

# The *Intelligent Entity* Project

Nicholas A. Pape  
Westlake High School  
Austin, Texas  
pape.nick@gmail.com

Paras Chopra  
New Delhi Institute of Technology  
New Delhi, India  
paras1987@gmail.com

May 16, 2010

## Abstract

This paper is the documentation for the Intelligent Entity (aka Artificial Being) Project. It is the documentation of the authors' research and development efforts over the past several years. This journal is organized chronologically from the start of the project to the last new writings.

## Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Paras' Introduction . . . . .	4
1.2	Old Introduction . . . . .	4
1.3	New Introduction . . . . .	4
<b>2</b>	<b>Paras' Writings</b>	<b>6</b>
2.1	November 7, 2002 . . . . .	6
2.1.1	The <i>Artificial Being</i> Project . . . . .	6
2.1.2	Introduction . . . . .	6
2.2	November 8, 2002 . . . . .	7
2.2.1	Idea for an Artificial Being . . . . .	7
2.2.2	Diagram . . . . .	8
2.2.3	Algorithm . . . . .	8
2.2.4	Learning Method . . . . .	9
2.2.5	Limitations . . . . .	9
2.3	November 9, 2002 . . . . .	9
2.3.1	Abandoning the 11/8/2002 Theory . . . . .	9
2.3.2	Characteristics of an Artificial Being . . . . .	9
2.3.3	Method of Coding the AB . . . . .	10
2.4	November 14, 2002 . . . . .	10
2.4.1	The Law of Consciousness . . . . .	10
2.4.2	Implications of the Law . . . . .	10
2.4.3	Machines can't be conscious because... . . . . .	10
2.4.4	Critical Consciousness . . . . .	11
2.4.5	New Direction in the Development of AB . . . . .	11
2.5	November 24, 2002 . . . . .	11
2.5.1	The Relative Theory of Consciousness . . . . .	11
2.5.2	Feedback to the Theory . . . . .	11
2.5.3	Software called SHRLDU . . . . .	12
2.6	December 10, 2002 . . . . .	12
2.6.1	Evolving Languages . . . . .	12
2.6.2	Structure of AB . . . . .	12
2.7	December 28, 2002 . . . . .	13
2.7.1	Can Genetic Algorithms be useful? . . . . .	13
2.8	January 13, 2003 . . . . .	14
2.8.1	Introducing... . . . .	14
2.8.2	Explanation of the GD Network . . . . .	14

2.8.3	Diagrams . . . . .	15
2.9	January 17, 2003 . . . . .	16
2.9.1	Features of the GD Net . . . . .	16
2.9.2	The Feedback Net . . . . .	17
2.9.3	Reinforcement Learning . . . . .	17
2.10	February 23, 2003 . . . . .	17
2.10.1	Components of AB . . . . .	17
2.11	February 24, 2003 . . . . .	19
2.11.1	Consciousness Might be an Illusion . . . . .	19
2.12	July 6, 2003 . . . . .	20
2.12.1	Creativity . . . . .	20
2.12.2	Another Proposed Theory for AB . . . . .	21
2.13	<b>Note to Readers</b> . . . . .	22
2.14	December 24, 2005 . . . . .	22
2.14.1	Feedback to the Theories . . . . .	22
2.15	November 2, 2006 . . . . .	23
2.15.1	New Developments... . . . .	23
2.15.2	Democracy as a Paradigm for Artificial Intelligence . . . . .	24
<b>3</b>	<b>Nick's Writings</b> . . . . .	<b>25</b>
3.1	September 2006 . . . . .	25
3.1.1	September 1, 2006 . . . . .	25
3.1.2	September 2, 2006 . . . . .	25
3.1.3	September 3, 2006 . . . . .	25
3.1.4	September 4, 2006 . . . . .	26
3.1.5	September 5, 2006 . . . . .	27
3.2	Winter 2006 . . . . .	28
3.2.1	October 21, 2006 . . . . .	28
3.2.2	October 28, 2006 . . . . .	28
3.2.3	November 1, 2006 . . . . .	29
3.2.4	November 2, 2006 . . . . .	30
3.2.5	November 7, 2006 . . . . .	31
3.2.6	December 26, 2006 . . . . .	31
3.2.7	February 19, 2007 . . . . .	32
3.3	Spring 2007 . . . . .	33
3.3.1	March 6, 2007 . . . . .	33
3.3.2	March 10, 2007 . . . . .	33
3.4	2009 on... . . . .	34
3.4.1	March 30, 2010 . . . . .	34

# 1 Introduction

## 1.1 Paras' Introduction

The reason why Artificial Intelligence is so much interesting task is because the nature of human mind. The human mind, always, wants to see what is beyond its limit and what's not. It then wants to conquer the limits to expand the boundaries of its knowledge. The basic reason for curiosity is not known to me (or anyone?) but curiosity is the most important factor which made human to progress to such a great extent and still progressing.

The AI field is hot and will remain hot until a saturation point is reached when there will be such smart robots that they cannot be made smarter. That saturation point is not very near nor very far off, with respect to current work in field of AI; it may be 2150 when smarter-than-human robots will arrive.

## 1.2 Old Introduction

One day in July 2006 I was looking for code that would allow me to create my own Neural Networks, and I ran across Paras Chopra's website, and I look around at it. I came across an interesting project called the "Artificial Being" project. I was so interested that on September 1<sup>st</sup>, 2006 I began emailing him to discuss ideas. This is my story since then. You can view his project journal at his website, [www.paraschopra.com](http://www.paraschopra.com).

He considered this project, which he started November 7<sup>th</sup>, 2002, dead by now, but I came by to make it come back!

## 1.3 New Introduction

Long in the history of humanity have we sought after machines which could do our bidding. Originally, we designed static mechanical devices, but increasingly, robotics researchers have turned to electronics to solve their problems and develop their machines. A robot is simply a system of parts, engineered in such a way as to achieve a certain task, usually deemed hazardous or extremely repetitive to humans. Increasingly, Artificial Intelligence has been incorporated into the research of roboticists, and this union has the only potential to create a form of intelligence, not otherwise created by evolutionary and biological means. Thus, Artificial Intelligence is a key field in the future of technology and has extreme immediate commercial benefits.

And so, we have come to the task of designing systems which are, in fact, deemed as truly artificially intelligent. Thus is the key goal of the project of all AI researchers.

In the Intelligent Entity project, Aerion Systems hopes to drive systems of the future, by doing research and experimentation into the fields of both robotics and artificial intelligence. The immediate goals and scope of the project are laid out in more detail within the body of the document.

## 2 Paras' Writings

### 2.1 November 7, 2002

#### 2.1.1 The *Artificial Being* Project

Today, I'm going to set a goal for myself. The goal is to create a complete artificial being with full human like capabilities. It will be conscious, intelligent, self sustaining organism. I take the pledge that what may come I'll complete this project and will not deviate myself from this task. I am a Lover of Artificial Intelligence and will remain a lover.

Goal: To create an Artificial Being just like Human Beings.

Difficulties: The following will be major hurdles in achieving this goal:

1. Lack of sufficient time
2. Lack of knowledge in this particular subject
3. Lack of resources

#### 2.1.2 Introduction

The reason why Artificial Intelligence is so much interesting task is because the nature of human mind. The human mind, always, want to see what's beyond its limit and what's not. It then wants to conquer the limits to expand the boundaries of its knowledge. The basic reason for curiosity is not known to me (or anyone?) but curiosity is the most important factor which made human to progress to such a great extent and still progressing. **IDEA:** The Artificial Being must be, always, curious for information and knowledge in order to progress.

The AI field is hot and will remain hot until a saturation point is reached when there will be such smart robots that they cannot be made smarter. That saturation point is not very near nor very far off, with respect to current work in field of AI; it may be 2150 when smarterthanhumans robots will arrive. But the question arises: *Why would we want to develop such systems which are smarter than we are?* I dont know why Im pursuing this project, but it could be possibly that there is some inner feeling in humans to do whatever we can and to try everything which we cant.

At first, this project seems to be impossible but developing an Artificial Being is not as difficult as it seems. If we want to code some system as intelligent as we are first of all we'll have to study our own brain. 'Till now we have not been able to know how brain works, so *how can we model the brain in computers?* Even before this question another question arises: *what's the difference between the brain and mind?* Brain is hardware or framework and mind is the software which is working on it. It's better to code brain and mind separately in computer. The mind could be possibly be:

1. Complex arrangement of information and knowledge inside brain
2. Virtual; Not existing at all
3. Some external force driving it (we're deviating from science a little)
4. May be existing in 5<sup>th</sup> or 6<sup>th</sup> dimension, and not discovered by humans yet

From above alternatives the most significant is the first one. If we consider brain as complex arrangements of roads and knowledge as cars then the mind may be a kind of router which regulates the flow of cars along the roads and helps them to reach their final destination. But *can this complex arrangement give rise to consciousness?*

## 2.2 November 8, 2002

### 2.2.1 Idea for an Artificial Being

For an AB to exist there must be 5 elements present. They are:

1. Environment
2. Input or senses
3. Memory or rule book
4. Output or motor
5. A goal meter

**Environment** It consists of all that data which can be fed into the AB through its senses and can be interacted by the AB.

**Input** It consists of various senses like sound, text, vision, etc. There may be any number of senses.

**Output** It consists of various output devices like text, muscles, internal thought, arms, legs, etc.

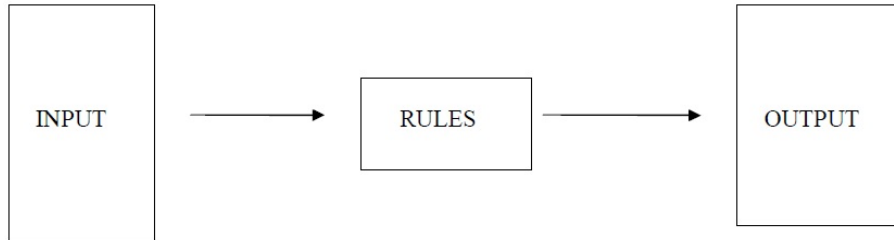
**Memory** It just records the inputs, outputs and the feedback they get.

**Goal Meter** Goal meter is the basic goal for which an organism lives. It may be health, age, progress, happiness, etc. If there are more than two major goals like happiness and health then there will be two goal meters.

### 2.2.2 Diagram

The basic idea for an AI being is as follows:

Documents/Programming/Writings/CS Projects/ieProject/pics/block1.jpg



### 2.2.3 Algorithm

The algorithm is as follows:

1. The input from various senses arrive
2. If any sense data is new its preference is set to initial sense value
3. To that data random motor action is linked
4. If it gets positive feedback (from goal meter) the link is strengthen otherwise it is weaken
5. All this gets recorded into the memory
6. Otherwise if the sense data is already there, that motor output is carried which is accompanying with highest sense data preference.

### 2.2.4 Learning Method

The basic learning method will be random-feedback method. In this method when new data is sensed, a random action will be taken and after that goal meter will be checked if goal meter has increased that specific random action will be encouraged for that specific data if goal meter decreases that action will be discouraged.

### 2.2.5 Limitations

1. Static.
2. Doesn't learn from past mistakes.
3. Too limited in interaction with the environment.
4. Not flexible in its rules.
5. Not clever.

## **2.3 November 9, 2002**

### **2.3.1 Abandoning the 11/8/2002 Theory**

I thought about the theory very much after writing it. I found it inadequate. Therefore, due to all its limitations I'm abandoning this theory. However, you may use it if you find it interesting.

### **2.3.2 Characteristics of an Artificial Being**

1. It should be able to reprogram itself
2. It should be able to change its learning method during its lifespan
3. There should be some method of reproduction.
4. The concept of ego should be pre-programmed in its genes in order for it to progress.
5. There should be some method to pass down knowledge to next generation.
6. Its interaction with the environment should not be limited in anyway, if limited its development will be limited.
7. The society must pass on the knowledge to others.

### **2.3.3 Method of Coding the AB**

The inclusion of Neural Networks or some method relating to it should be avoided because if NN's are used then the method of coding the AB would be called plagiarism.

## **2.4 November 14, 2002**

### **2.4.1 The Law of Consciousness**

Consider that you want to code software which is conscious. Then there is no method through which you can verify that the software is conscious. This is because of mind-body problem. Even if the software considers itself as conscious (first person perspective), but when you'll examine the internal processes of the software you'll find nothing but code and data relating to that software (third person perspective). Just because of data and information, software can't be regarded as conscious because if regarded then every existing software would be conscious. In case, that specific software tries to convince you that he/she/it is conscious, you would probably assume but not be 100% sure that he/she/it is conscious because on re-examining it, the software would still be code and information only and nothing new. Therefore you could only assume that the software is conscious and can never really be sure that the system actually considers itself as conscious.

This law states that *'The consciousness, in system(s) other than the system itself, cannot be verified it can only be assumed to be present'*.

#### **2.4.2 Implications of the Law**

1. Maybe existing software are conscious
2. Maybe we are not conscious
3. Maybe a rock is also conscious

#### **2.4.3 Machines can't be conscious because...**

1. We are not able to define the term 'consciousness' very correctly. This term is very ambiguous.
2. When we ourselves don't know what consciousness is how we can code into the machine?
3. The law of consciousness states *'The consciousness, in system(s) other than the system itself, cannot be verified it can only be assumed to be present.'* No matter how hard we try to code consciousness into the machine we can never be really sure that it is conscious.

#### **2.4.4 Critical Consciousness**

Due to the Law of Consciousness, our artificial being can never be truly conscious ever. In this way no machine can ever be conscious but definitely it can aim to achieve something near consciousness. Let us call that state 'Critical Consciousness'. Thus the state is very near to the real consciousness but it is not actually conscious. This is called 'Critical Consciousness'. This critical consciousness can definitely be achieved by the machines because we can only assume that a machine is conscious and that assumption is not provable (or disprovable).

#### **2.4.5 New Direction in the Development of AB**

Aim to achieve critical consciousness.

### **2.5 November 24, 2002**

#### **2.5.1 The Relative Theory of Consciousness**

The consciousness is not absolute but it is relative. The consciousness is how we perceive the world and respond to it. You are conscious because you feel your presence in the world through the environment's reaction. Do you think a fish is conscious? And then, do you think a piece of steel is conscious? You would certainly say that a fish may be conscious but steel can't be. That's because consciousness is not absolute but is relative. You consider fish as a conscious

being because she responds to your action in some way, but a piece of metal doesn't. The same might be happening with the metal. The metal might be considering itself as conscious but not us. In this way, existing software might be conscious in its own dimensions but not in ours.

### **2.5.2 Feedback to the Theory**

Very Interesting. I also believe that awareness could be divided into levels — sort of like your dimensions — only each level is determined by intelligence. Level 0 would be death. Level 1 would be alive, but not aware; the only instinct is to survive like like animals and insects. Level 2 would be humans: we have the belief that we are aware but we can only figure it out by discovery. Level 3 would be AI; an increase in intelligence would be better able to understand the next level, and so on and so on. Make sense? It's a little messed up I know. But what do you think? Also watch the movie 'Colossus'. It is an excellent movie about AI becoming aware. It's old (1950's) but has good acting.

Heliopic Monoptic

### **2.5.3 Software called SHRLDU**

I have come to know about software called SHRLDU that just resembles an Artificial Being. We can make SHRLDU our role model in the process of development of the Artificial Being. SHRLDU basically is a simulated robot which has its own little environment. It can perceive this environment and can give a motor output also. The environment consists of him and little boxes of different kinds of boxes of various kinds of big and small boxes of various colors. It can perform specified number of actions on it. E.g. picking a red box from the table, putting small blue box onto large green box. The user can give it commands in natural language like 'Pick the small box.', 'Place the green box over large red box', 'Open the bottommost box', etc. The user can also ask questions like 'Where the red box', 'Why is the blue box inaccessible'. It can even learn new words from the user like 'put means place' and it also remembers some of the last actions performed by it.

## **2.6 December 10, 2002**

### **2.6.1 Evolving Languages**

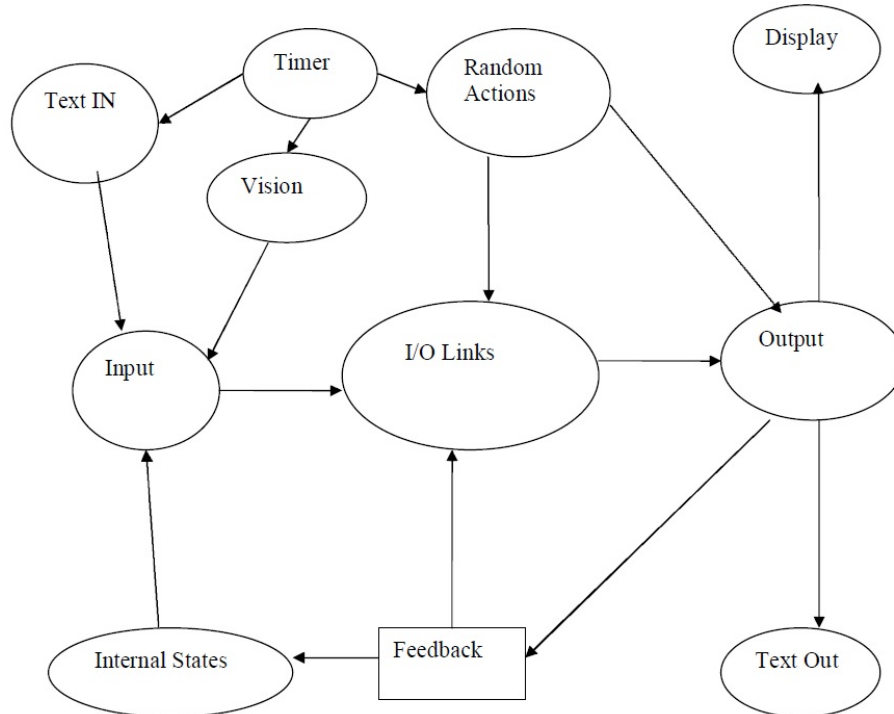
The biggest difficulty in the development of the AB is Natural Language Processing (NLP). It is because we'll have to define vocabulary and grammar for it which is a very tedious and difficult task. One solution for this problem is that allowing AB to evolve its own language. Thus, we would allow AB to evolve its own language by relating words to its current and opponent's state

and emotions. Thus gradually the whole community of AB's will build up its own language.

### 2.6.2 Structure of AB

The following is the proposed structure of the Artificial Being:

Documents/Programming/Writings/CS Projects/ieProject/pics/block2.jpg



## 2.7 December 28, 2002

### 2.7.1 Can Genetic Algorithms be useful?

I don't think so that GA's will be useful for the AB. This is because GA's are static by nature. This means GA's can only be trained (or evolved) once and after that they may not change themselves. We need a system which is capable of modifying itself e.g. Neural Networks. Or we need to create some new Self-Modifying system which can learn from its environment.

Conclusion: The method of constantly evolving the brain (Neural Network) of the AB may prove to be useful.

## 2.8 January 13, 2003

### 2.8.1 Introducing...

**Dual-Decision, Self-Training, Cooperating, Feedback Neural Networks (Group Discussing networks)** I have invented these new kinds of neural networks. This network consists of two individual neural networks, which work simultaneously to provide answer. It is just like two persons trying to solve a particular problem and thus giving each individual's views. It is better to have various views on a particular problem and then decide the solution according to it. Just as we, humans, can change our learning methods, these networks can also change there own learning method without any human intervention.

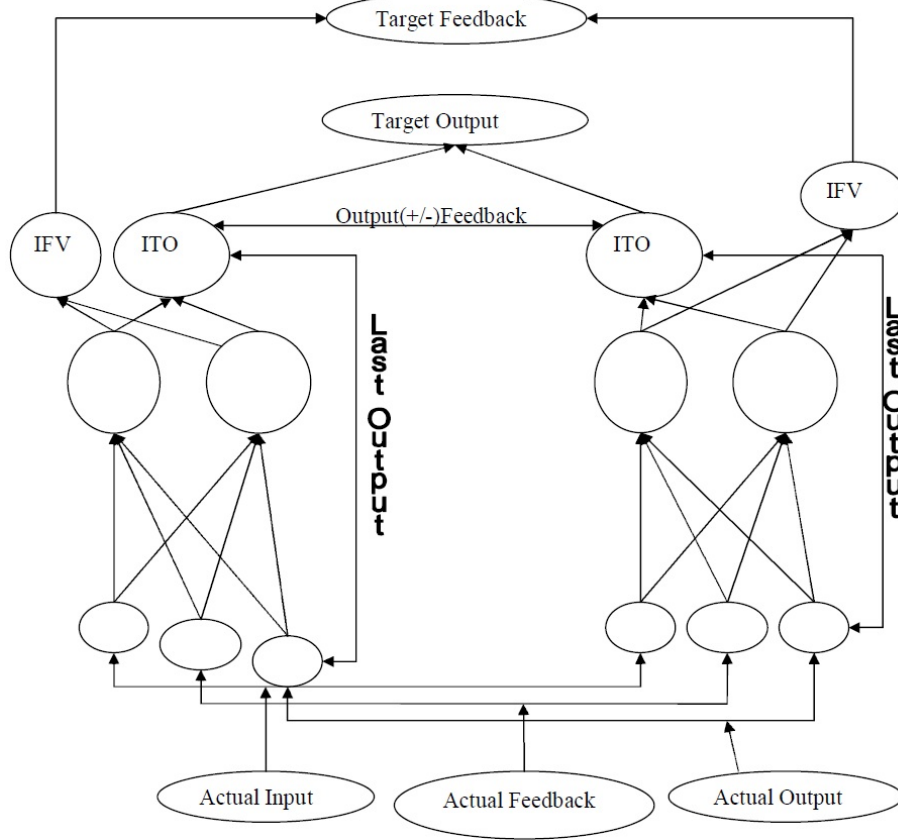
### 2.8.2 Explanation of the GD Network

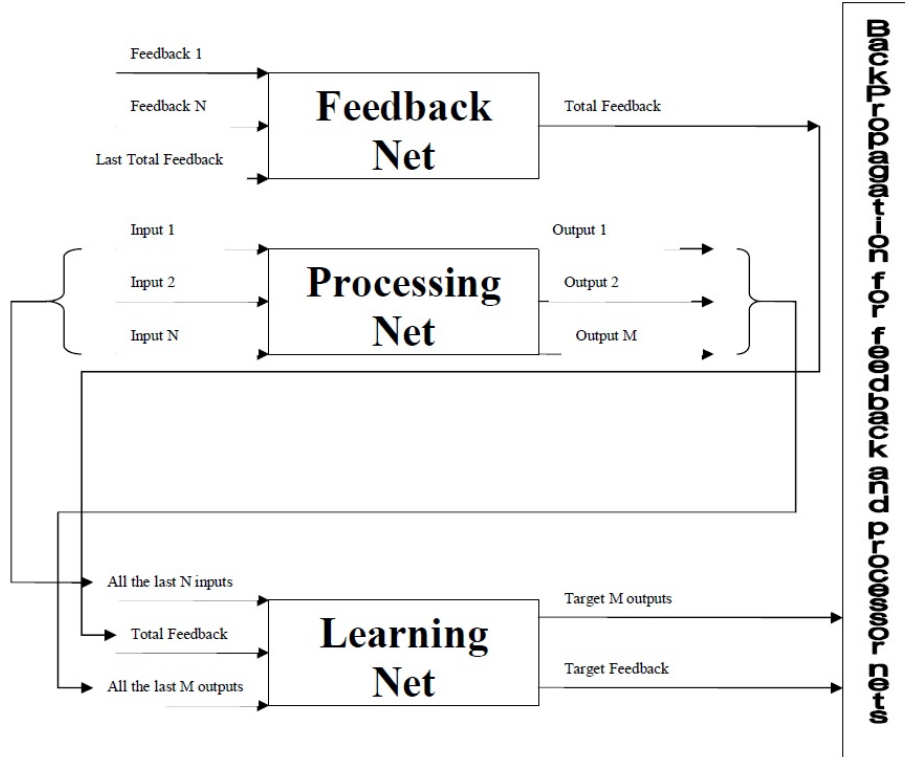
The GD net consists of following three parts:

- Learning Net
- Processing Net
- Feedback

### 2.8.3 Diagrams

Documents/Programming/Writings/CS Projects/ieProject/pics/GDnet1.jpg





## 2.9 January 17, 2003

### 2.9.1 Features of the GD Net

- It uses unsupervised learning.
- It uses BackProp to train the nets.
- It does not need the user to supply target values for various nets.
- The target values are supplied by the system itself.
- Every parameter and feature of this net is changeable and flexible.

### 2.9.2 The Feedback Net

The feedback is method of telling the GD net that whether the output produced by the system is correct or not. It also tells the effects of that output on that system itself. For example, for an ALife net, feedback may be health, age, happiness, sadness, etc. The feedback value can lie between 1 and  $-1$ . It has to be provided by the user (or environment). There can be any number of

these individual feedbacks to the system and the task of the feedback net is to combine these all values in some or the other way to produce a single feedback.

### **2.9.3 Reinforcement Learning**

I think reinforcement learning is the type of learning which was wanted to have in AB. Learning through environment feedback was wanted and exactly this can be achieved through Reinforcement Learning. (This topic needs more attention).

## **2.10 February 23, 2003**

### **2.10.1 Components of AB**

- Environment
- Inputs
  - Sensors
  - Current state of effectors
- Feedback
  - Pain
  - Pleasure
  - Energy
- Memory
  - Short term
  - Long Term
- Focus or Attention
- Goals
  - Short term goals
  - Long term goals
  - Life time goals
- Motivation to
  - Create
  - Invent
  - Discover
  - Survive
  - Achieve goals

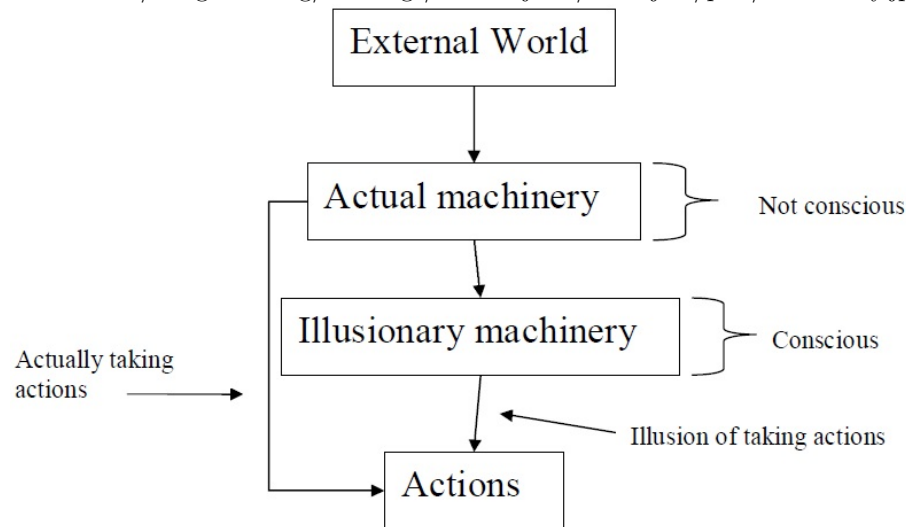
- Increase energy
- Desire
  - To increase pleasure and energy
  - To decrease pain
- Feature detectors
- Decision makers
- Effectors
- Learning
  - Unsupervised
  - Supervised
  - Self supervised
  - Competitive
  - Reinforcement
- Learning to learn
- Emulation of
  - Past events
  - Future events
- Emotion engine : Directly changing value of pain or pleasure to show emotions.
- Hard core reflexes
- Right and Left networks : Two individual networks each with whole network and each controlling each another just like right and left brains in humans.
- Meta-cognition: Ability to modify any parameter of network like goals, learning, its thinking process, desire, motivation and meta-cognition itself.
- Sleep
- Dreams

## 2.11 February 24, 2003

### 2.11.1 Consciousness Might be an Illusion

Sometimes it happens that we take some action e.g. we laugh but later, after a few microseconds, we realized why we laughed. This happened to me very recently. This prompted me to think that there must some kind of machinery which controls the whole thinking process without letting us know about what it is thinking. Much later after that machinery completes the thinking process that it lets us know only about a fragment of what that machinery was doing. Thus it creates illusion for us to live in. The following diagram will make this concept clear:

Documents/Programming/Writings/CS Projects/ieProject/pics/illusionary.jpg



Note: the term 'machinery' in the diagram is just for representation, the terms processes, systems, or any other suitable terms could have been used.

From the above figure it should be clear that the 'actual machinery' receives inputs from the world processes the inputs. When all the thinking processes have been finished it sends only partial processed information to the 'illusionary machinery'. The actions to be taken are already included in the information received. The 'illusionary machine' just extracts that information and creates an illusion that it is thinking and taking actions. Actually all the thinking is done by 'actual machinery' and it is only who takes actions in practice.

The thoughts taking place in 'illusionary machinery' are in language whereas thought taking place in 'actual machinery' is in form of electric signals. Just because of this phenomenon we are not able to tell how we think or what is consciousness and all these kinds of questions.



- Short Term Memory for HLB (STMH)
- Long Term Memory for LLB (LTML)
  - Global memory
  - Personal memory
- Long Term Memory for HLB (LTMH)
  - Global memory
  - Personal memory
- Self analyzer or consciousness
- Learning method

## 2.13 Note to Readers

**April 9, 2005** Document will always remain under construction.

People, this project is a community project. All humanity must take part to develop the idea of 'Artificial Being' further. If you want to suggest changes, additions or removals, please contact the author at [www.paraschopra.com](http://www.paraschopra.com).

You must also contact the author prior using any concept presented in the document.

## 2.14 December 24, 2005

### 2.14.1 Feedback to the Theories

Yesterday, I posted this document on Google Groups. I got an email from this person regarding this document. He has an interesting idea of putting nodes between modules (such as LTMH, Global memory, etc.) which can route and condition data between modules. And this nodes will mainly be responsible for the learning of AB. Nice proposition. I may try to integrate or develop AB on this model.

Hello Dude,

a) Interesting how far you've come, apparently on your own, and/or with little awareness of what's already been done.

Thanks...

b) You ask for contact before 'using concepts' in the article. Trouble is, many of those concepts are widely known and disseminated. Eg, the notion that short/long term memories are located in different 'modules' is old (and in many ways outdated, ie, there are serious problems with it.)

Oops.. I didn't know about it. Maybe I'll remove that notice. But, some of the The Artificial Being Project concepts are my original, and in fact most of the concepts are original. I just don't know if they are already been proposed and if they are outdated.

c) Your block diagrams of an AB are interesting to me because they resemble block diagrams I drew many years ago, when I was thinking about a career in computer science.

Great!

Specifically, multiple feedback and propagation paths seemed to me obviously necessary for an 'artificial being' (I called it a robot.) My diagrams were even messier than yours, with multiple entry points in the loops themselves. That is, I posited routing nodes **between** 'modules', whose function was to 'decide' where the signals should go. I had a vague notion that these routing nodes would pass/block data to the modules depending both on the signals' strength and on the 'requests' from the modules.

Ok. So you used the concept of routing between modules. Interesting stuff!

Now that I write this, it seems to me that these nodes could/should also condition the data, ie, change its strength and/or shape, and thus 'alert' the receiving modules to the 'significance' of the data. Further, these nodes could change the way they condition the data, thus they could be the parts that enable the learning. That means they would have to be 'sensitised' to certain kinds of data, ie, they would have to be hardwired to do certain things (just as a biological brain is hardwired to do certain things.) They would have to be connected directly to some sensor(s) as well; and they would have to resist future changes to a greater or lesser extent, else learning would not be 'retained.'

Hmm... Perhaps you may allow the modules to influence the way nodes condition data. So there can be a feedback loop. Nodes helping modules run/learn and modules providing feedback to nodes about how efficiently they are performing thus indicating the efficiency of nodes performance. And I don't really think that hardwiring is a good idea. Because we as programmers might not be able to decide the best way to wire the brain. Why not let evolution do this stuff for you?

Please note that I am an amateur: I like thinking about possible structures that might implement various kinds of thinking, but I couldn't specify hardware or write software to actually do it.

Same with me, but perhaps I can write software but I cannot engineer hardware.

From *Paras Chopra* to *Wolf Kirchmeir*.

## 2.15 November 2, 2006

### 2.15.1 New Developments...

A long time has passed since I last wrote in this document. I thought the Artificial Being is dead because I did not have enough time to devote to its development. But a guy named Nick Pape wanted to develop it further. I thought it was a brilliant idea; now I am codeveloping the Artificial Being whom Nick calls *Intelligent Entity*.

### 2.15.2 Democracy as a Paradigm for Artificial Intelligence

I have lately realized that Artificial Intelligence can be best achieved if it is modeled on democracy. Democracy works in a decentralized manner and all the decisions it makes go through rigorous debate. The best of all, there is no single entity responsible for decision; hence no scope of blaming things to one entity.

The AB (simple single celled type) will be composed of several modules with different priorities/weights. Say, module 'motion', 'hunger', 'vision', etc. When vision sees a food it gives the analysis of trajectory to food. Then motion estimates the energy required to grab up the food and vision estimates amount of energy in the food. And hunger tells if the endeavor will be worthwhile.

Something like this:

Hunger says Yes  
Motion says No, path too difficult  
Vision says Yes, a tasty pizza ahead

They would decide among themselves about the final decision: if AB will spend energy to fetch food. Each modules vote their consent for a proposal made by module X. Then, taking into the votes and the weight age attached to voting modules, X decides if to go ahead with the proposal. In a nutshell, AB will work like a democracy. I guess our brains work like democracy too!

Therefore, an AB will consist of N number of modules each with a weightage for other modules. Therefore there will be  $N(N - 1)$  weights for an AB.

Now, to get the ideal values of weightage we can put a lot of AB's in a common environment to compete for a limited resource such as energy. The best ones survive and reproduce; worst ones perish (a Genetic Algorithm).

## 3 Nick's Writings

### 3.1 September 2006

#### 3.1.1 September 1, 2006

This Project Officially Created. On this day I have proposed several architectural plans. I am planning on using the Modular Brain Model for development. I started development in Visual Basic .NET. I created the Theory\_V1.doc file to explain theory.

#### 3.1.2 September 2, 2006

Project development files wiped. Must create new project and re-develop. My goal is to have a simple version created by September 30. By the end of the year, I hope to have a more advanced product created. I also hope that by April 15th, I will have a stable post-production product available for commercial use.

I re-developed software. I got farther than before. I also created flowchart to help with development. I also started writing the paper on the creation. The development was going fine, but I have encountered a major problem; Natural Language Processing. I need to find a way to use the smallest amount of code possible to recognize these words.

I have almost successfully run the first code into the program.

#### 3.1.3 September 3, 2006

I changed the Command Line. I am hoping now I will be able to process Natural Language. I am also creating a natural language for use with James (The name of our IE). I also re-created the flowchart for use with EnAct. I also started development with EnAct. EnAct works well with this project. I will have to do a bunch of grammar stuff that will be programmed into hard code. I am going to get some sleep because it's 4 in the morning!

I have also decided that I will not be using a general public license until we get the code patented. Then we will use a license compatible with the patent. I might end up using the Apache License ([http://en.wikipedia.org/wiki/Apache\\_License](http://en.wikipedia.org/wiki/Apache_License)).

I have succeeded in programming a basic vocabulary machine. The current machine has a few limits; it doesn't do definitions or multiple synonyms. I also have not made the program parse modifiers of modifiers (i.e. 'very fast' must be 'fast'). I hope to have these corrected by tomorrow night.

### 3.1.4 September 4, 2006

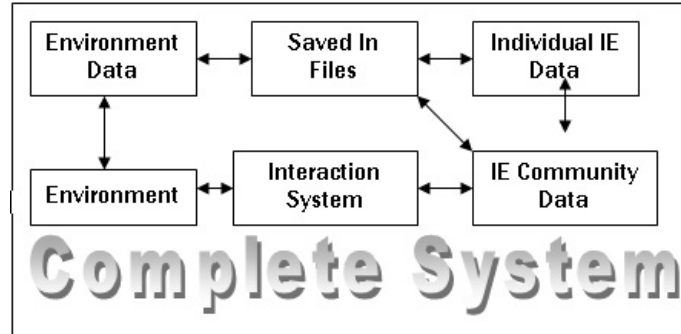
I fixed the problem of only one synonym. I am now working on the way that the words are linked to memories. I am working it to where you can send three types of messages to the IE, a question, a statement, and an input. I will have inputs treated as system values. An input will look like this: xx xx. The xx represents a 2 digit number. 1 would be 01, 2 are 02, etc. View the environment read me for environment information. I am making the inputs check for changes and then use fuzzy logic to decide the message data.

Please note that one of the things I am trying to do is disregard all the current AI methods and create my own for use with this system. The natural language values are saved in nlfuzzy.txt. Please view this file before running the program.

I am now trying a different approach where the IE is geared to surviving. This is called the Small IE Experiment. There will be no further development today due to time restrictions.

The main problem I am having is having an environment where the IE is in, I must make an environment based on what I want the IE to do. This is a major issue; I must resolve what I want the first IE to do. I will now create a new project that uses the old code, but will use a special environment, and then I will create another, and so on. I do not believe you can have an all use entity unless you create different ones and somehow fuse them together. I believe that the main problem with development is that you must define an environment; for now I will develop them individually, but maybe the program could load a file containing environment data, this data could also possibly be changed. Remind me to look into this. I am going to now re-develop a template where I may start development on many diverse projects; maybe a homework helper, a business decision maker, or a small person that interacts with multiple beings. I believe that the most interesting project would be where there are multiple IEs and their data is saved individually along with the environment, this would allow for a next generation to come forward. I believe that this would be a good way to model, study, and predict human behavior. Please view diagram 1; it will make things clearer for you.

Documents/Programming/Writings/CS Projects/ieProject/pics/block4.jpg



Note that all the code that we program does it to decipher the files, use them for decisions, run the IE interactions, and evaluate the result. Please note that now; the files are what make up the IE, not the code. Also, this system works one IE as well as multiple IE's.

The items saved in the IE files would be: Memory, Vocabulary, Genetics, Sensors, and decision variables. The environment saves all changes to it as well as every decision that an IE has made. I believe this is a great way to solve many problems that we had. To make an IE work for a new purpose, change the environment. Also note that multiple IEs would produce better results.

### 3.1.5 September 5, 2006

Today I started to develop my new theory. If you don't understand it, imagine it where all the files go into the program.

Later today, I put aside this theory to test the other development project I had. Please view the IE.World.pdf file for project Information. I have got very far in this project to where it loads a map. Here are my goals for each night:

- Ability: To move across map (Wed.)
- Program: Movement Decision Machine (Thurs.)
- Functionality: To Save and Load IE files (Fri.)
- Enlarge: Map and Map Engine (Sat.)
- Ability: To communicate with other IE's (Sun.)
- Ability: To create new items (Mon.)
- Functionality: Body Sensors (Tues.)
- Functionality: Genetics (Weds.)

- Ability: To evolve (Thurs.)
- Functionality: Natural Language Processing (Fri.)

## 3.2 Winter 2006

### 3.2.1 October 21, 2006

I have come up with several new systems and ideas for this project. I will complete the logic for the system before I start to program it because I already tried that and it didn't work. So, here are some ideas I have before starting this. What I will do is break up everything in the bulleted list and program it into separate systems. The first two systems I will be creating are the communication and genetic mutation evolution systems. Here are my first ideas:

- The Genetic system
- A simple peer-peer communication system

### 3.2.2 October 28, 2006

I was re-reading Paras' tutorial on Neural Networks, but here is the large problem, the reason why we have been stuck at a wall at every idea. The problem is that the computer thinks mathematically, it does equations, communicates, does everything mathematically; we need to find a method of communication as well as defining an environment and defining the being. We must make the intelligent entity aware of itself and its surroundings; therefore, we must come up with a method of defining the following:

In the environment I think it is a waste of time (for now) to program meaningless things like Light and Temperature, I think we must define a map, a graph of where it should live. I think we have been trying to push ourselves too far by trying to create something that models a real-life human; all we want is the Human Behaviour. I think that therefore, we should model a human mind, and then create a simple environment for which it reacts.

So we have yet to define the new model of the human mind. What we need to do is have different sectors of the mind.

- Mathematical Sector – This sector allows for the conscious computation of mathematical problems.
- Logic Sector – Allows for conscious computation of logical operands (I.E. or, not, xor). This sector will be limited to math first.
- Output Sector This sector takes internal data and puts it into a 'waiting bay' where the data then gets processed into the re-designed EnAct (Please view the EnAct page).

Ok, I was just talking to Paras and he has a new idea for a system. I believe it is something like this, Mod A says, he I want a bike. Then that is transferred to Mods. B, C, and D. B says yes, C says yes, but D says no, I want a motor scooter. So A says yes. So then B, C, and D say yes, so then we get a motor scooter. So a basic flowchart would be like this:

1. A Proposes X
2. Mod A says yes or no. If 'No', then alternate answer, if alternate Y, then A can say 'Yes', in which case the Mods. vote on Y, otherwise, if A says no, Y & X are killed.
3. Mod B says yes or no. If 'No', then alternate answer, if alternate Y, then A can say 'Yes', in which case the Mods. vote on Y, otherwise, if A says no, Y & X are killed.
4. Mod B says yes or no. If 'No', then alternate answer, if alternate Y, then A can say 'Yes', in which case the Mods. vote on Y, otherwise, if A says no, Y & X are killed.
5. When everyone settles on an answer, it is passed.
6. We believe this system would allow for specialized sections to allow for separate ideas, but this also may lead to endless debates. Also, this theory could be used for IE communication.

### **3.2.3 November 1, 2006**

Okay, so this is what is going to happen:  
The first system we will be putting in place will be a combination of the US government, my sector idea, and Paras' equal representation idea. Assuming that we would want to make the system more intelligent, we would assign dynamic weights to the sectors, for example, if a system gives a bad/incorrect response, that module will have its weights lowered on that connection to the proposing module. To give you an idea, x is the proposing and y is another sector (e.g. math) and x says, 'let's run around in circles', and math says, 'No way, let's listen to music.', and we run around, math's weight to sector x will be raised, but let's say math says to go shoot somebody, that is a bad decision, and its weights are lowered.

There are two problems I can think of now. Number one would be that according to the models presented, we would have the possibility of debating forever. There are several fixes to this problem, after time x, that could be named hesitation, and therefore no action. Second, we could have the whole decision process terminate after a certain amount of time, but I believe this would allow for modules that would be cut off. Third, we could have a module propose that we should end debate with no action, and the other could vote, but this would lead to problems because the modules wouldn't be allowed to 'think',

they would be forced to have a set time limit. I think that what we should do is allow a pattern recognition algorithm implemented into the modules, a module that is neutral would be looking at the pattern and see that maybe 2 sectors are debating 1 and then the module could call for a previous question, all systems would then vote. So, to give an example, there are sectors A, B, C, D, and E. A proposes to go get a drink of water. B says 'Yes', C says 'No, the water tasted bad', A says, 'Yes, I agree', but B Says 'No way, we should drink!' etc. etc. D looks at this pattern and realizes the votes are going Yes, Yes, No, Yes, Yes, No So D comes along and says, 'Previous Question', then they take an immediate vote, not on previous question, but on the actual debate. A, C vote no water, and B, D, and E vote to get water, so the whole system gets water.

Here is the problem (I think I have a solution). So, the problem is that we don't have a concrete message communication system; my first goal was to make it NLP, but now I realize that was too high. Imagine we are going to create a cell, an amoeba. Now I realize this isn't a particularly high goal, and I realize you can't make a 'neural network', but it is a basic animal, so I will use it. Now, this is Amoebas Firstionus. AF is a simple creature, therefore deserving 3 logical sectors. Now AF has several actions: Move Left, Move Right, Move Up, Move Down, Be Dormant, Awaken, Eat and Attack. AF lives in a 9x9 Grid, this grid contains an Amoebas Evilus, who is evil (not to mention more powerful than AF), a piece of rotten food, and a piece of good food). On a spot where there us no object, AF may only move, be dormant, or awake. AF starts on spot (2,2 [center]), Sec. A. says go north, Sec. B. says yes, but Sec. C. says no, C says go south, and then all the other systems agree. I have to keep typing, but that is the basic system of how it will work.

#### **3.2.4 November 2, 2006**

I still think we have a problem with endless debate. Who would be the moderator? But I don't know what to do. Here are some possible solutions:

- A) Set time to cancel debate.
- B) Set time to immediate vote.
- C) Let sectors end votes.

But all these have downfalls, and I need something better, this will definitely be an important part of the project as well as the other problem of a 'universal message system' for all parts of a being to react to, but right now we don't need it as an amoeba is a single-celled simple-DNA organism capable of sustaining minimum life.

#### **3.2.5 November 7, 2006**

I have been thinking very hard about the logic of this program, and earlier today, I decided upon the following 'modules' or democratic representations in the Amoeba:

- **Sight** Sight gives arguments based on observations on all surrounding spaces; sight chooses what is immediately best for the Amoeba. (IE food is a plus for movement in a certain direction)
- **Smell**
- **Memory**
- **Artificial Recognition Machinery (ARM)**

### 3.2.6 December 26, 2006

#### On Amoebas

These are the classic amoebae of the general biology textbooks, and part of the intention of this collection of photomicrographs and observations is to dispel the notion that an amoeba is a shapeless blob of animated jelly, randomly engulfing objects it happens upon en route to nowhere in particular.

It has been known for a long time that the general direction of an amoeba's movement is a chemotaxic response to concentration gradients of nutrient (and other) substances in the surrounding water, and that most amoebae maintain a fairly definite shape, with a recognizable front and back end, during all of their activities.

**Safe-Wandering** The Ability of a Group of Agents to move about while avoiding collisions with obstacles and Each Other.

**Following** The ability of an agent to move behind another retracing its path and maintain a line or queue.

**Dispersion** The ability of a group of agents to spread out in order to establish and maintain some minimum interagent distance.

**Aggregation** The ability of a group of agents to gather in order to establish and maintain some maximum inter-agent distance.

**Homing** The ability to find a particular region or location.

### 3.2.7 February 19, 2007

Ok, in the past few days I have been making a couple changes to the plan. This is how the large weights system works. There is a table with Y being the action (wait, up, down, left, right) and X being the decision modules (AGB (Flowchart, Rough Draft Code), PIC (Programmed, but one section needs to be edited, and a function for what is described here), and CRM (Need to rough

draft, but planned)). The initial value of all the values in the box is 0.5. Each module outputs a main decision, and a secondary decision. Basically, every 'main' value vote equals whatever (Decision, Vote) equals. A secondary vote equals half that value. Then the votes are tallied. The maximum value of the votes is 1 and the minimum is 0.

After the decision is made, CRM finds if that was a smart decision by taking the sum of the movement values in PIC. If the value goes up, then CRM decides that it was a good choice. Then every module with a main vote for that movement goes up .1, and every secondary one goes up the same. If it was a bad choice, all the other decisions are rewarded, while the weights are punished by the same amount.

For example, all the votes are tallied and CRM (Up, Down), AGB (Right, Down), PIC (Left, Down). The value formula is  $.5 * .75 * .5 * .5 * 0$ . Therefore the decision is Down. CRM calculates that this was a good decision, so the values are rewarded by .05.

Essentially, this is a basic method for machine learning. It is not very complicated and has many limitations that will be discussed. The PIC module just takes inputs, multiplies by a table, and sums the values for every region to make a decision.

Documents/Programming/Writings/CS Projects/ieProject/pics/matrix.jpg

<b>Vision:</b>					<b>Smell:</b>				
1	0	-1	-1	-1	-1	0	-1	-1	-1
0	0	0	0	-1	0	0	0	0	-1
-1	0	-	0	-1	-1	-	0	-1	
0	0	0	0	-1	0	0	0	0	-1
0	1	0	0	0	0	1	0	0	0

**Multiplication table:**

1	2	3	2	1
2	3	4	3	2
3	4	-	4	3
2	3	4	3	2
1	2	3	2	1

<b>Vision:</b>					<b>Smell:</b>				
1	0	-3	-2	-1	-1	0	-3	-2	-1
0	0	0	0	-2	0	0	0	0	-2
-3	0	-	0	-3	-3	-	0	-3	
0	0	0	0	-2	0	0	0	0	-2
0	2	0	0	0	0	2	0	0	0

**Final Table:**

0	0	-6	-4	-2
0	0	0	0	-4
-6	0	-	0	-6
0	0	0	0	-4
0	4	0	0	0

**Regions:**

Up:	-2
Down:	0
Left:	-3
Right:	-3

### 3.3 Spring 2007

#### 3.3.1 March 6, 2007

Hello Ya'll. I'm back after a short break and I have some very VERY important comments that I need you guys to understand and at least reply to me about. Here's what it is. After looking at the nonproofed (non-digital) code I believe I have found some fundamental problems and questions. First, I would like to know what you define as intelligence, the definition I was using was the population divided by the number of years (turns) in existence. But I think this answer is a little too shallow. I think the problem with the amoeba is that it is actually not conscious, it is not even conscious of it's own thought process, and I believe this is a fundamental part of intelligence. I have an idea for this, I believe that maybe we need to rewrite not the way the decisions are made, but how they are handled and processed. I think that maybe it is possible to create several layers or shells to the code that would allow one layer to look deeper into a layer that would look deeper and so on. Imagine the following text boxes for the output of the different shell layers. (Also maybe we could incorporate Syllogy into the code to allow one layer to 'read' another layer and understand what it is thinking).

Inner Decision Core	Lower Level Processing	Higher Level Processing
>Recieved Impulse {Decide}	>Decision Scan On	>Lower Level Scan On
>Call PIC Decisions	>Gather Decision Structure Outputs	>The Environment wants decision.
Decisions {Up, Down}	PIC {U, D}	>I am looking at environment.
>Call AGB Decisions	AGB {D, L}	>What's good immediatly?
Decisions {Down, Left}	CRM {L, D}	Up then Down
>Call CRM Decisions	>Adjust based on weight table.	>What about other entities and objects?
Decisions Left, Down	>Tally {.5, 1, .75, 0, 0}	Down then Left
>Tally {.5, 1, .75, 0, 0}	>Call Hesitation	>Is where I am better than last time?
>FD {Down}	>FD {Down}	Left then Down
		>My values are adjusting the weights of my decisions.
		>The final weight tally is {.5, 1, .75, 0, 0}.
		>The weights are close...should I do this?
		Yes
		>Telling Environment

### 3.3.2 March 10, 2007

**From Paras** It seems to me we are reinventing the wheel. The thing which we have planned (the table with Up, Down, etc. on Y axis and CRM, PIC and AGB on X axis) resembles a neural network and the rewarding the values by 0.1 if the decision is good resembles the training procedure of the neural network.

Our process is just analogous to reinforcement learning. Have a look at this: [http://en.wikipedia.org/wiki/Reinforcement\\_learning](http://en.wikipedia.org/wiki/Reinforcement_learning)

I think the next best step would be to see where our theory stands vis-à-vis with already established theories and then move on to refine and code our theory. Perhaps, it would be much better if you can share the code for amoebas' simulation so that we all can know what exactly the status of the project is.

## 3.4 2009 on...

### 3.4.1 March 30, 2010

**Back to GD Nets...** I have a working backpropagation neural network written in Java, and I am using it now in the development of a Group Discussing Network. Inside of the entity, there will be three individual Neural Networks: Feebback, Processing, and Learning. Environment-driven feedback will be sent into the Feedback net, which will combine them to provide a single feedback.

The processing net will take all  $N$  inputs and calculate all  $M$  outputs. The outputs  $M$ , inputs  $N$ , and total feedback will be processed by the learning net to develop target  $M$  outputs, and target feedback. Thus, the feedback and processing nets experience unsupervised learning. The learning network, though, continues to modify itself. We will most likely implement this in a genetic algorithm that will run over the course of possibly a thousand generations.

My only fear is the bulk of the system. I feel like the system could require a massive amount of training. There is possibly some error in the Neural Network that has been developed, as it requires ten thousand iterations to come up with the correct answer. Possibly, the neural network will calculate every half-second and every generation will last 100 seconds.