pp. 87-100.

3. Piaget’s theory of cognitive development-from Wikipedia.

4. Bobby Ojose, “Applying Piaget’s Theory of Cognitive Development to Mathematics Instruction”, The Mathematics Educator, Vol. 18, No. 1, 2008, pp. 26-30.

Chapter 4

New Neuro-inspired Architectures for Learning and Control

4.1 Neuro-inspired Networks for Learning

4.1.1 Introduction

Various sections before have discussed about how decision process takes place in the human brain with biological and psychological findings. Also, a couple of models of the brain were discussed. This section discusses about various models which were inspired from the decision process of the brain for learning and control purposes. It includes reinforcement learning ([1] and [2]), and artificial neural networks ([4] and [5]).

4.1.2 Shunting Inhibitory Artificial Neural Networks

Bouzerdoum [5] proposed biologically inspired Shunting Inhibitory Artificial Neural Networks (SIANNs) in which a nonlinear shunting inhibition mechanism mediates the synaptic interactions among neuron. One or more number of hidden layers can have neurons which use shunting inhibition. The outputs of these hidden layer neurons are linearly combined to form the output. With help of the inherent nonlinearity, the SIANN is capable of constructing very complex nonlinear decision boundaries which can be used for classification and function approximation. This requires to have an efficient training algorithm. The SIANN has already been used successfully as adaptive filters and for pattern recognition.

4.1.2.1 Definition of SIANN

The dynamics of a feedback shunting inhibitory neural network can be described as follows [5]:

$$\frac{dx_i}{dt} = I_i - a_i x_i - f\left(\sum_j c_{ij} x_j\right) x_i + b_i \qquad (4.1.1)$$

where x_i is the activity of the ith neuron; I_i is the external excitatory input; a_i is a positive constant representing the passive decay rate of the neuron activity; c_{ij} is the connection weight from the jth neuron to the ith neuron; b_i is a constant bias; and f is a positive activation function.

The external inhibitory input I_j replaces he activation of the jth neuron, x_j in feedforward network which has the following dynamics [5]:

$$\frac{dx_i}{dt} = I_i - a_i x_i - f\left(\sum_j c_{ij} I_j\right) x_i + b_i \tag{4.1.2}$$

The steady-state response of the feed-forward network is given by [5],

$$x_i = \frac{I_i + b_i}{a_i + f\left(\sum_j c_{ij} I_j\right)} \tag{4.1.3}$$

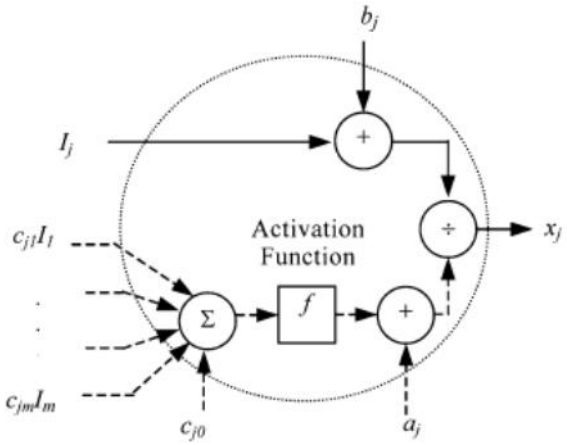

Figure 34 Steady state model of a shunting neuron [4]

The final output y linearly combines the outputs of the shunting neurons through an activation function g [5],

$$y = g\left(\sum_i w_i x_i + b\right) \tag{4.1.4}$$

where w_i is a connection weight and b is a bias term. Figure 35 shows an SIANN with m inputs, one hidden layer and n outputs.

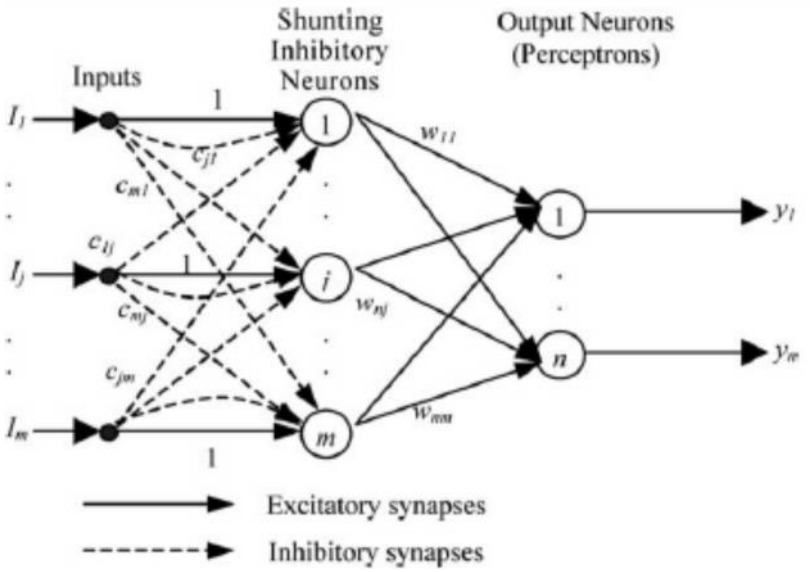

Figure 35 Feedforward SIANN [4]

4.1.2.2 Decision Regions of SIANN

Let us consider a simple SIANN with two neurons in the shunting (hidden) layer. The activations of the shunting neurons are [5],

$$x_1 = \frac{I_1 + b_1}{a_1 + f\left(c_{11}I_1 + c_{12}I_2\right)}, x_2 = \frac{I_2 + b_2}{a_2 + f\left(c_{21}I_1 + c_{22}I_2\right)} \tag{4.1.5}$$

and the output of the SIANN is [5],

$$y = g(\mathrm{w}_1 x_1 + w_2 x_2 + b) \tag{4.1.6}$$

The decision boundary of this network is given by [5],

$$\mathrm{w}_1 x_1 + w_2 x_2 + b = 0 \tag{4.1.7}$$

Now, quadratic decision boundaries can be constructed by selecting a linear activation function. A nonlinear activation function would construct more complex decision surfaces. Figure 36 illustrates the decision regions of the above mentioned SIANN with the logistic activation function $f(x) = 1/(1+e^{-x})$.

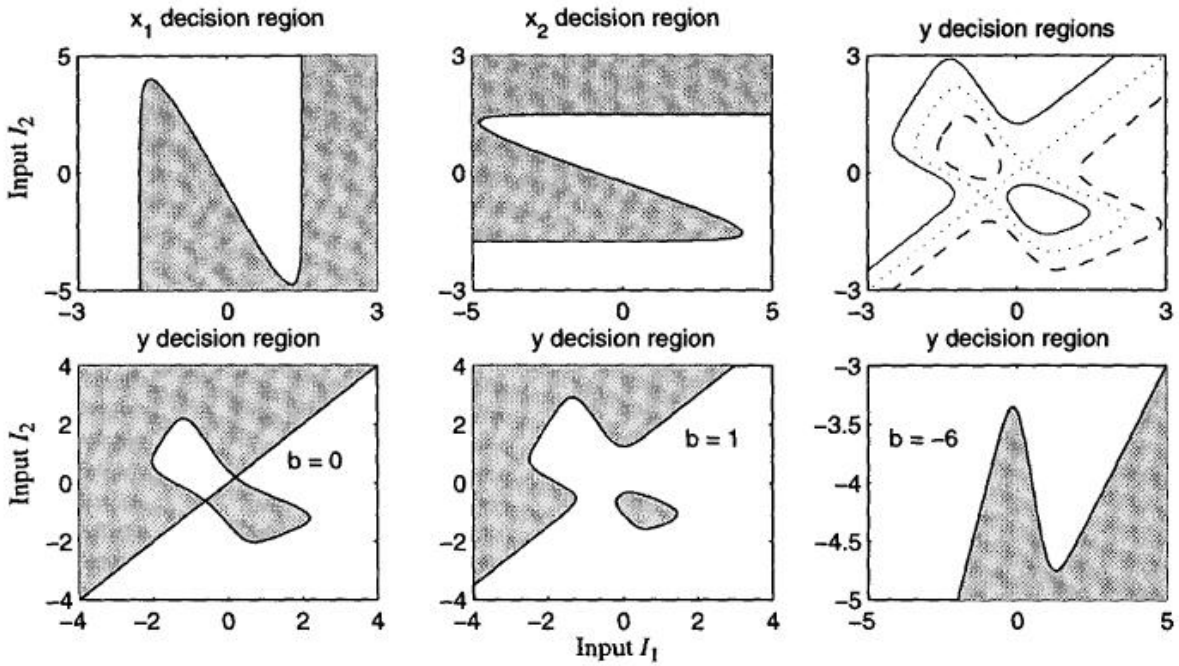

Figure 36 Decision regions of SIANN. a1=a2=1, b1=b2=5, w1=-2, w2=2, c12=c21=1, c11=c22=5 [4].

The shunting layer does a nonlinear transformation on the inputs form a pattern such that it classifies easily with a discriminant function [4].

4.1.2.3 Training SIANN

To produce desired outputs for given inputs, a neural network has to be trained to find appropriate connection weights. The backpropagation is a widely used algorithm for training MLPs in which weights are updated that minimizes the mean-square error along with the use of differentiable activation function. This method can be adapted to train SIANNs. For an SIANN with a single output neuron, the squared error can be defined as [5],

$$E = \frac{1}{2}e^2 = \frac{1}{2}\left(y - d\right)^2 \tag{4.1.8}$$

where, d is the desired output, y is the actual output and e is the error.

The gradient of the squared error E with respect to the parameter c_{ij} is [5]

$$\frac{\partial E}{\partial c_{ij}} = \frac{\partial E}{\partial x_i}\frac{\partial x_i}{\partial c_{ij}} = ¢_i \frac{-x_i I_j f'\left(\sum_j c_{ij} I_j\right)}{a_i + f\left(\sum_j c_{ij} I_j\right)} \tag{4.1.9}$$

with $¢_i = \frac{\partial E}{\partial x_i} = \frac{\partial E}{\partial y}\frac{\partial y}{\partial x_i} = w_i g' e$

where $¢_i$ is the backpropagated error signal and the prime represents differentiation with respect to the argument. Similarly, partial derivatives of E with respect to a_i, b_i, w_i and b can be derived. In the gradient descent algorithm, a network parameter « can be updated as in [5],

$$«_{new} = «_{old} - ¦ \frac{\partial E}{\partial «} \tag{4.1.10}$$

where ¦ is the learning rate and « is one of the network parameters [4].

4.1.2.4 SIANN versus MLP

Figure 37 displays the performances of SIANN and MLP where the function to be approximated is $h(t) = 0.1 + 1.2t + 2.8t\sin(4®t^2)$, $t \in [0,1]$. It can be seen that the SIANN approximates the function more accurately than the MLP.

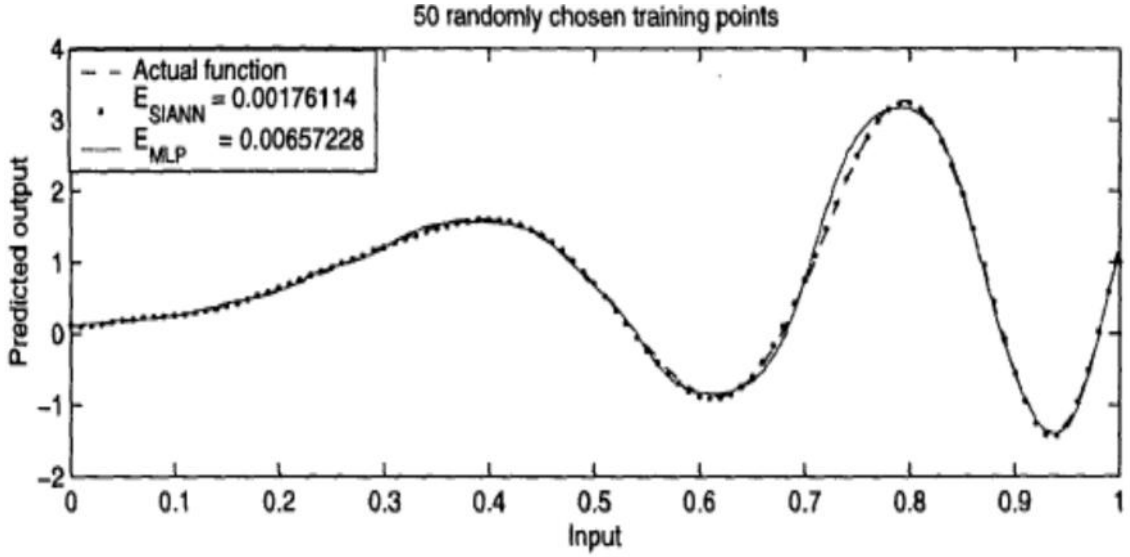

Figure 37 Estimated function using SIANN and MLP [4]

Figure 38 illustrates the decision boundaries of two SIANNs with fewer hidden units (2 and 4) and fewer parameters (11 and 21), and two MLPs with 5 hidden units and 21 parameters to classify the points inside and outside the unit circle indicated by dashes. Here also, the SIANNs do better classification than the MLPs [4].

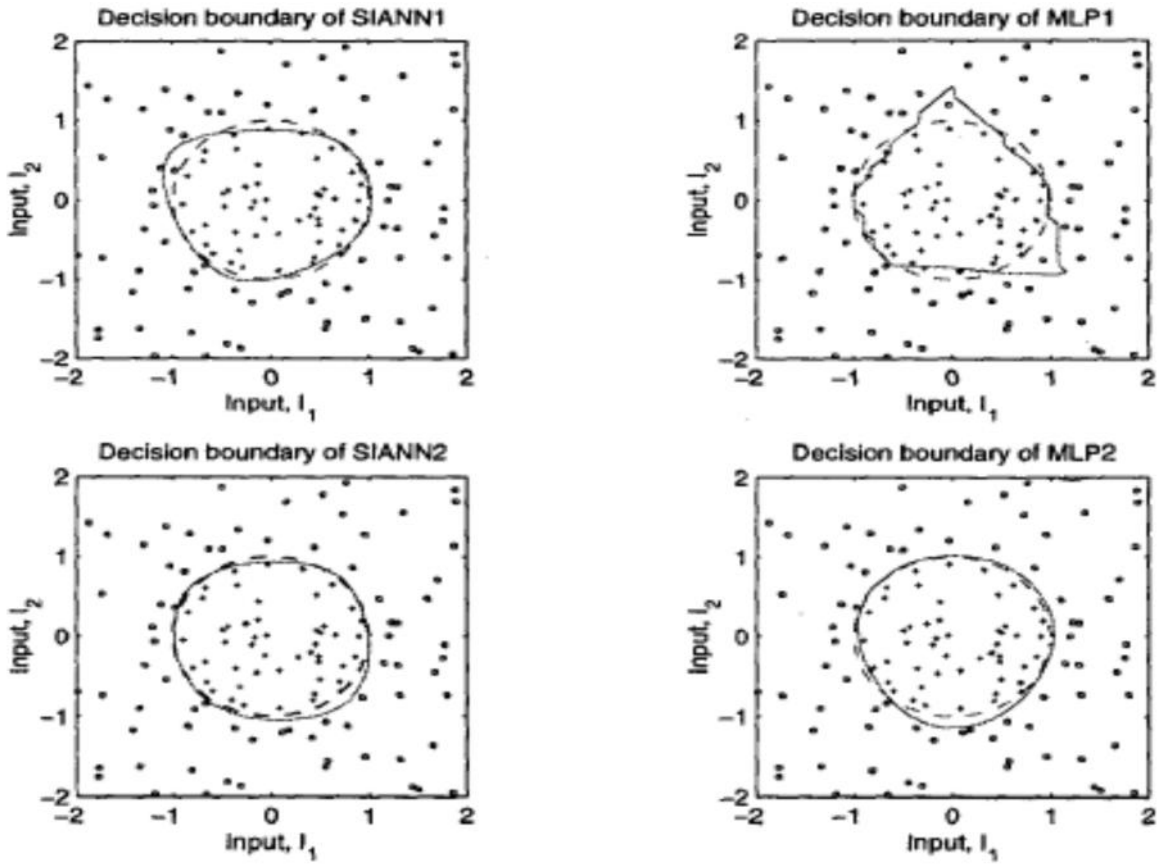

Figure 38 Classification using SIANN and MLP [4]

4.1.3 Neuro-Inspired Robot Cognitive Control with Reinforcement Learning

This section discusses about work done by Mehdi Khamassi, Stéphane Lallée, Pierre Enel, Emmanuel Procyk, and Peter F. Dominey [1]. They utilized neuro-physiologically motivated brain-like model to design cognitive system for learning and control to deal with uncertainties arising in the environment. They deal with two types of uncertainties: (1) a familiar, expected uncertainty that results from noise due to sensor-motor interaction and, (2) unexpected

uncertainty that results from varying stochastic environment. The model could deal with these uncertainties by adaptively regulating a meta-parameter called .

4.1.3.1 Experiment Setup

A humanoid agent–a physical robot or a simulation is considered in the experiments done in [1]. The interaction with the environment is done through visual perception and motor commands. All three experiments are similar and use vision systems to provide all the inputs. An agent has to search by trial and error (exploration) to find the best possible reward. Then the agent has to repetitively select the best award which is exploitation. Indication of new exploration task is provided by a problem-changing cue (PCC). The agent has to learn to decide probability and the PCC in the experiments.

4.1.3.2 The Neural Network Model

In the work done by Khamassi and the others [1], the neural network model whose architecture is inspired by anatomical connections in the brain of the monkeys is shown in figure 39. The locations in the visual space are encoded by a 3 * 3 array of leaky integrator neurons. A neuron's membrane potential mp is given by [1]:

$$\tau \frac{\partial mp}{\partial t} = -mp + s \qquad (4.1.11)$$

where τ is a time constant and s is input. Then average firing rate output based on a nonlinear function is generated. The posterior (PPC) receives the visual input. The ACC estimates action values associated with each possible selections based on temporal difference algorithm. This action values are sent to dopamine neurons in the ventral tegmental area (VTA) where a reward prediction error is computed after reception of the reward [1]:

$$¢ = r - Q(\mathrm{a}_i) \qquad (4.1.12)$$

where a_i, $i \in [1,\ldots, 4]$ is the action value and r is the reward value. The reinforcement signal ¢ is sent to ACC to update the weights corresponding to the action value neuron [1]:

$$Q(a_i) \leftarrow Q(a_i) + \sim \cdot ¢ \cdot trace(a_i) \quad (4.1.13)$$

where trace is the efferent copy of the chosen action sent by the premotor cortex (PMC) to reinforce the ACC and ~ is a learning rate.

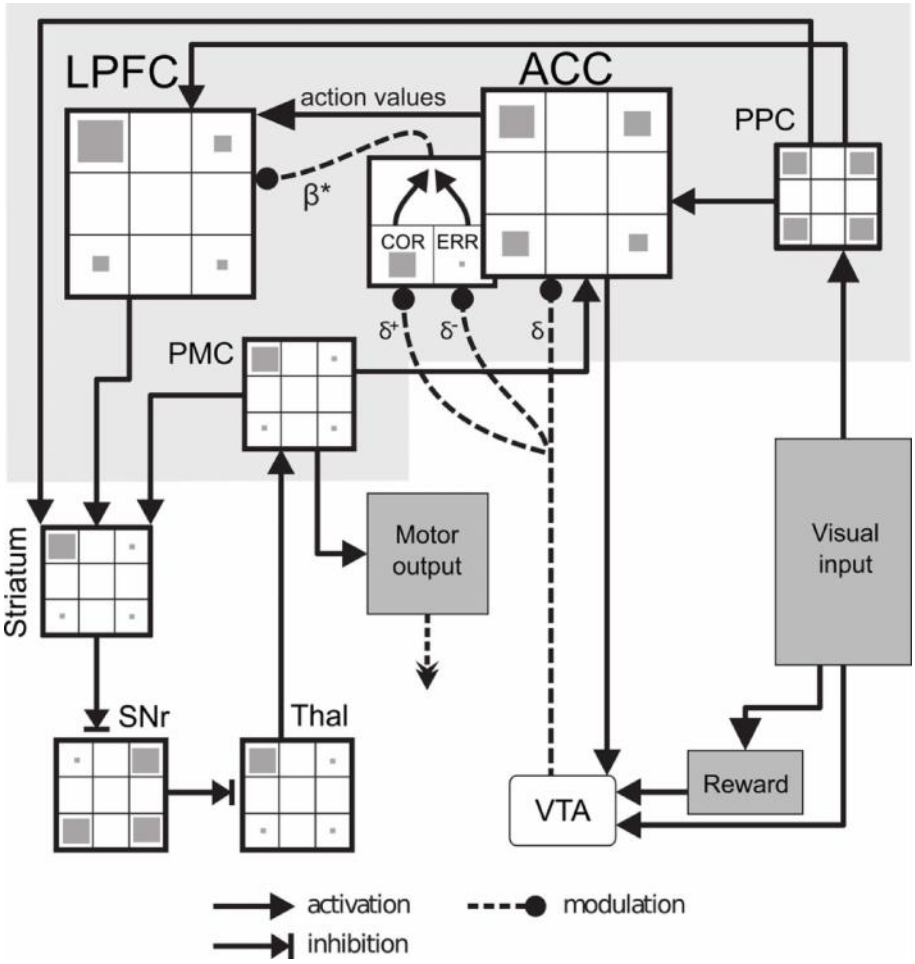

Figure 39 The neural model [1]

Now, an outcome history (COR-correct neuron; ERR- error neuron) is computed modulate the exploration level *ß in ACC [1]:

$$COR(t) = ¢(t), if\ ¢(t) \geq 0$$
$$ERR(t) = -¢(t), if\ ¢(t) < 0$$
$$^{*}(t) \leftarrow \ ^{*}(t) + \sim_{+} \cdot COR(t) + \sim_{-} \cdot ERR(\mathrm{t}) \quad (4.1.14)$$

where $\sim_{+} = -2.5$ and $\sim_{-} = 0.25$ are updating rates with * (0< *<1). The ACC sends action values * to DLPFC. The DLPFC regulates the exploration rate $(0 < < 10)$ by a sigmoid function with sign reversal [1]:

$$= \frac{\mu_1}{(1 + \exp(\mu_2 \cdot [1 - ^{*}] + \mu_3))} \tag{4.1.15}$$

where μ_1=10, $\mu_2 = -6$ and $\mu_3 = 1$. The DLPFC decides upon the action to be taken based on a Boltzman softmax function [1]:

$$P(a_i) = \frac{\exp(\cdot Q(a_i))}{\sum_j \exp(\cdot Q(a_j))} \tag{4.1.16}$$

where 0< regulates the exploration rate. A small leads to exploration as each action has almost equal probabilities where as a high leads to exploitation. The selected action to be executed is gated through the cortico-basal ganglia loop consisting of striatum, substantia nigra reticulata (SNr), and thalamus (Thal) until the premotor cortex (PMC). The PMC is used to command the agent.

The ACC learns association of average reward with different objects using meta-values and they are updated with dependence on the variations in the average reward at the end of each trial [1]:

$$M(o_i, t) \leftarrow M(o_i, t) + \cdot \bar{}(t) \tag{4.1.17}$$

where is learning rate and $\bar{}(t)$ is the estimated reward average. If the meta-value of any object is lower than a threshold, then action values are reset to some random values and * is increased. This means a low and the agent will start exploration instead of doing exploitation.

References

1. Mehdi Khamassi, Stéphane Lallée, Pierre Enel, Emmanuel Procyk, Peter F. Dominey, “Robot cognitive control with a neuro-physiologically inspired reinforcement learning model”, Frontiers in Neurorobotics, volume 5, article 1, 2011, pp. 1-14.

2. Christian Balkenius, Stefan Winberg, “Cognitive Modeling with Context Sensitive Reinforcement Learning”, Proceedings of AILS 04, 2004, pp. 10-19.

3. Petru E. Stingu, Frank L. Lewis, “Neuro Fuzzy Control of Autonomous Robotics”, Springer (Book), 2009.

4. Ganesh Arulampalam, Abdesselam Bouzerdoum, “A generalized feedforward neural network architecture for classification and regression”, Neural Networks 16, 2003, pp. 561–568.

5. Abdesselam Bouzerdoum, “Classification and function approximation using feed-forward shunting inhibitory artificial neural networks”, IJCNN, 2000, pp. 613-618.

6. Joshua Brown, Daniel Bullock, Stephen Grossberg, “How the basal ganglia use parallel excitatory and inhibitory learning pathways to selectively respond to unexpected rewarding cues”, The Journal of Neuroscience, 1999, pp. 10502–10511.

4.2 Multiple Model Based Learning and Control

4.2.1 Introduction

Previous section discussed various controllers inspired from the study of the decision process done by the human brain. They all used single controller/predictor to achieve a particular goal. But, further study of brain suggests that are multiple control structures working simultaneously in the brain. This has led to use of multiple controllers and/or predictors used. Here, some of them are discussed which use multiple reinforcement learning structures, multiple neural networks and multiple adaptive controllers.

4.2.2 Multi-Model Adaptive Control

This section discusses the work done by Narendra and Balakrishnan in [5]. They believed that an intelligent controller needs to have the ability to adapt rapidly in any unknown, rapidly changing environment. To serve this purpose, they proposed different switching and tuning schemes for multiple model adaptive control. It combines fixed and adaptive models in various ways. They have studied this multiple model architecture with all fixed models, all adaptive models and one adaptive-rest fixed models. These schemes are proved to be stable. When the environment of a system changes abruptly, a new model of the environment along with the appropriate controller has to be chosen other than the current one. The controllers can be pre-designed if the models are already available for different environments. As there are only finite number of models with any possible environment, switching and tuning are very important. The switching is to select the model matching with some defined criteria rapidly. The tuning is the adjustment of the parameters of the chosen model tom improve accuracy. It is difficult to decide the moment of switch, which model and the rule to be used for tuning.

4.2.2.2 Design of the Control System

The proposed architecture by Narendra and Balakrishnan [5] for intelligent control is shown in figure 40. This is a general one and can be applied to both linear and nonlinear

systems. The system to be controlled has input u and output y. The aim is to make the control error $e_c = y^* - y$ go to zero, where y^* is the desired output. There are N identification models which are denoted by $\{I_j\}_{j=1}^{N}$ operate in parallel. $\hat{p}_j$ is the parameter vector of each I_j. The identification error between the output y_j of I_j and that of the plant is denoted as $e_j = \hat{y}_j - y$. There is a controller C_j corresponding to each I_j, which has parameter vector $\bar{}_j$. The output of C_j is denoted by u_j. One I_j is selected by a switching rule and the corresponding control input u_j controls the plant. The design problem here is to select the number of models and controllers along with their parameters. The control problem is to decide the rules for switching and tuning that can give a stable and the best performance.

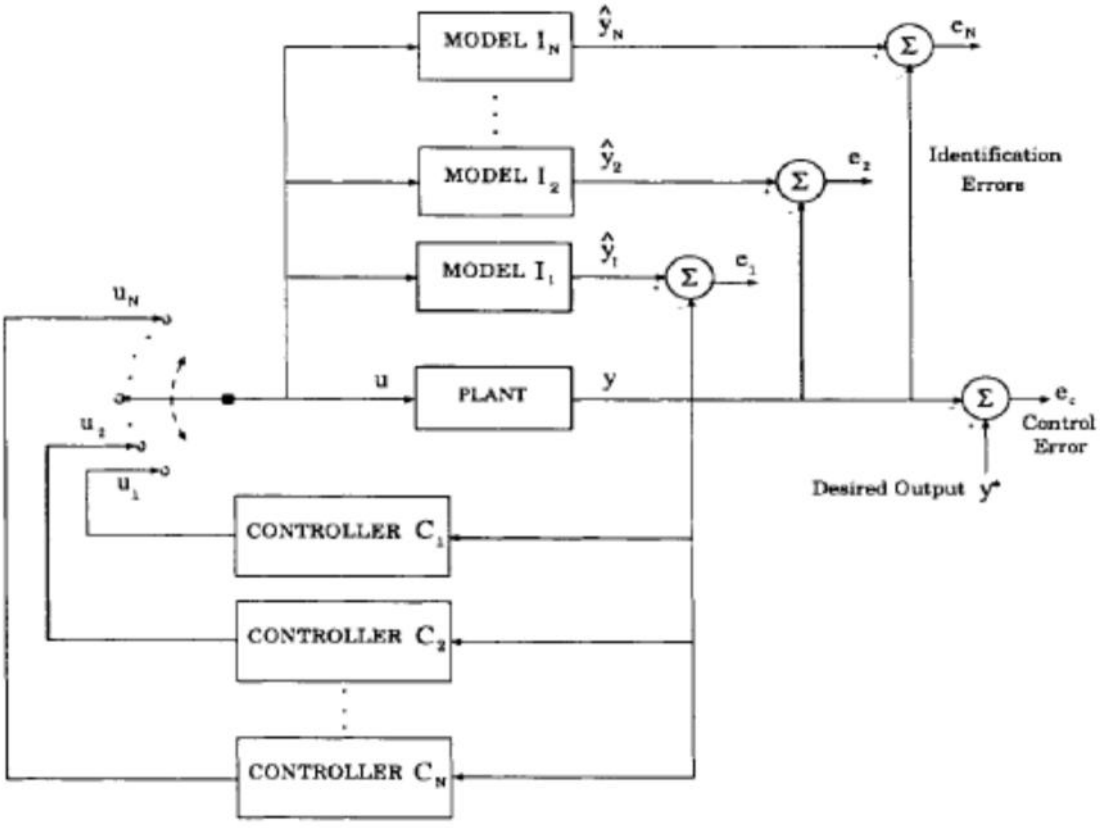

Figure 40 A multi-model architecture [5]

4.2.2.3 Switching Rule Selection

Narendra and Balakrishnan have presented methods to select switching rule in [5]. One way is to compute performance cost indexes $J_i(t)$ corresponding to every controller C_j at every moment and then switching to the one with the minimum index; but this requires to apply all the control inputs which is not possible. So, a feasible option of simultaneous computation of performance indexes of all the identification models is used, which means that identification errors $\{e_j\}$ are used. The performance index can be computed as [5],

$$J_i(t) = \sim e_j^2(t) + \int_0^t e^{-\mathrm{a}(t-²)} e_j^2(²)d² \,, \sim \geq 0, \quad , \mathrm{a} > 0 \qquad (4.2.1)$$

where ~ and are utilized for desired combination of instantaneous and long-term accuracy measures, a is the forgetting factor from which the memory of the index is decided and it is an assurance of boundedness of $J_i(\mathrm{t})$ for bounded e_j. A minimum wait period is allowed to elapse after every switch in order to prevent arbitrarily fast switching.

4.2.3 Multiple Model-Based Reinforcement Learning

Kenji Doya and the others [1, 2] proposed a multiple model based reinforcement learning (MMRL) architecture for nonlinear, non-stationary control tasks. It has a modular architecture of multiple models. Basically, a complex task is divided into multiple domains in space and time. It can learn all possible outcomes resulting from a same cue stimulus. Additionally, this multiple model architecture has provided a valid explanation of the behavior of dopamine neurons in reward prediction. An implementation of the MMRL is shown in the figure 41. Every module has a reward predictor and a value estimator. After the cue stimulus, the model predicts the presence or absence of reward at each time step. A vector of the predicted amount of reward is given by each reward predictor. A responsibility signal λ_i corresponding to the each module is calculated based on the prediction errors of the reward predictors. The reward predictors and value estimators are

updated by gating the responsibility signal. Also, it is used to weight the output of the value estimators. This architecture can be used for control as extended in [1] and it is called modular selection and identification for control (MOSAIC).

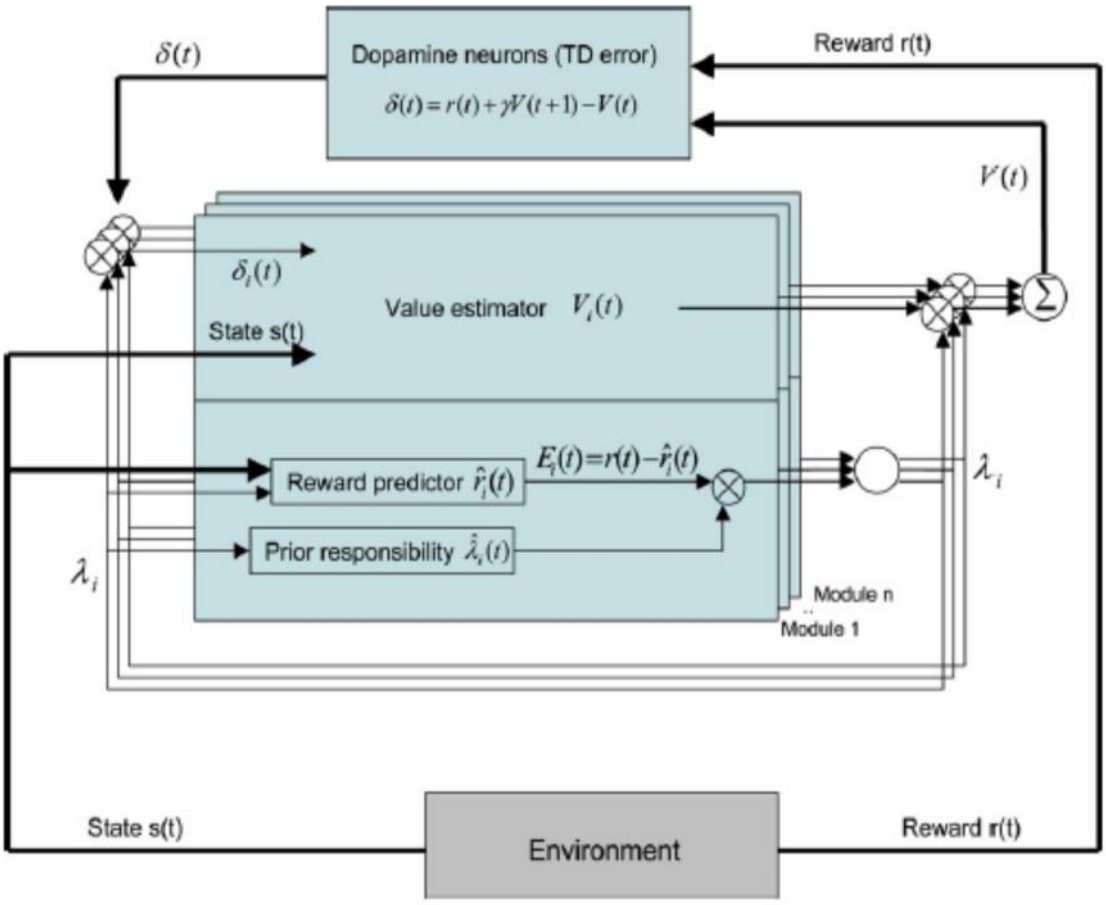

Figure 41 The MMRL architecture [2]

Let us understand how this model works: Assuming there are N > 2 modules, an MMRL has previously trained two reward predictors 1 and 2, predicting a Stimulus-Reward Interval (SRI) of t1 and t2 respectively, with t1 < t2. These to modules predict an amount of reward depending on the previous training at t1 and t2. If the current time t elapsed since the stimulus occurred is small (t < t1 < t2), they both have high responsibility with likely prediction. If no reward occurs at t=t1, then the first predictor makes a large prediction error and its responsibility is downgraded. So, while t1 < t < t2, the overall prediction will mostly depend on second predictor. If at t = t2 no

reward occurs, the predictor 2 will also have a large error and then low responsibility. If a reward occurs afterwards, the most likely of the rest of the predictors will have the highest responsibility [2].

4.2.3.1 Reward Predictor

The reward predictor $\hat{r}_i(t)$ of each module i gives a vector of the predicted reward. The responsibility signal for each predictor is computed as [2]:

$$\hat{a}_i(t) = \frac{\hat{a}_i(t) e^{-\frac{E_i(t)^2}{2\sigma^2}}}{\sum_j \hat{a}_j(t) e^{-\frac{E_j(t)^2}{2\sigma^2}}} \tag{4.2.2}$$

where σ is a constant and $E_i(t) = r(t) - \hat{r}_i(t)$ the prediction error of each reward predictor, $r(t)$ is the actual amount of reward at time step t and $\hat{a}_i(t)$ is the estimated responsibility value encoding some prior knowledge. If the temporal continuity of module selection is considered as the prior knowledge, then using the previous responsibility [2]:

$$\hat{a}_i(t) = a_i(t-1)\tilde{} \tag{4.2.3}$$

where $0 < \tilde{} < 1$ a parameter that controls the strength of the memory effect. Each reward predictor $\hat{r}_i(t)$ is initialized to random values and is updated according to the following equation [2]:

$$\hat{r}_i(t) = \hat{r}_i(t) + \alpha a_i E_i(t) \tag{4.2.4}$$

with $0 < \alpha < 1$ an update rate.

4.2.3.2 Value Estimator

The value estimator of each module updates values similarly to a tap delay line model [2]:

$$V_i(t) = s(t) * \left(w_i(t)\right)' \tag{4.2.5}$$

where w_i is a weight row vector. The global predicted value signal $V(t)$ is a weighted sum of the modules' values [2]:

$$V(t) = \sum_i \alpha_i V_i(t) \tag{4.2.6}$$

The Temporal Difference (TD) error $\delta(t)$ is [2]:

$$\delta(t) = r(t) + \gamma V(t+1) - V(t) \tag{4.2.7}$$

where $0 < \gamma < 1$ is discounting parameter. The TD error for each module is gated by [2]:

$$\delta_i(t) = \alpha_i \delta(t) \tag{4.2.8}$$

The weight vector of each value estimator is updated by [2]:

$$w_i(t) = w_i(t) + \eta s(t) \delta_i(t) \tag{4.2.9}$$

with learning rate η.

4.2.4 Extended Modular Selection and Identification for Control

4.2.4.1 Introduction

Norikazu Sugimoto, Jun Morimoto, Sang-Ho Hyon, and Mitsuo Kawato in [6] present an extension of the MOSAIC architecture for humanoid robot control. A MOSAIC architecture is proposed by Doya and the others in [1]. Also, another is proposed consisting of multiple linear state predictors and controllers. The MOSAIC architecture is flexible enough to learn and control the nonlinear and non-stationary environment. Still, it has the limitations of susceptibility to observation noise and requirement of fully observable system. The eMOSAIC architecture includes state estimators to cope with these two problems in a real environment. This inclusion of the state estimators can better explain the sensorimotor function of the central nervous system. The authors of this work successfully generated squatting and object-carrying with a real humanoid robot.

4.2.4.2 The eMOSAIC Model

Figure 42 shows the eMOSAIC model proposed by Norikazu Sugimoto, Jun Morimoto, Sang-Ho Hyon, and Mitsuo Kawato in [6]. Each module consists of a state estimator, a responsibility predictor, a value function estimator, and a controller. They used switching of linear models and quadratic models to estimate nonlinear states and nonlinear cost function, respectively. The dynamics are [6]:

$$\mathbf{x}(t+1) = A_i\mathbf{x}(t) + B_i\mathbf{u}(t) + c_i + \mathbf{n}(t), \tag{4.2.10}$$

$$\mathbf{y}(t) = H_i\mathbf{x}(t) + \mathbf{v}(t) \tag{4.2.11}$$

$$r_i(\mathbf{x}(t),\mathbf{u}(t)) = \frac{1}{2}\mathbf{x}(t)^T Q_i\mathbf{x}(t) + \frac{1}{2}\mathbf{u}(t)^T R_i\mathbf{u}(t) \tag{4.2.12}$$

where $A_i \in R^{N\times N}$ and $B_i \in R^{N\times D}$ are regression parameters of the ith linear dynamics, $c_i \in R^N$ is bias parameter, and $H_i \in R^{L\times N}$ is an observation matrix. $\mathbf{x} \in R^N, \mathbf{u} \in \mathrm{R}^D$ and $\mathbf{y} \in R^L$ are state, action and observation vectors, respectively, and $\mathbf{n}(t) \sim N(0, \sum_{\mathbf{x}})$ and $\mathbf{v}(t) \sim N(0, \sum_y)$ are system and observation noises. $N(0, \sum)$ denotes a Gaussian distribution with zero mean and covariance $\sum$. $Q_i \in R^{N\times N}$ and $R_i \in R^{D\times D}$ are parameters of the ith quadratic cost function r_i . Optimal controller is found by minimizing the objective function [6]:

$$J = E\left[\sum_{s=0}^{\infty} r(\mathbf{x}(s),\mathbf{u}(s))\right] \tag{4.2.13}$$

And the objective function is minimized by estimating the value function [6]:

$$V(\mathbf{x}(t)) = E\left[\sum_{s=t}^{\infty} r(\mathbf{x}(s),\mathbf{u}(s))\right] \tag{4.2.14}$$

Its working is similar to that of MMRL discussed in the previous section. Each modular function is discussed in the next sub-sections.

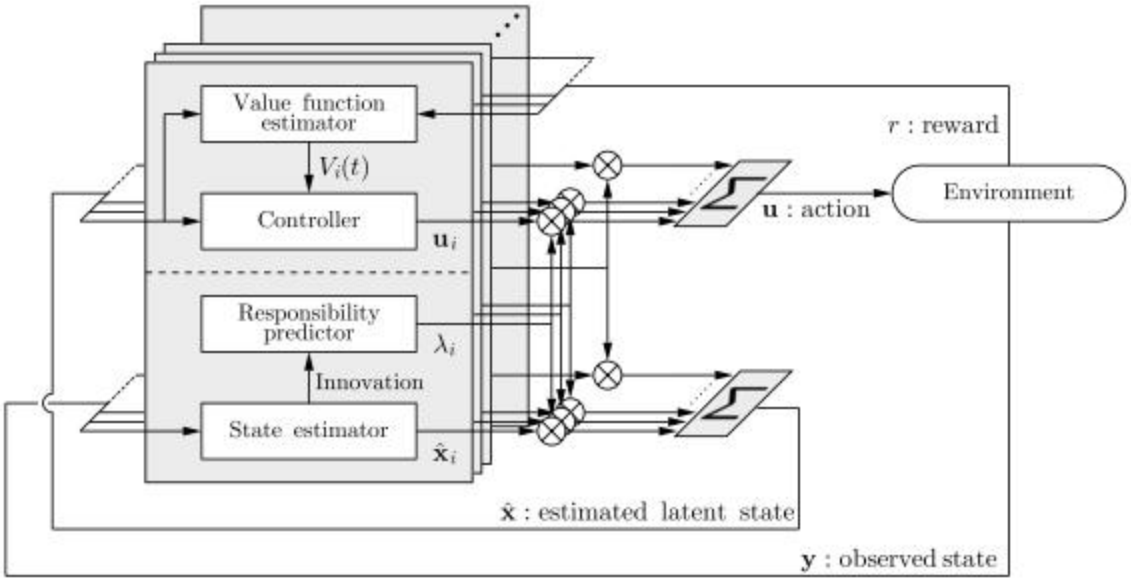

Figure 42 The eMOSAIC model [6]

4.2.4.3 Learning Forward Models

The objective is to minimize the weighted prediction error. If each model is represented in a linear form $\mathbf{x}_i(t+1) = W_i \mathbf{z}(t),$ the expected values for the regression parameters W_i can be derived as [6]:

$$W_i = \frac{\langle \mathbf{xz} \rangle_i (T)}{\langle \mathbf{zz}^T \rangle_i (T)} \tag{4.2.15}$$

The notation $\langle g \rangle_i (T)$ annotated a weighted mean of a function g with respect to the responsibility signal $\mathsf{a}_i(t)$:

$$\langle g \rangle_i (\mathrm{T}) = \frac{1}{T} \sum_{t=1}^{T} g(t) \mathsf{a}_i(t) \tag{4.2.16}$$

The responsibility signal is a probability distribution of the module selection. The likelihood of the learning of the forward model can be made suboptimal by iterating the responsibility signal calculation and parameter update.

4.2.4.4 State Estimators

A linear state estimator estimates the latent dynamics [6]:

$$\hat{\mathbf{x}}_i(t+1|t) = A_i\hat{\mathbf{x}}_i(t) + \mathrm{B}_i\,\mathbf{u}(t) + c_i \tag{4.2.17}$$

$$\hat{\mathbf{x}}_i(t+1) = \hat{\mathbf{x}}_i(t+1|t) + \mathrm{K}_i(\mathbf{y}(t) - H_i\hat{\mathbf{x}}_i(t+1|t)) \tag{4.2.18}$$

where $\hat{\mathbf{x}}_i$ is the estimated state and K_i is the parameter of the state estimator which can be found by solving the linear optimal estimation problem.

4.2.4.5 Responsibility Predictors

The probability distribution a_i weightage of each module and it is given by the Bayes' rule [6]:

$$\mathrm{a}_i(t) = \frac{P(i)\,p(\mathbf{x}(t)\,|\,\mathbf{y}(1:\mathrm{t}),\mathrm{i})}{\sum_{i'\in M} P(i')\,p(\mathbf{x}(t)\,|\,\mathbf{y}(1:\mathrm{t}),\mathrm{i}')} \tag{4.2.19}$$

where M is the set of module indices, $p(\mathbf{x}(t)\,|\,\mathbf{y}(1:\mathrm{t}),\mathrm{i})$ is the likelihood of the ith module, and $P(i)$ is the error. It is assumed that the prediction error and estimation error are Gaussian with covariances $\sum_{\mathbf{x}}$ and $\sum_{\mathbf{y}}$. So, the likelihood of the ith module $p(\mathbf{x}(t)\,|\,\mathbf{y}(1:\mathrm{t}),\mathrm{i})$ is given by [6]

$$p(\mathbf{x}(t)\,|\,\mathbf{y}(\mathrm{t}),\mathrm{i}) \propto p(\mathbf{y}(t)\,|\,\mathbf{x}(\mathrm{t}\,|\,\mathrm{t}-1),\mathrm{i})\,p(\mathbf{x}(t)\,|\,\mathbf{x}(\mathrm{t}\,|\,\mathrm{t}-1),\mathrm{i}) \tag{4.2.20}$$

$$p(\mathbf{y}(t)\,|\,\mathbf{x}(t\,|\,t-1),i) = \frac{1}{\sqrt{(2®)^L\,|\sum_y|}}\exp\left[-\frac{1}{2}\mathbf{e}_i(t)^T\sum_{\mathbf{y}}^{-1}\mathbf{e}_i(t)\right] \tag{4.2.21}$$

$$\mathbf{e}_i(t) = \mathbf{y}(t) - H_i\hat{\mathbf{x}}_i(t\,|\,t-1) \tag{4.2.22}$$

$$p(\mathbf{x}(t\,|\,t-1),\mathrm{i}) = \frac{1}{\sqrt{(2®)^N\,|\sum_{\mathbf{x}}|}}\exp\left[-\frac{1}{2}\mathbf{d}_i(t)^T\sum_{\mathbf{x}}^{-1}\mathbf{d}_i(t)\right] \tag{4.2.23}$$

$$\mathbf{d}_i(t) = \mathbf{x}_i(t) - \left\{A_i\mathbf{x}_i(t-1) + B_i\mathbf{u}(t-1) + c_i\right\} \tag{4.2.24}$$

where $\hat{\mathbf{x}}_i(t\,|\,t-1)$ is the predicted state of the state estimator of the ith module and $\mathbf{e}_i(t) = \mathbf{y}(t) - H_i\hat{\mathbf{x}}_i(t\,|\,t-1)$ is the so called error of innovation.

Finally, the estimated state is computed by the weighted sum of each module [6]:

$$\hat{\mathbf{x}}(t) = \sum_{i \in M} \mathsf{a}_i(t)\hat{\mathbf{x}}_i(t) \tag{4.2.25}$$

where $\hat{\mathbf{x}}_i(t)$ is the estimated state of the ith module at time t.

4.2.4.6 Value Function Estimators

The value function is locally estimated by using a quadratic function [6]:

$$V_i(\mathbf{x}(t)) = \frac{1}{2}(\hat{\mathbf{x}}(t) - \mathbf{x}_i^v)^T P_i(\hat{\mathbf{x}}(t) - \mathbf{x}_i^v) \tag{4.2.26}$$

where the matrix P_i is given by solving Riccati equation [6]:

$$0 = P_i A_i + A_i^T P_i - P_i B_i R_i^{-1} B_i^T P_i + Q_i \tag{4.2.27}$$

The center term $\mathbf{x}_i^v$ of the ith value function is given by [6],

$$\mathbf{x}_i^v = -(Q_i + P_i A_i)^{-1} P_i c_i \tag{4.2.28}$$

4.2.4.7 Controllers

A linear-quadratic optimal control problem is solved locally to derive the controller. With the estimated value function in Eq. (4.2.26), linear optimal controller for the ith module can be derived as [6],

$$\mathbf{u}_i(t) = -R_i B_i^T P_i(\hat{\mathbf{x}}(t) - \mathbf{x}_i^u) \tag{4.2.29}$$

The final output of the controller is computed by the weighted sum of each module output [6]:

$$\mathbf{u}(t) = \sum_{i \in M} \mathsf{a}_i(t)\mathbf{u}_i(t) \tag{4.2.30}$$

References

1. K. Doya, Kazuyuki Samejima, Ken-ichi Katagiri, Mitsuo Kawato, “Multiple Model-Based Reinforcement Learning”, Neural Computation, 2002, pp. 1347-1369.

2. Mathieu Bertin, Nicolas Schweighofer, K. Doya, “Multiple model-based reinforcement learning explains neuronal dopamine activity”, Neural Networks 20, 2007, pp. 668–675.

3. Okihide Hikosaka, Hiroyuki Nakahara, Miya K. Rand, Katsuyuki Sakai, Xiaofeng Lu, Kae Nakamura, Shigehiro Miyachi, Kenji Doya, “Parallel neural networks for learning sequential procedures”, TINS Vol. 22, No. 10, 1999, pp. 464-471.

4. Tetsuya Minatohara, Tetsuo Furukawa, “The Self-Organizing Adaptive Controller”, International Journal of Innovative Computing, Information and Control, Vol 7, number 4, April, 2011, pp. 1933-1947.

5. Kumpati Narendra, Jeyendran Balakrishnan, “Adaptive control using multiple models”, IEEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. 42, NO. 2, 1997, pp. 171-187.

6. Norikazu Sugimoto, Jun Morimoto, Sang-Ho Hyon, Mitsuo Kawato, “The eMOSAIC model for humanoid robot control”, Neural Networks, 2012, pp. 8-19.

Chapter 5

Satisficing

5.1 Satisficing and Control

This section introduces a new concept in control theory named ‘satisficing’. It is different from optimality and even sub-optimality. First, satisficing is explained with respect to various faculties. Then the focus is put on using satisficing for control theory which involves the work done by Goodrich [2] and Curtis [3].

5.1.1 What is Satisficing?

Satisficing is a decision making strategy in which the first option is selected that meets an acceptability threshold out of the available alternatives. This differs from an optimal decision in which the best option is chosen. The term satisficing can be thought of as combining the two words satisfy and suffice. The concept was first presented in Administrative behavior in 1947. Herbert A. Simon introduced satisficing to explain non-optimal decisions made by humans due to the lack of cognitive resources like limited memory and less knowledge of probabilities of outcomes [1]. This is related to an approach of bounded rationality which is discussed in the next chapter.

Let us understand satisficing through an example: The objective to sew a patch onto a pair of jeans for which a 4 inch long needle with a 3 millimeter eye, is the best-optimal option. Now, this needle is put in a pile of 1000 other needles varying in size from 1 inch to 6 inches. Searching for the best needle from this haystack is very effortful and time-consuming. Instead, using the first needle that can sew on the patch is a satisficing solution. Later on, satisficing may led to optimization. Satisficing can help to make a an effective and an efficient decision like in the choice of an outfit; but it may led to wrong decision as in case of medical issues. A person can be a maximizer (optimizer), a satisficer or something in between [1].

A satisficing problem can be defined as an optimization problem using the indicator function of the satisficing requirements as an objective function. With X denoting the set of all options and $S \subseteq X$ denoting the set of "satisficing" options, a satisficing solution from S can be found by solving the optimization problem [1]:

$$\max_{s \in X} I_S(s)$$

where I_S denotes the indicator function of S, that is

$$I_S(s) := \begin{cases} 1, s \in S \\ 0, s \notin S \end{cases}, s \in X$$

A solution $s \in X$ to this optimization problem is optimal if and only if it is a satisficing option. Thus, the difference between "optimizing" and "satisficing" is essentially a stylistic issue rather than a substantive issue. The important thing is to decide whether and what performance measure should be optimized or satisficed. Jan Odhnoff noted in his paper in 1965 that an optimal result can be an unsatisfactory result in a satisficing model [1].

Simon and the others suggested the idea of aspiration level which is the payoff that the agent aspires to. With aspiration level A and maximum payoff U^*, let s define $A \leq U^*$. Then, A: $s \in S$ if and only if $A \leq U(s)$. Also, the set of all options that yield maximum payoff, $O \subseteq S$ since $A \leq U^*$ [1].

Another way to define satisficing is epsilon-optimization. If the "gap" between the optimum and the aspiration is $\varepsilon = U^* - A$, then the set of satisficing options $S(\varepsilon)$ can be defined as all those options s such that $U(s) \geq U^* - \varepsilon$, that is the actions for which the payoff is within epsilon of the optimum [1].

5.1.2 Satisficing Decisions

Goodrich, Stirling and Frost discuss a way to use satisficing for decision and control in [2]. They extend their theory by introducing ‘strong satisficing’ which supports a systematic

procedure for a control design. It is then tested for some application yielding comparable results with optimal control design. Their work adapts from epistemic utility theory. Their approach involves performance measures and design principles to characterize terminal and transition costs. Two independently developed utilities, accuracy (benefit-like attribute) and liability (cost-like attribute), are compared which are defined over the expected consequences of a decision. Similar to the methodology epistemic utility theory, the goal here is of 'error avoidance'; not truth seeking (optimality).

The epistemic utility function is convex and is defined as [2]:

$$£(G,\mathrm{g}) = \sim P_A(G;\mathrm{g}) + (1-\sim)(1-\mathrm{P}_L(G;\mathrm{g})) \tag{5.1.1}$$

where $\sim \in [0,1]$, $P_A(G;\mathrm{g})$ is a probability measure of accuracy support associated with G when g is the state of nature, and $P_L(G;\mathrm{g})$ characterizes the liability exposure associated with G. With a positive linear transformation the utility function is [2]:

$$£_b(G,\mathrm{g}) = P_A(G;\mathrm{g}) - b\,\mathrm{P}_L(G;\mathrm{g})) \tag{5.1.2}$$

where $b = (1-\sim)/\sim$ is the index of rejectivity. If the state of nature is viewed as a random variable and $P_\Theta(W)$ represents the belief that W contains the actual state of the nature, then the expected value of the epistemic utility function is defined as [2]:

$$\bar{£}_b(G) = \int_\Theta \left[P_A(G;\mathrm{g}) - bP_L(G;\mathrm{g})\right] P_\Theta(d\mathrm{g}) = \bar{P}_A(G) - b\bar{P}_L(G) \tag{5.1.3}$$

where $\bar{P}_A(\mathrm{G})$ and $\bar{P}_L(\mathrm{G})$ are expected density function for accuracy and liability, respectively and have unit mass. Now satisficing is defined in terms of epistemic utility theory. The equivalence class of sets [2]:

$$C_b = \left\{S \in B : S = \arg\max_{G \in B} \bar{£}_b(G)\right\} \tag{5.1.4}$$

which is the family of all measurable sets that maximize expected epistemic utility. If S_b is a member of this equivalence class, then it is known as 'maximally satisficing set' for rejectivity b and if $G \subset S_b$, then G is a satisficing set [2]. In terms of control action u [2]:

$$S_b = \{u : \bar{P}_A(\mathrm{u}) - b\bar{P}_L(\mathrm{u}) \geq 0\} \tag{5.1.5}$$

This approach does not require a unique best decision; all the decisions are included for which the above condition is met. Then any one action can be chosen out of this set with the confidence of achieving a justifiable performance. This satisficing controller is insensitive to time-variance as it uses only temporally local information.

5.1.3 Constructive Nonlinear Control using Satisficing

J. W. Curtis and R. W. Beard [3] used satisficing to constructively parameterize a class of universal formulas. They used a control Lyapunov function (CLF) to design satisficing control. The two functions used for the parameterization are constrained to be locally Lipschitz and satisfy convex constraints. They give two examples to illustrate the approach. A CLF is a positive definite, radially unbounded function whose derivate along the system trajectories is negative definite. The CLF can be easily defined without a need to specify any feedback function for a systems with inputs; unlike traditional Lyapunov functions. Sontag has shown that by using a known CLF and the universal formulas, a nonlinear can be rendered to be asymptotically stable system. Here, the universal formulas are parameterized by using the notion of satisficing decision theory. The basic idea is to define two utility functions, 'selectability' (benefits of choosing an action) and 'rejectability' (cost of choosing an action), that quantify the benefits and costs of an action. The "satisficing" set is defined to be those options for which selectability exceeds rejectability. The selectability" function is linked to a CLF [3].

5.1.3.1 Satisficing Set

Curtis and Beard [3] consider an affine nonlinear system

$$\dot{x} = f(x) + g(x)u \tag{5.1.6}$$

where $x \in \mathbf{R}^n$, $f : \mathbf{R}^n \to \mathbf{R}^n, \mathrm{g} : \mathbf{R}^n \to \mathbf{R}^{n\times m}$ and $u \in \mathbf{R}^m$. f and g are locally Lipschitz functions and $f(0) = 0.$ A twice continuously differentiable function $(C^2)\,\mathrm{V} : \mathbf{R}^n \to \mathbf{R}$ is said to

be a CLF for the system defined by (5.1.6), if V is positive definite, radially bounded, and if $\inf_u V_x^T(f+gu) < 0$ for all $x \neq 0; V_x \triangleq \frac{\partial V}{\partial x}$.

A CLF V is said to satisfy the small control property for (5.1.6) if there exists a control law $\sim_c(x)$ continuous in $\mathbf{R}^n$ such that [3],

$$V_x^T(f + g\sim_c) < 0, \forall x \neq 0 \tag{5.1.7}$$

The satisficing set $S_b(\mathrm{x})$ is defined to be the set of control values such that the selectability times the selectivity index $0 < b(x) < \infty$ is greater than the rejectability [3],

$$S_b(x) = \left\{ u \in \mathbf{R}^m : p_s(u,x) > \frac{1}{b(x)} p_r(u,x) \right\},$$

where $p_s(u,x)$ is the selectability function and $p_r(u,x)$ is the rejectability function. $S_b(\mathrm{x})$ is a convex set (figure 43), if for each x, $p_s(u,x)$ is a concave function of u and $p_r(u,x)$ is a convex function of u.

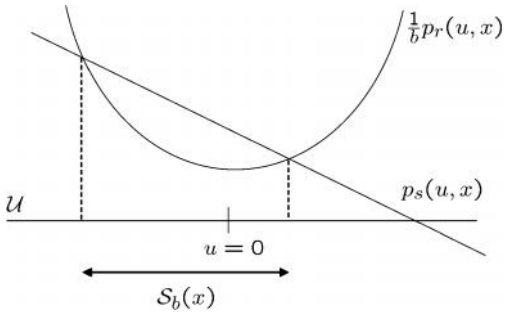

Figure 43 Selectability and rejectability functions and satisficing set for particular x and single input u [3]

If $p_s(u,x) = -V_x^T(f+gu)$ and $p_r(u,x) = l + u^T \mathbf{R} u,$ then the satisficing set at state x is nonempty if and only if $b(x)$ satisfies the following inequality at that state [3]:

$$l(x) + b(x)V_x^T f(x) - \frac{1}{4}b(x)^2 V_x^T g(x)\mathbf{R}^{-1}(x)g^T(x)\,\mathrm{V}_x(x) < 0 \tag{5.1.8}$$

It shows that the selectivity index $b(x)$ plays a critical role in the size of $S_b(x)$. The nonempty $S_b(x) \subset \mathbf{R}^m$ is [3],

$$S_b(x) = \left\{ -\frac{1}{2} b\mathbf{R}^{-1} g^T V_x + \mathbf{R}^{-1/2} \neg \sqrt{\frac{1}{4} b^2 V_x^T g\mathbf{R}^{-1} g^T V_x - l - bV_x^T f} : \|\neg\| < 1 \right\} \quad (5.1.9)$$

For the satisficing set to be nonempty at each $x \neq 0$ [3],

$$\underline{b}(x) \triangleq \left\{ \begin{array}{c} -\frac{1}{V_x^T f}, \text{if} V_x^T g = 0 \\ \frac{2V_x^T f + 2\sqrt{\left(V_x^T f\right)^2 + lV_x^T g\mathbf{R}^{-1} g^T V_x}}{V_x^T g\mathbf{R}^{-1} g^T V_x}, otherwise \end{array} \right\} \quad (5.1.10)$$

From [3], it can be proved that if V is a CLF for system (5.1.6), $\underline{b}$ is given by (5.1.10) and S_b is given by (5.1.9), then for each $x \neq 0$

1. $\underline{b}(x) > 0$;
2. $b > \underline{b}(x)$ implies that $S_b(x) \neq \varnothing$;
3. If $l : \mathbf{R}^n \to \mathbf{R}^+$ satisfies the property

$$(g^T V_x \neq 0 and V_x^T f = 0) \Rightarrow l > 0 \quad (5.1.11)$$

then $\underline{b}(x)$ is locally Lipschitz on $\mathbf{R}^n \setminus \{0\}$. The following are defined as in [3],

$$\pm_1(x,b) \triangleq \frac{1}{2} b\mathbf{R}^{-1} g^T V_x \quad (5.1.12)$$

$$\pm_2(x,b) \triangleq \mathbf{R}^{-1/2} \sqrt{\frac{1}{4} b^2 V_x^T g\mathbf{R}^{-1} g^T V_x - l - bV_x^T f} \quad (5.1.13)$$

Now the union of $S_b(x)$ overall $b \geq \underline{b}(x)$ can be taken for all $x \neq 0$ to obtain

$$S(x) = \left\{ -\pm_1(x,b) + \pm_2(x,b)\neg : b > \underline{b}(x), \|\neg\| < 1 \right\} \quad (5.1.14)$$

Then, $S(x)$ is guaranteed to be nonempty for $x \neq 0$. Thus, the satisficing set is parameterized by the selection functions $b : \mathbf{R}^n \to \mathbf{R}$ and $\neg : \mathbf{R}^n \to \mathbf{R}^m$, where $b(x) \geq \underline{b}(x)$ and $\|\neg(x)\| < 1$.

5.1.3.2 Satisficing Controls

Curtis and Beard [3] defined satisficing controls to be locally Lipschitz selections from the satisficing set. It asymptotically stabilize the closed-loop system.

The mapping $k : \mathbf{R}^n \to \mathbf{R}^m$ is called a satisficing control for the system (5.1.6) if [3],

1. $k(0) = 0$;
2. $k(x) \in S(x)$ for each $x \in \mathbf{R}^n \setminus \{0\}$;
3. k is locally Lipschitz on $\mathbf{R}^n \setminus \{0\}$.

The following parameterizes the set of satisficing controls via two locally Lipschitz selection functions [3]:

If

1. V is a CLF for system (5.1.5);
2. $\neg : \mathbf{R}^n \to \mathbf{R}^m$ is locally Lipschitz on $\mathbf{R}^n \setminus \{0\}$ and satisfies $\|\neg(x)\| < 1$;
3. $\mathrm{b} : \mathbf{R}^n \to \mathbf{R}^+$ is locally Lipschitz on $\mathbf{R}^n \setminus \{0\}$ and satisfies $b(x) > \underline{\mathrm{b}}(\mathrm{x})$;

then

$$k(x) = \begin{cases} 0, & x = 0 \\ -\pm_1(x, b(x)) + \pm_2(x, b(x))\neg(\mathrm{x}), & otherwise \end{cases} \tag{5.1.15}$$

is a satisficing control for system (5.1.6). Moreover, as V satisfies the small control property, $b(x) = ¦(x)\underline{b}(x)$ in a neighborhood close to the origin; where $1 < ¦(x) < N < \infty$, and $\mathbf{R}(x)$ satisfies $\underline{r}I \le \mathbf{R}(x) \le \overline{r}I, \forall x$, where $\underline{r}$ and $\overline{r}$ are positive constants, then k is continuous at the origin. Then, inverse optimality is also proven. This satisficing approach can be used for the design and analysis of nonlinear control problems [3].

References

1. Satisficing from Wikipedia

2. Michael A. Goodrich, Wynn C. Stirling and Richard L. Frost, “A theory of satisficing decision and control,” IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNATICS-PART A: SYSTEMS AND HUMANS, VOL. 28, NO. 6, NOV. 1998, pp. 763-779.

3. J. W. Curtis, R. W. Beard, “Satisficing: A new approach to constructive nonlinear control”, IEEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. 49, NO. 7, JULY 2004, pp. 1090-1102.

4. Samuel J. Gershman, Jonathan D. Cohen, Yael Niv, “Learning to Selectively Attend”, Cognitive Science Society, 2006, pp. 1270-1275.

5.2 Satisficing Games

Section 5.1 explained about satisficing and its use to control a single agent/system. Now this section illustrates its use in game theory. It mainly involves the work done by Stirling.

5.2.1 Satisficing in Game Theory

Stirling and Goodrich introduced the concept of satisficing in game theory [1-3]. In their work, notion of ‘comparative rationality’ is defined for satisficing. Preferences of m-agents are prescribed by a 2m-dimensional ‘interdependence function and the set of jointly satisficing decision is derived. This satisficing decisions are robust and functional which can deal with uncertainty very efficiently. Then the authors analyzed some games using this theory.

Game theory is a theory of decision making in which two or more agents/decision makers are involved. The game generally consists of players, preferences and choices of players, rules, probabilities, outcomes and payoffs. Nash equilibria, dominance and Pareto-optimality are the most highly developed solutions derived by extending the principle of optimality to a multi-agent domain; individual optimality is not always jointly optimal, so a satisficing solution is found. It is common to evaluate options by comparing potential gains with potential losses and choosing the one for which the gain exceeds the losses. This decision is called ‘comparatively rational’. This comparative rational decision is defined by the authors as a satisficing one. The notion of epistemic utility theory is used which was explained in the previous section (5.1.2). Expression of utility theory in the language of probability makes it well equipped to extend from single-agent games to multi-agent games. Requirement of multi-variate credibility (benefit, gain) and rejectability (cost, lost) functions is the main difference. These derivation is performed by interdependence function [3].

Accuracy (benefit) and liability (cost) are assumed to be independent concepts. This means that it is neither a guaranty nor a necessity to simply consume the resources for achievement of the goal. When the joint credibility/rejectability function factors into the product of

the ‘marginal’ credibility and liability function for a single agent, it called the ‘intra-dependence’ of accuracy and liability. It generally holds for single agent decision problems; but not for multi-agent decision problems [3].

Consider a system of m agents $\mathbf{X} = \{X_1, ..., X_m\}$. U_i denotes the decision or option space for agent X_i, F_i denotes the Boolean algebra of option sets available to X_i. A satisficing game is defined as the probability space $\mathbf{U} \times \mathbf{U}, \wp \otimes \wp, \mathrm{P}_{CR}, \mathbf{U} = U_1 \times \cdots \times U_m$ is the multipartite option space, $\wp = F_1 \otimes \ldots \otimes F_m$ is a Boolean algebra of multi-partite option sets, and $P_{CR} : \wp \otimes \wp \mapsto [0,1]$ is a probability measure over $\wp \otimes \wp$, the smallest Boolean algebra that contains all products of measurable rectangles of the form $\mathbf{V} \times \mathbf{W}$, with $\mathbf{V}, \mathbf{W} \in \wp$. This is represented as $P_{CR}(\mathbf{V}, \mathbf{W})$, interdependence measuring function of two set variables, where $\mathbf{V}$ is a measurable subset of $\mathbf{U}$ with respect to which credibility is considered, and $\mathbf{W}$ is a measurable subset of $\mathbf{U}$ with respect to which rejectability is considered. It characterizes the joint credibility/rejectability of all elements of $\wp \otimes \wp$; that is, for $\mathbf{V} \in \wp$ and $\mathbf{W} \in \wp$, $P_{CR}(\mathbf{V}, \mathbf{W})$ expresses the accuracy associated with $\mathbf{V}$ and the liability associated with $\mathbf{W}$.

Let $\mathbf{V} = V_1 \times \cdots \times V_m$ and $\mathbf{W} = W_1 \times \cdots \times W_m$ be rectangles in $\mathbf{U}$. The interdependence measure can be rewritten as $P_{C_1 \cdots C_m R_1 \cdots R_m}(V_1, \ldots, V_m, W_1, \ldots, \mathrm{W}_m)$. Since $\mathbf{U}$ is discrete, the interdependence probability mass function can be defined as [3],

$$P_{C_1 \cdots C_m R_1 \cdots R_m}(V_1, \ldots, V_m, W_1, \ldots, \mathrm{W}_m) = P_{C_1 \cdots C_m R_1 \cdots R_m}(\{v_1\}, \ldots, \{v_m\}, \{w_1\}, \ldots, \{w_m\}) \quad (5.2.1)$$

where $\{v_i\}$ and $\{w_i\}$ are singleton sets in F_i, i = 1,..., m. The interdependence function represents the joint benefit and cost of X_i considering the adoption of v_i from the perspective of accuracy (achieving the goal) and, simultaneously, considering the adoption of w_i from the perspective of liability (being exposed to things undesirable), for i = 1,..., m. This function simply

encodes the information. The function can be factored into a product of conditional probability mass functions which makes the specification the conditional behavior easier. Due to modularity features, it can be used to characterize local or specific responses. For two agents, the interdependence function may be factored as [3],

$$P_{C_1C_2R_1R_2}(v_1, v_2, w_1, \mathrm{w}_2) = P_{C_1R_1|C_2R_2}(v_1, w_1 \mid v_2, w_2) P_{C_2R_2}(v_2, w_2)$$
$$= P_{C_1|C_2R_1R_2}(v_1 \mid v_2, w_1, \mathrm{w}_2) P_{R_1|C_2R_2}(w_1 \mid v_2, \mathrm{w}_2) P_{C_2|R_2}(v_2 \mid \mathrm{w}_2) P_{R_2}(\mathrm{w}_2)$$

where $P_{C_1|C_2R_1R_2}(v_1 \mid v_2, w_1, \mathrm{w}_2)$ represents X_1's conditional credibility given that X_2 places its entire unit of credibility mass on v_2 and all of its rejectability mass on w_2, and X_1 places all of its rejectability mass on v_1. Also, $P_{R_1|C_2R_2}(w_1 \mid v_2, \mathrm{w}_2)$ represents X_1's rejectability of w_1 given that X_2 places all of its credibility mass on v_2 and all of its rejectability mass on w_2. Due to intra-independence for single agents, $P_{C_2|R_2}(v_2 \mid \mathrm{w}_2) = P_{C_2}(v_2)$. The probabilities P_{C_2} and P_{R_2} represent X_2's myopic (isolation from the influence from others) credibility and rejectability, respectively.

. The satisficing set is obtained via Levi's rule, which requires the multi-variate credibility and rejectability probability mass functions, P_C and P_R. These functions can be obtained as follows [3],

$$P_C(\mathbf{v}) = \sum_{\mathbf{w} \in \mathbf{U}} P_{CR}(\mathbf{v}, \mathbf{w}) \tag{5.2.2}$$

$$P_R(\mathbf{w}) = \sum_{\mathbf{v} \in \mathbf{U}} P_{CR}(\mathbf{v}, \mathbf{w}) \tag{5.2.3}$$

and Levi's rule is extended to the multi-agent case by defining the Multipartite Rule of Epistemic Utility [3]:

$$S_b = \{\mathbf{u} \in \mathbf{U} : P_C(\mathbf{u}) \geq b P_R(\mathbf{u})\} \tag{5.2.4}$$

S_b is termed the multipartite satisficing set, and elements of S_b are multipartite satisficing options. This is strengthened by restricting attention to the multi-partite strictly satisficing options

$$S_b^+ = \left\{ \mathbf{u} \in \mathbf{U} : P_C(\mathbf{u}) > bP_R(\mathbf{u}) \right\} \tag{5.2.5}$$

Now, this can be applied to multi-player games that illustrate epistemic utility-based satisficing.

References

1. Matthew Nokleby, Wynn Stirling, “Attitude Adaptation in Satisficing Games”, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, PART B: CYBERNETICS, VOL. 39, NO. 6, DEC. 2009, pp. 1556-1567.

2. Wynn C. Stirling, “Satisficing Games and Decision Making”, Cambridge University Press (Book), 2003.

3. Wynn C. Stirling, Michael A. Goodrich, “Satisficing games”, Information Sciences 1999, pp. 255-280.

4. Surachai Charoensri, H.W. Corley, “The Scalarization and Equivalence of Standard Optimization Criteria”, COSMOS Technical Report 11-03.

Chapter 6

Bounded Rationality

6.1 What is Bounded Rationality?

This section throws light on another concept of human decision making which is called ‘Bounded rationality’. It is based on the limitation of resources and time available for decision making. This way it is related with satisficing discussed in the previous chapter. The first sub-section explains the concept of bounded rationality. The following sections relate it to psychology, economics and management. It includes the work done by Kahneman, Castellaneta, Fogel, Vreeswijk and Park.

6.1.1 Introduction

Decision-making is viewed in economics and related disciplines as a fully rational process that finds the optimal choice. But Herbert Simon proposed the idea of ‘bounded rationality’ in decision making because the optimality of individuals is limited by the availability of information, limited cognitive skills and the constraint of time. Another point of view is that the people simplify the available choices and then only apply their rationality. Thus the decision-maker is one seeking a satisfactory solution rather than the optimal one, a ‘satisficer’. Simon compares bounded rationality with of a pair of scissors where one blade is the "cognitive limitations" of actual humans and the other the "structures of the environment". Thus, pre-existing structures and regularity in the environment can be successfully exploited with the limitations [1].

Some models in the social sciences assume humans as "rational" entities. People are assumed to be on an average as rational and they are approximated to act according to their preferences (selective attention, Reyna-2012). This is in contrast to the concept bounded rationality that finite computational resources hinder the feasibility of making rational choices [1]. Simon suggests that when most people make decisions, they are only partly rational and partly irrational. He also suggests that boundedly rational agents face constraints in formulating and

solving complex problems, and in receiving, storing, retrieving, and transmitting information. He describes a number of dimensions including limited the types of utility functions, the costs and possible multi-valued utility function, along which rationality can be made somewhat more realistic with a rigorous formalization. He also suggests the use of heuristics in decision making instead of a rigid rule of optimization [1].

Ariel Rubinstein proposed that bounded rationality should be modeled by explicitly specifying decision-making procedures including deciding the method and time of the decision. Gerd Gigerenzer opines that decision theorists have not really adhered to Simon's original ideas. Rather, they have assumed that either decisions are crippled by limitations or people try to cope with their inability to optimize. Gigerenzer proposes and shows that simple heuristics often lead to better decisions than optimality [1].

Edward Tsang suggests that computational intelligence can be used to measure degree of rationality of a decision-maker. This is based on a proposal that decision procedure can be encoded in algorithms along with heuristics. An agent with higher computational intelligence can make choice nearer to optimality than one with lower algorithms and heuristics, considering everything else being equal [1].

6.1.2 Bounded Rationality and Behavioral Psychology:

Kahneman [2] along with others attempted to explore the psychology of intuitive beliefs and choices and examined their bounded rationality. The research obtained a map of bounded rationality by distinguishing between the beliefs of people (including optimal one) and the choices assumed in rational-agent models. This was the start and the main source of their null hypotheses. They had three research programs: (1) First explored the heuristics used by people along with the biases to make decisions under uncertainty, including predictions and evaluations of evidence, (2) The second was related with prospect theory, risky choice model and, (3) The third one dealt with framing effects and their implications.

6.1.2.1 Architecture of Cognition:

Kahneman [2] distinguishes two modes of thinking and deciding, of reasoning and intuition. Reasoning is related to the cognitive process undertaken while computing the product of 17 by 258. Intuition similar to the gists, essential meaning. Reasoning is a deliberate and effortful process, but intuitive thoughts come spontaneously with almost no effort of search or computation. Research has given an indication of intuitive thoughts and actions most of the times.

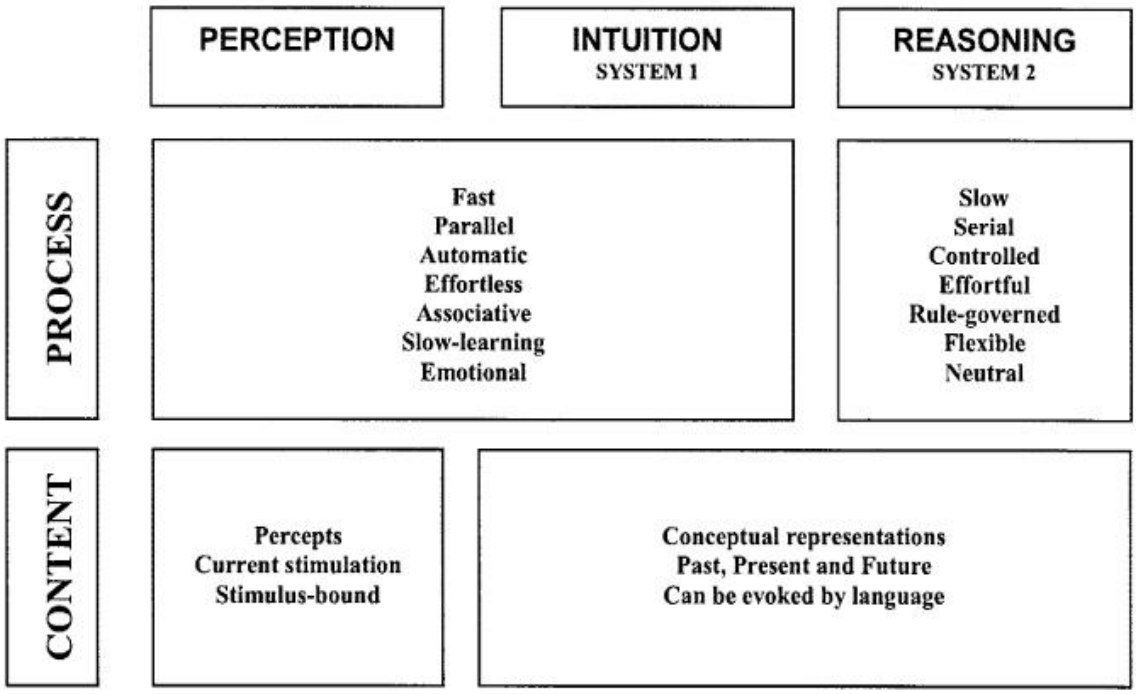

Figure 44 Cognitive system architecture [2]

Figure 44 shows the widely agreed, different characteristics of reasoning and intuition. These two types of cognitive processes were proposed (Stanovich and West 2000) to be neutrally labeled as System 1 and System 2. The scheme shown in Figure 49 summarizes these characteristics. The operations of System 1 are uncontrollable as they are governed by habit, influenced by emotions, fast, automatic, effortless, and associative; while the System 2 has slower, serial, rule-governed, effortful, and deliberately controlled. Though, sometimes System 2 is faster than System 1 (Levine, Ramirez Jr., McClelland, Rebecca Robinson, & Krawczyk, 2014).

The most useful characteristic that can be used to classify a mental process into System 1 or System 2 is the difference in effort. With total capacity being limited, effortful processes interfere with each other, whereas effortless processes do not while multi-tasking. As an example, a skillful driver can do driving and talking simultaneously with very less effort. Different tasks require different amount of attentional demands and different levels of involvement of System 1 and System 2. For example, the self-monitoring function belongs to System 2. People occupied in a demanding mental activity would respond to another task by first instantaneous thought. Both, System 1 and System 2 can deal with stored information (experiences and knowledge). Thus, it can be hypothesized that a high numeracy person can deal with probabilities with just intuition; while the person low in numeracy would have to use reasoning [2].

6.1.2.2 Dimension of Accessibility:

Kahneman explains the accessibility dimension in [2]. It is a technical term indicates that how easily thoughts come to mind. Some thoughts are accessible and others are not.

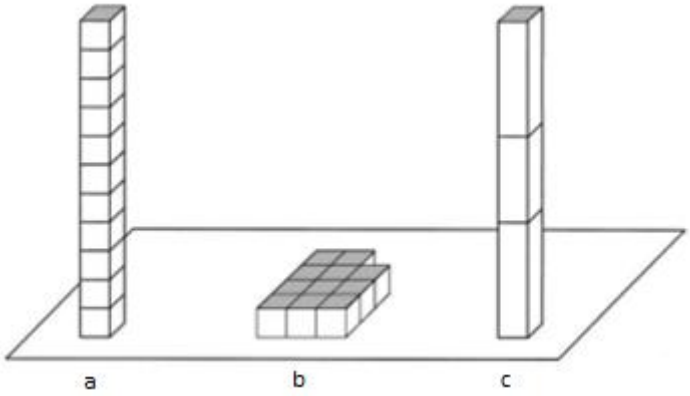

Figure 45 Different accessibility dimension example [2]

The first ideas that come to mind by looking at the figure 45 are the height of the tower and the area of the top surface. These two impressions are highly accessible, though calculating height or area would require a deliberate operation. For the figure 45a, other attributes such as the area covered by blocks of the dismantled tower is not accessible, though it can be estimated

by a deliberate procedure by multiplying the area of a block by the number of blocks. The situation is reversed with figure 45b. An impression of total area is immediately accessible, but the height of the tower constructed from is inaccessible. Additionally, some relational properties like, similarity (height) between figure 45a and figure 45c and dissimilarity (height) between figure 45b and figure 45a (or 45c) are easily accessible [2]. This relates to bounded rationality as the accessibility of dimension are bounded.

References

1. Bounded Rationality from Wikipedia

2. Daniel Kahneman, “Maps of Bounded Rationality: Psychology for Behavioral Economics”, THE AMERICAN ECONOMIC REVIEW, 2003, pp. 1449-1474.

3. Francesco Castellaneta, “The boundaries of bounded rationality: Experience, superstition and the weight of activity load in management buyouts”, Strategic Management Journal, 2014, pp. 1-41.

4. David B. Fogel, Kumar Chellapilla, and Peter J. Angeline, “Inductive Reasoning and Bounded Rationality Reconsidered”, TRANSACTIONS ON EVOLUTIONLARY COMPUTATIONS, VOL. 3, NO. 2, JULY 1999, pp. 142-146.

5. Jaap Vreeswijk, Jing Bie, Eric van Berkum, Bart van Arem, “Effective traffic management based on bounded rationality and indifference bands”, IET Intelligent Transport Systems, Vol. 7, Issue 3, 2013, pp. 265-274.

6. Hyunggon Park, Mihaela van der Schaar, “On the Impact of Bounded Rationality in Peer-to-Peer Networks”, IEEE SIGNAL PROCESSING LETTERS, VOL. 16, NO. 8, AUGUST 2009, pp. 675-678.

7. Daniel S. Levine, Patrick A. Ramirez Jr., M. Michelle McClelland, Rebecca Robinson and Daniel C. Krawczyk, “Ratio Bias Revisited: Behavioral and Brain Imaging Studies of Ratio Comparison Tasks”, Thinking and Reasoning, 2014 (In review).

8. Valerie F. Reyna, “A New Intuitionism: Meaning, Memory and Development in Fuzzy Trace Theory”, Judgment and Decision Making, Vol. 7, No. 3, 2012, pp. 332-359.

6.2 Metacognition

6.2.1 Introduction

Metacognition is "cognition about cognition", or "knowing about knowing". It involves knowledge about timing and methodology to use a strategy for problem solving. An important form of metacognition is 'metamemory', which is described as knowing about memory and mnemonic strategies [1].

J. H. Flavell first used the word "metacognition". In his words: Metacognition refers to one's knowledge concerning one's own cognitive processes or anything related to them, e.g., the learning-relevant properties of information or data. For example, I am engaging in metacognition if I notice that I am having more trouble learning A than B; if it strikes me that I should double check C before accepting it as fact.

—J. H. Flavell (1976, p. 232).

A. Demetriou, used the term hypercognition to referring to self-monitoring, self-representation, and self-regulation processes which participate in general intelligence, and processing efficiency and reasoning. Metacognitive skills include study skills, memory capabilities, and the ability to monitor learning. These capacities are used to regulate one's own cognition, maximization of thinking potential, and learning and evaluation of rules [1].

In cognitive neuroscience, it has been found that the prefrontal cortex monitors and controls metacognition by receiving sensory signals from other cortical regions and through feedback loops, respectively. In the domain of artificial intelligence and modeling, metacognition is of interest of emergent systemics. It is used to denote self-awareness of mortality [1].

6.2.2 Components of Metacognition

Metacognition is generally classified into three components [1]:

1. Metacognitive knowledge is awareness about own self others as cognitive processors.

2. Metacognitive regulation is the regulation of cognition and learning experiences through a set of activities.
3. Metacognitive experiences are those experiences that related with the current, on-going cognitive endeavor.

Metacognition involves active control over the process of thinking that is used in learning. Planning of the approach to a learning task, monitoring comprehension, and evaluation of the progress towards the completion of a task are some of the metacognitive skills [1].

6.2.2.1 Metacognitive Knowledge

Metacognition includes at least three types of metacognitive awareness [1]:

1. Declarative Knowledge: The knowledge about oneself as a learner and factors affecting performance, "world knowledge".
2. Procedural Knowledge: The knowledge about executing things. It is displayed as heuristics and strategies. A high degree of procedural knowledge can allow individuals to perform tasks more automatically by more efficient access to a large variety of strategies.
3. Conditional knowledge: understanding of the right moment and reason to use declarative and procedural knowledge. It facilitates students to allocate their resources using more effective strategies.

6.2.2.2 Metacognitive Regulation

Similar to metacognitive knowledge, metacognitive regulation or "regulation of cognition" contains three important skills [1]:

1. Planning: refers to the appropriate selection of strategies and the correct allocation of resources that affect task performance.
2. Monitoring: refers to one’s awareness of comprehension and task performance

3. Evaluating: refers to appraising the final product of a task and the efficiency at which the task was performed. This can include re-evaluating strategies that were used.

Similarly, sustaining motivation and effort till the task end along with the awareness of both internal and external distracting stimuli involve metacognitive functions. Students high in metacognition can perform more efficiently with fewer strategies without any prior knowledge [1].

6.2.2.3 Metacognitive Experience

Metacognitive experience is creates the identity of one's self and it is linked to motivation. Identity is important as it provides a support for meaning making and for action. Importance of identity is determined with meta-cognitive experience by comparing its worth to other important identities and whether and then its pursuance or abundance is decided. Metacognitive difficulty is considered as a link to abandoning identity. Theories like the theory of incremental ability put difficulty as a way to pursue an identity in which when effort is important, more effort required is termed as difficulty; while the entity theory of ability describes the opposite that when effort is not of importance, suspension of effort is considered difficulty due to the lack of that ability.

.

References

1. Metacognition from Wikipedia

Bibliography

1. Abdesselam Bouzerdoum, “Classification and Function Approximation using Feed-Forward Shunting Inhibitory Artificial Neural Networks”, IJCNN, 2000, pp. 613-618.

2. Bobby Ojose, “Applying Piaget’s Theory of Cognitive Development to Mathematics Instruction”, The Mathematics Educator, Vol. 18, No. 1, 2008, pp. 26-30.

3. Brenda R. Jansen, Anna C. Duijvenvoorde, Hilde M. Huizenga, “Development of Decision Making: Sequential versus Integrative Rules”, Journal of Experimental Child Psychology 111, 2012, pp. 87-100.

4. Christian Balkenius, Stefan Winberg, “Cognitive Modeling with Context Sensitive Reinforcement Learning”, Proceedings of AILS 04, 2004, pp. 10-19.

5. Daniel Kahneman, “Maps of Bounded Rationality: Psychology for Behavioral Economics”, The American Economic Review, 2003, pp. 1449-1474.

6. Daniel S. Levine, “Emotion and Decision Making: Short-term Reactions versus Long-term Evaluations”, International Joint Conference on Neural Networks, 2006, pp. 195-202.

7. Daniel S. Levine,” Modeling the Evolution of Decision Rules in the Human Brain”, International Joint Conference on Neural Networks, 2006, pp. 625-631.

8. Daniel S. Levine, “Neural Dynamics of Affect, Gist, Probability, and Choice”, Cognitive System Research, Science-Direct, 2012, pp. 57-72.

9. Daniel S. Levine, “Seek Simplicity and Distrust it: Knowledge Maximization versus Effort Minimization”, KIMAS talk, 2007.

10. Daniel S. Levine, Britain Mills, Steven Estrada, “Modeling Emotional Influences on Human Decision Making under Risk”, IJCNN Special Session, Vol. 3, 2005, pp. 1657-1662.

11. Daniel S. Levine, Leonid I. Perlovsky, “A Network Model of Rational versus Irrational Choices on a Probability Maximization Task”, IJCNN, 2008, pp 2820-2824.

12. Daniel S. Levine, Patrick Ramirez, “An Attentional Theory of Emotional Influences on Risky Decisions”, Progress in Brain Research, Vol. 202, Elsevier (Book), 2013.

13. Daniel S. Levine, Patrick A. Ramirez Jr., M. Michelle McClelland, Rebecca Robinson and Daniel C. Krawczyk, “Ratio Bias Revisited: Behavioral and Brain Imaging Studies of Ratio Comparison Tasks”, Thinking and Reasoning, 2014 (In review).

14. David B. Fogel, Kumar Chellapilla, and Peter J. Angeline, “Inductive Reasoning and Bounded Rationality Reconsidered”, Transactions On Evolutionary Computations, Vol. 3, No. 2, JULY 1999, pp. 142-146.

15. Edward T. Cokely, Colleen M. Kelley, “Cognitive Abilities and Superior Decision Making under Risk: A Protocol Analysis and Process Model Evaluation”, Judgment and Decision Making, Vol. 4, No. 1, February 2009, pp. 20–33.

16. Francesco Castellaneta, “The Boundaries of Bounded Rationality: Experience, Superstition and the Weight of Activity Load in Management Buyouts”, Strategic Management Journal, 2014, pp. 1-41.

17. Ganesh Arulampalam, Abdesselam Bouzerdoum, “A Generalized Feedforward Neural Network Architecture for Classification and Regression”, Neural Networks 16, 2003, pp. 561–568.

18. http://www.wikipedia.org/

19. Hyunggon Park, Mihaela van der Schaar, “On the Impact of Bounded Rationality in Peer-to-Peer Networks”, IEEE Signal Processing Letters, Vol. 16, No. 8, August 2009, pp. 675-678.

20. Jaap Vreeswijk, Jing Bie, Eric van Berkum, Bart van Arem, “Effective Traffic Management based on Bounded Rationality and Indifference Bands”, IET Intelligent Transport Systems, Vol. 7, Iss. 3, 2013, pp. 265-274.

21. J. W. Curtis, Randal W. Beard, “Satisficing: A New Approach to Constructive Nonlinear Control”, IEEE Transactions on Automatic Control, Vol. 49, No. 7, July 2004, pp. 1090-1102.

22. Joshua Brown, Daniel Bullock, Stephen Grossberg, “How the Basal Ganglia use Parallel Excitatory and Inhibitory Learning Pathways to Selectively Respond to Unexpected Rewarding Cues”, The Journal of Neuroscience, 1999, pp. 10502–10511.

23. Kenji Doya, “What are the Computations of the Cerebellum, the Basal Ganglia and the Cerebral Cortex?”, Neural Networks 12, 1999, pp. 961-974.

24. Kenji Doya, Hidenori Kimura, Aiko Miyamura, “Motor Control: Neural Models and System Theory”, International Journal of Applied Mathematics and Computer Science, Vol. 11, 2001, pp. 101-128.

25. Kenji Doya, Hidenori Kimura, Mitsuo Kawato, “Neural Mechanisms of Learning and Control,” IEEE Control System Magazine, Vol. 21, 2001, pp. 42-54.

26. Kenji Doya, Kazuyuki Samejima, Ken-ichi Katagiri, Mitsuo Kawato, “Multiple Model-Based Reinforcement Learning”, Neural Computation, 2002, pp. 1347-1369.

27. Kumpati Narendra, Jeyendran Balakrishnan, “Adaptive Control using Multiple Models”, IEEE Transactions on Automatic Control, Vol. 42, No. 2, 1997, pp. 171-187.

28. Mathieu Bertin, Nicolas Schweighofer, Kenji Doya, “Multiple Model-based Reinforcement Learning Explains Neuronal Dopamine Activity”, Neural Networks 20, 2007, pp. 668–675.

29. Matthew Nokleby, Wynn Stirling, “Attitude Adaptation in Satisficing Games”, IEEE Transactions on Automatic Control, Man, and Cybernetics, Part B: Cybernetics, Vol. 39, No. 6, December 2009, pp. 1556-1567.

30. Mehdi Khamassi, Stéphane Lallée, Pierre Enel, Emmanuel Procyk, Peter F. Dominey, “Robot Cognitive Control with a Neuro-physiologically Inspired Reinforcement Learning Model”, Forntiers in Neurorobotics, Volume 5, Article 1, 2011, pp. 1-14.

31. Michael A. Goodrich, Wynn C. Stirling and Richard L. Frost, “A Theory of Satisficing Decision and Control,” IEEE Transactions on Automatic Control, Man, and Cybernetics-Part A: Systems and Humans, Vol. 28, No. 6, November 1998, pp. 763-779.

32. Norikazu Sugimoto, Jun Morimoto, Sang-Ho Hyon, Mitsuo Kawato, “The eMOSAIC Model for Humanoid Robot Control”, Neural Networks, 2012, pp. 8-19.

33. Okihide Hikosaka, Hiroyuki Nakahara, Miya K. Rand, Katsuyuki Sakai, Xiaofeng Lu, Kae Nakamura, Shigehiro Miyachi, Kenji Doya, “Parallel Neural Networks for Learning Sequential Procedures”, TINS Vol. 22, No. 10, 1999, pp. 464-471.

34. Paul J. Werbos, “Handbook of Intelligent Control: Neural, Fuzzy and Adaptive Approaches”, Van Nostrand Reinhold, 1992, pp. 493-525.

35. Paul J. Werbos, “Intelligence in the Brain: A Theory of How it Works and How to Build it”, Neural Networks 22, 2009, pp. 200-212.

36. Paul J. Werbos, “Neural Networks and the Human Mind: New Mathematics Fits Humanistic Insight”, IEEE International Conference on Systems, Man, and Cybernetics, vol. 1, 1992, pp. 78-83.

37. Paul J. Werbos, “Using ADP to Understand and Replicate Brain Intelligence: the Next Level Design”, IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning, 2007, pp. 209-216.

38. Paul Werbos, “What is Mind? What is Consciousness? How Can We Build and Understand Intelligent Systems?”, Werbos’ website.

39. Petru E. Stingu, Frank L. Lewis, “Neuro Fuzzy Control of Autonomous Robotics”, Springer (Book), 2009.

40. R. Nieuwenhuys, J. Voogd, C. van Huijzen, “The Human Central Nervous System”, Forth Edition, Springer (Book), 2008.

41. Ranuifo Romo, Eugenio Scarnati, Wolfram Schultz, “Role of Primate Basal Ganglia and Frontal Cortex in the Internal Generation of Movements-II. Movement Related Activity in the Anterior Striatum”, Experimental Brain Research 91, 1992, pp. 385-395.

42. Ranuifo Romo, Wolfram Schultz, “Role of Primate Basal Ganglia and Frontal Cortex in the Internal Generation of Movements-III. Neuronal Activity in the Anterior Striatum”, Experimental Brain Research 91, 1992, pp. 396-407.

43. Samuel J. Gershman, Jonathan D. Cohen, Yael Niv, “Learning to Selectively Attend”, Cognitive Science Society, 2006, pp. 1270-1275.

44. Surachai Charoensri, H.W. Corley, “The Scalarization and Equivalence of Standard Optimization Criteria”, COSMOS Technical Report 11-03.

45. Tetsuya Minatohara, Tetsuo Furukawa, “The Self-Organizing Adaptive Controller”, International Journal of Innovative Computing, Information and Control, Vol 7, number 4, April 2011, pp. 1933-1947.

46. Valerie F. Reyna, “A New Intuitionism: Meaning, Memory and Development in Fuzzy Trace Theory”, Judgment and Decision Making, Vol. 7, No. 3, 2012, pp. 332-359

47. Wolfram Schultz, Ranuifo Romo, “Role of Primate Basal Ganglia and Frontal Cortex in the Internal Generation of Movements-I. Preparatory Activity in the Anterior Striatum”, Experimental Brain Research 91, 1992, pp. 363-384.

48. Wolfram Schultz, Leon Tremblay, Jeffrey R. Hollerman, “Reward Prediction in Primate Basal Ganglia and Frontal Cortex”, Neuropharmacology 37, 1998, pp. 421-429.

49. Wolfram Schultz, Leon Tremblay, Jeffrey R. Hollerman, “Reward Processing in Primate Orbitofrontal Cortex and Basal Ganglia”, Cerebral Cortex, 2007, pp. 272-283.

50. Wynn C. Stirling, “Satisficing Games and Decision Making”, Cambridge University Press (Book), 2003.

51. Wynn C. Stirling, Michael A. Goodrich, “Satisficing games”, Information Sciences 1999, pp. 255-280.

Biographical Information

Patanjalikumar Shashankkumar Joshi graduated with a degree of Master of Science in Electrical Engineering from University of Texas at Arlington in December, 2014. He has worked at University of Texas at Arlington Research Institute under the supervision of Dr. Frank L. Lewis from January, 2013 to December, 2014. He earned his Bachelor of Engineering degree in Electronics and Communications Engineering in July, 2011 from Sardar Patel University, India. His area of interests are control systems, automation and robotics. He is also interested in embedded microcontroller/microprocessor systems.