Human Level Artificial Intelligence

Second Edition

Human level artificial intelligence is the total emulation of the human brain and body. These are self-aware entities that can, think, act, learn information, adapt quickly to new conditions, and experience consciousness and emotions the same way humans do. HLAI is indistinguishable from human cognition and intelligence

By Mitchell Kwok
6/19/2007

Table of contents:

Preface

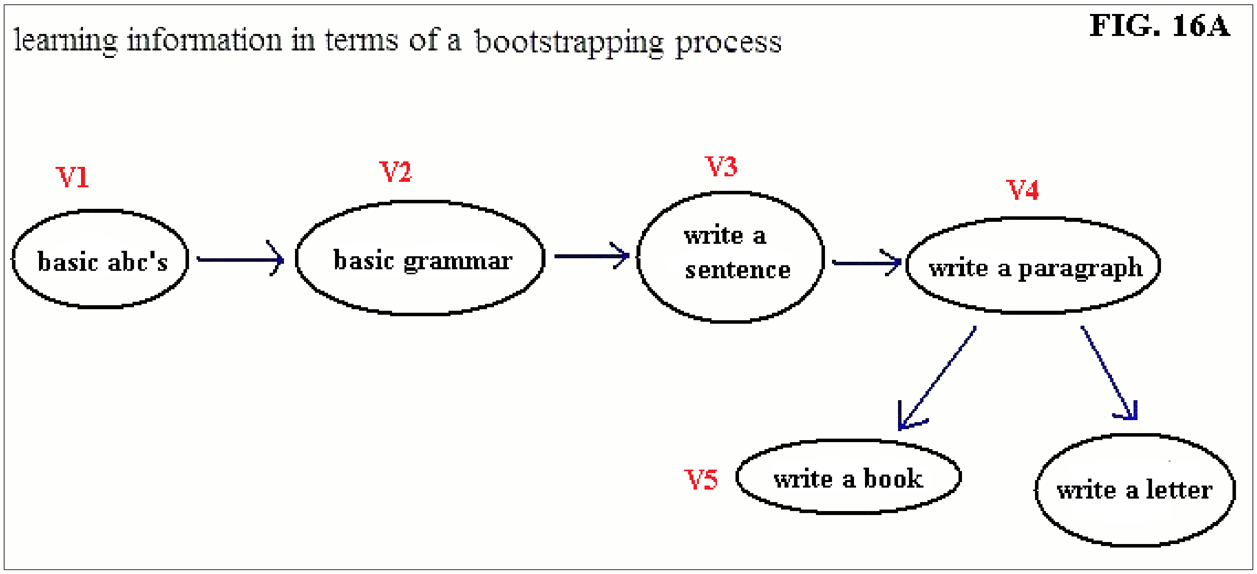
If you read Mit and Stanford University's books on Artificial Intelligence, they separate AI into topics. These topics include: NLP, logical induction and deduction reasoning, decision making, recursive planning, hierarchical planning, predicate calculus, semantic networks, neural networks, decision trees, searching algorithms, and generating common sense knowledge. However, they don't combine these topics together to form one cohesive AI program that can do everything. What is special about this book is that I combine all Artificial Intelligence topics together and present an AI program that is comparable to intelligence of a human brain, with college-level intelligence. Capabilities of said AI program includes: thinking, making decisions, generating common sense knowledge, logical inferencing, solving problems, understanding natural language, learning a new skill by itself, practicing said skill, and making decisions using said skill, and so forth.

Keep in mind the data structure of my invention, a digital human brain, is very very long. In order to understand how my robot works, every chapter has to be read. If you skip a chapter you might miss something important. All major topics in AI is covered in this book.
All knowledge and skills from the robot is learned through teachers in school. This includes: decision making, predicting the future, generating common sense knowledge, mobility skills, induction/deduction, solving problems, understanding natural language, and so on. For example, there are no semantic networks or decision trees in my robot's data structure to make decisions. The knowledge of making decisions is learned through teachers in school. Teachers/parents teach the robot how to make decisions.

Humans learn knowledge from kindergarten to college. Old information is used to learn new information and knowledge from the robot's brain builds on top of each other to form complex intelligence. MIT and Stanford university call this ability: learning information in terms of a bootstrapping process.

Below is a simple example of how my robot learns complex subject matters like English grammar. In FIG. 16A, first, the robot learns to write words and understand basic grammar rules (V1-V2). Next, he takes those skills to write a sentence (V3). Then he takes the knowledge of writing a sentence to write a paragraph (V4). Finally, the robot takes previous knowledge to write a book (V5).

FIG. 16A



A robot must go to school from kindergarten to college to learn knowledge. If the robot graduates from college with a difficult degree, like an engineering degree or computer science degree, then he has achieved human level artificial intelligence. An art degree doesn't count.

Another major topic in this book is the robot's conscious (aka the robot's mind), which is the voice in his head that controls all decision making. This book will invest a lot of time discussing what the human mind is and how it works.

"The voices in a human mind is like an invisible teacher that: gives information, make decisions, alert the host to danger, observe the environment, generate common sense knowledge, predict the future, schedule tasks, manage tasks, solve problems, do induction/deduction, understand natural language, etc. This invisible teacher, which exist in a human mind, was created from a lifetime worth of learning from school, through personal experience, and knowledge from books."

The big question is how can researchers build a robot that can think like a human mind? It took me a very long time to design this Human Level AI program. First I had to figure out how the human mind works. I discovered how the human mind worked by writing my thoughts down in a notebook while doing various tasks, like cooking, driving, playing video games, or watching a movie. After months of collecting data and compiling them into meaningful information, I finally had the basic structure of a digital human brain. After years of refining the AI program, in 2006 I published my first book, entitled: Human Level Artificial Intelligence.

The book you're reading now is my 5th and final book. I decided to call it Human Level Artificial Intelligence, 2nd edition. I did a thorough job in my first 4 books, and I did a great job in explaining the data structure of my AI program to the public. However, over the years, I still have missing subject matters that I didn't fully reveal.

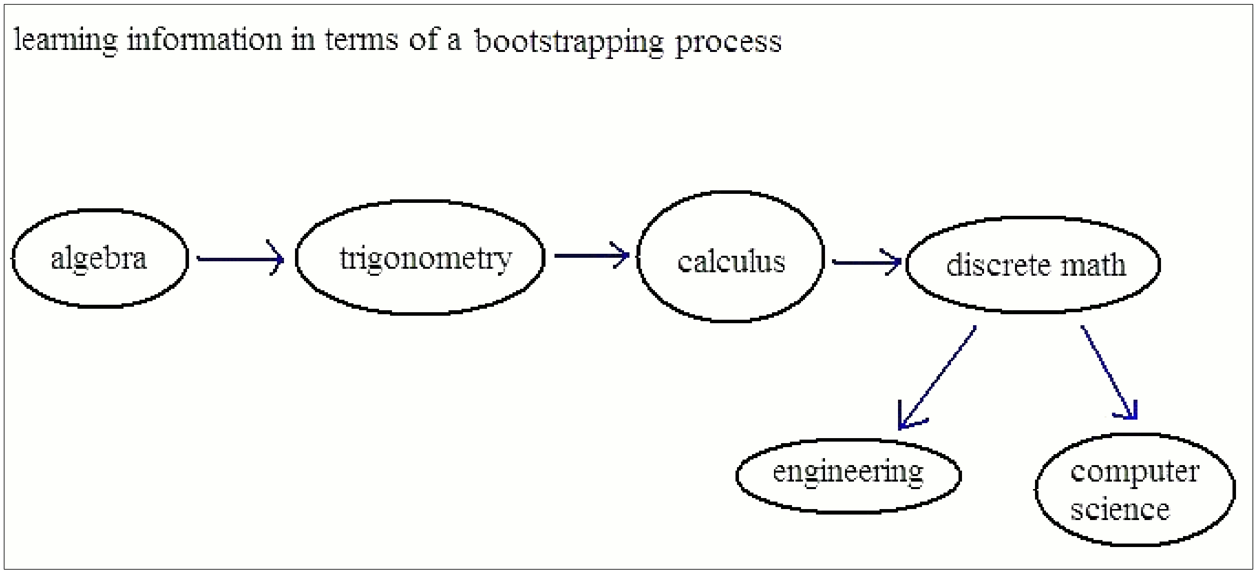
This book is about revealing, in microscopic details, how the human brain works and how to build an Artificial Intelligent program that can mimic human intelligence. My advice to readers is to read the entire book from beginning to ending, chapter1 to chapter13. If you're press for time, read the last chapter, chapter13, because the majority of HLAI is explained there.

Summary of Invention

Scientists from MIT and Stanford University have been trying to teach their robots to learn information in a bootstrapping manner. If the robot learns simple math like addition, how can he use that knowledge to solve a problem? In another example, if the robot learns a complex concept like a binary tree, how can he use that knowledge to write a customer database system?

Humans learn knowledge from kindergarten to college. Old information is used to learn new information and the knowledge in the robot's brain "recursively" builds on top of each other to form complex intelligence.

Humans learn math through a bootstrapping manner, whereby information builds on top of each other to form complex intelligence. First we learn algebra, then we take that knowledge to learn trigonometry. Next, we take trigonometry to learn calculus. Finally, we take calculus to learn discrete math or computer science.



Learning English in a bootstrapping manner.

Language encapsulate entire knowledge and allow the robot to learn different skills in a bootstrapping manner. FIG. 16A shows a diagram depicting how the robot learns to write a book. English sentences are used to encapsulate entire instructions, called pathways.

In FIG. 16A, first, the robot learns to write words and understand basic grammar rules (V1-V2). Next, he takes those skills to write a sentence (V3). Then he takes the knowledge of writing a sentence to write a paragraph (V4). Finally, the robot takes previous knowledge to write a book (V5). Notice in FIG. 16B, said robot is using previously learned skills to write a book. For example, he is repeatedly using basic grammar rules, writing sentences, and writing paragraphs to write a book. As you might recall, writing a paragraph requires writing several sentences and using basic grammar rules.

FIG. 16A

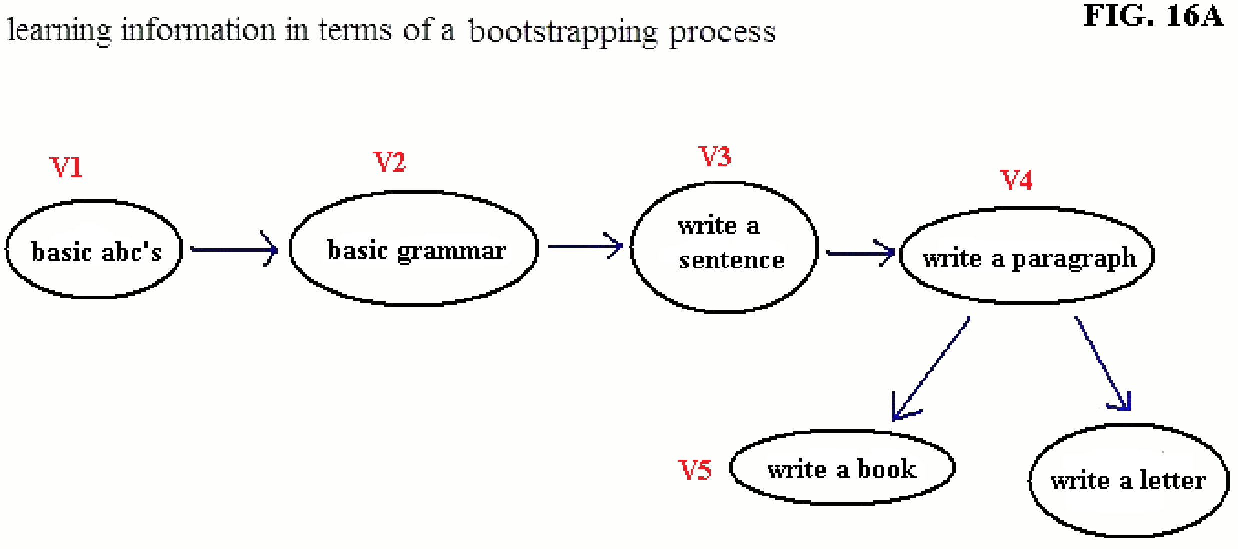
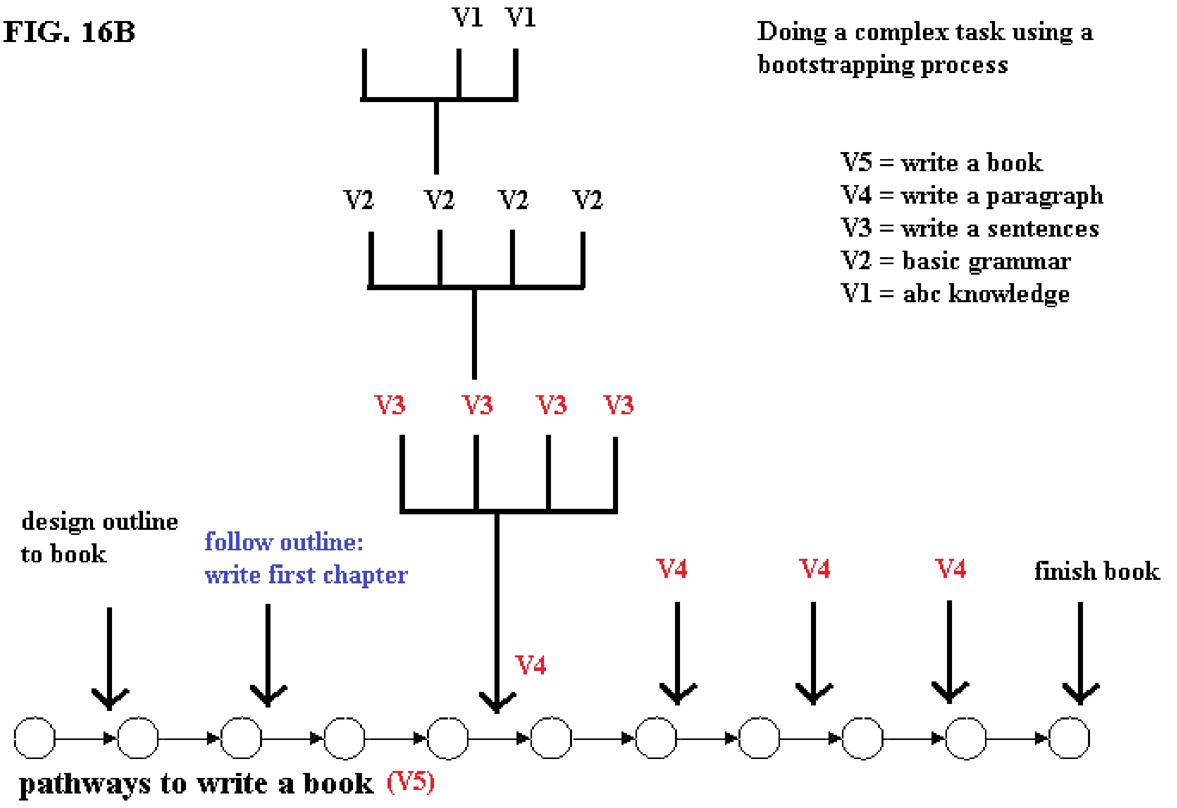


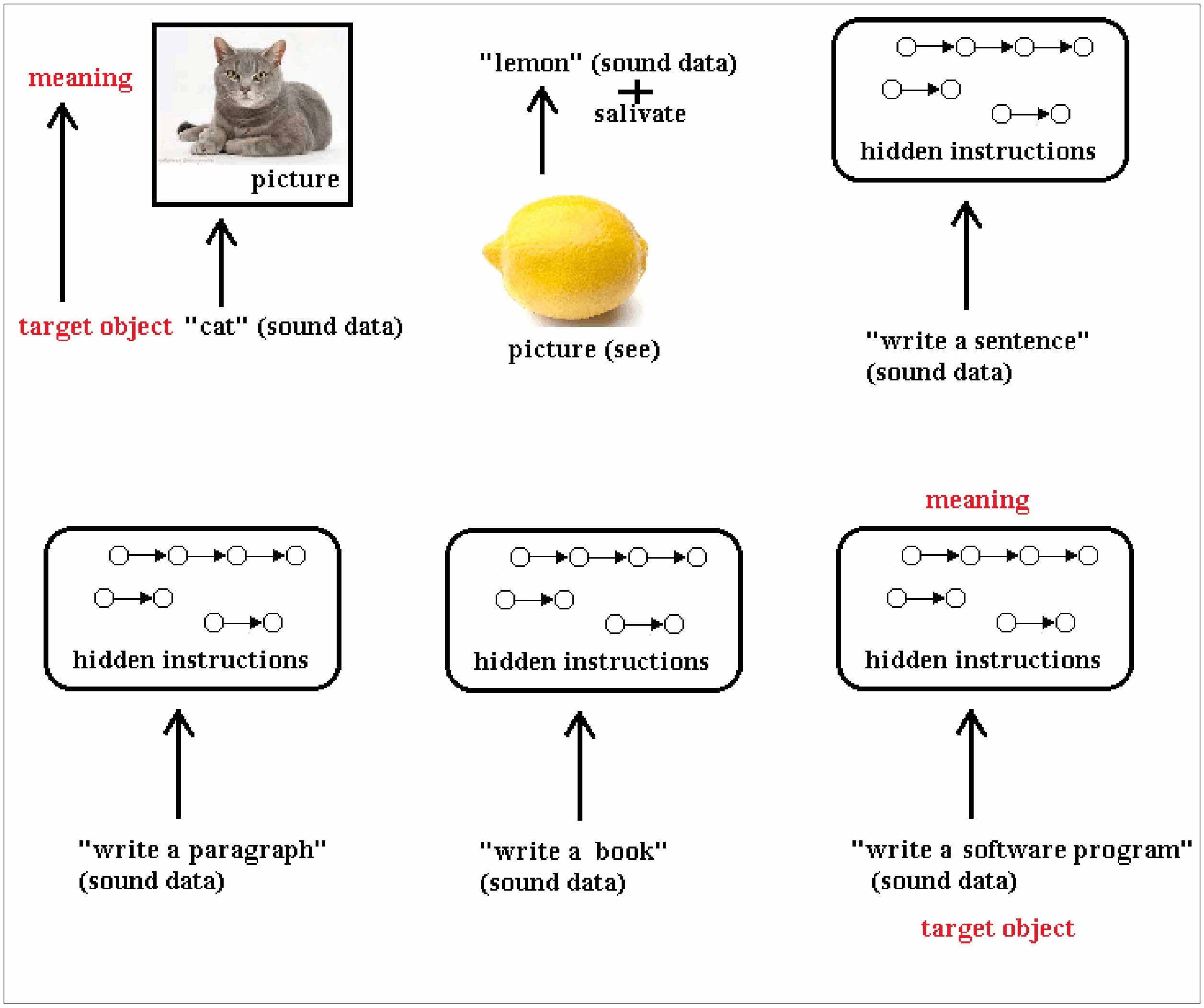
FIG. 16B



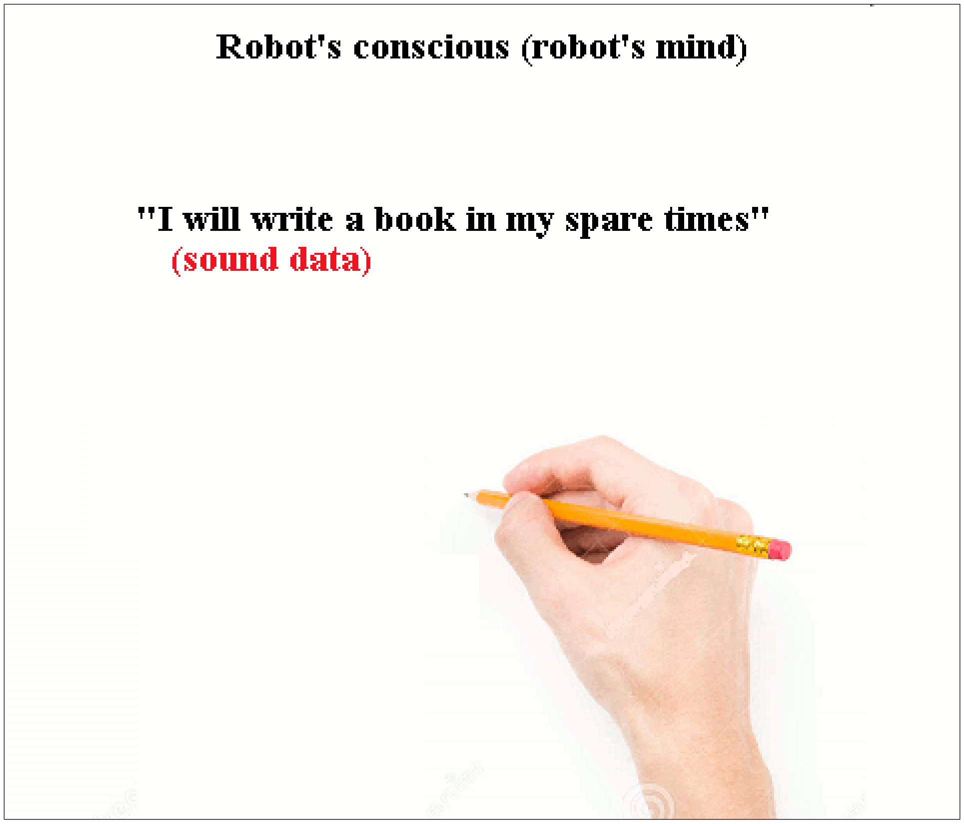
V5 = write a book
V4 = write a paragraph
V3 = write a sentences
V2 = basic grammar
V1 = abc knowledge

English sentences represent a task or sub-task. Sentences is a fixed object that can represent a fuzzy or abstract concept. There are many cats in this world, coming in different sizes, shapes, and 3-d animation, but the word cat is a fixed object to represent a broad range of the species. In this case, sentences are used to represent or encapsulate very complex tasks. This type of encapsulation allows complex intelligence to form in the human brain, and enabling us to solve college level problems.

FIG. 11 depicts several examples of words and sentences representing abstract objects or actions. When the robot senses the target object, the meaning automatically activates in his mind. A simple example is shown in the first 2 items. If the robot hears the sound "cat" the meaning to the word will activate in his mind. In this case, a picture of a cat or a primitive cat movie will activate in the robot's mind. In the second item, a picture of a lemon is sensed by the robot and the meaning to the picture activates in his mind. For the robot, the sound "lemon" will be heard in his mind. Also the picture of a lemon causes the robot's saliva to secrete from his mouth. This is because sour and the sound "lemon" has strong association with a picture of a lemon.



Sentences can also represent very complex instructions. The sentence, "write a book(V5)", encapsulate the entire instructions to write a book. The sentence, "write a paragraph(V4)", encapsulates the entire instructions to write a paragraph. As a reminder, writing a book(V5) utilizes (V4) repeatedly and V4 uses V3 repeatedly and V3 uses V2 repeatedly. The English language allows the robot's brain to recursively encapsulate groups of instructions to a fixed media, which is a word or a sentence/s and allow knowledge to form complex structures. when the robot is making a decision and the sentence: "write a book in your spare times", activates in his mind, the sentence encapsulate all the complex instructions to write a book. That one sentence represents all the knowledge the robot needs to write a book. This sentence is also known as an internal instruction, given by the robot himself to take action.



Self-Awareness (the robot's conscious)

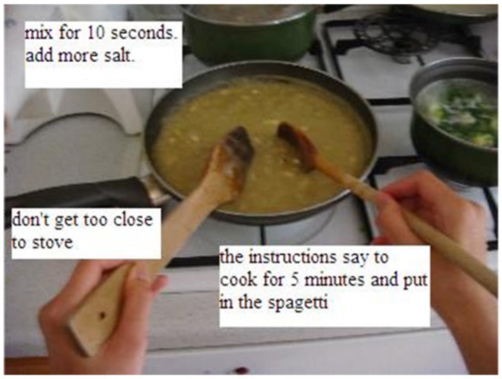
"The term, human conscious (n), used in this book means: a human being who is self-awared, alive, and thinking".

The robot's conscious (or the robot's mind) is the imaginary voice that provides instructions to take action; it can: do human tasks, do multiple simultaneous tasks, solve interruptions of tasks, solve conflicts of tasks, provide knowledge about an object/s, give meaning to language, generate common sense knowledge, solve problems, learn information, answer questions, follow commands given, search memory for information, or take any human action.

Examples of the robot's conscious (aka activated thoughts)
The three pictures below show different types of robots: a robot driver, a robot soldier, and a robot cook. The robot's conscious is the imaginary voice in his head that: provide information, think, give instructions, recall information, make decisions, solve problems, predict the future,

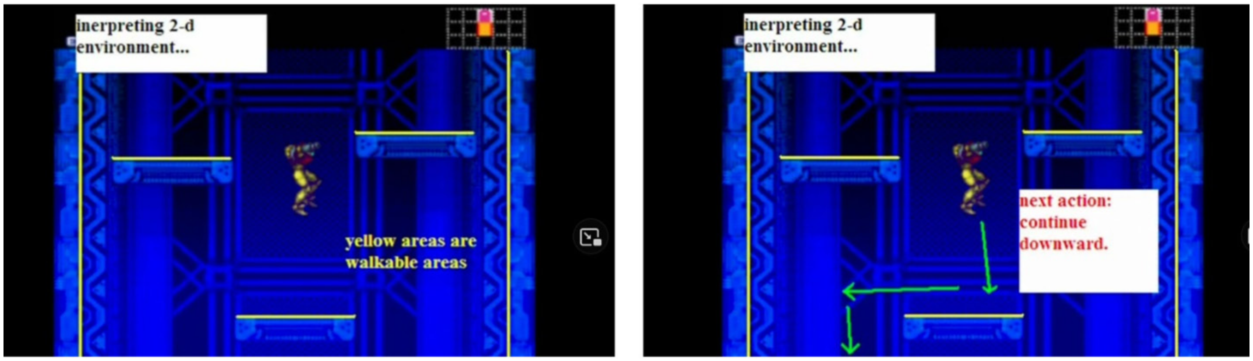
generate future steps to reach a goal and so forth. The robot's conscious is usually represented by internal sentences (sound data) or simple visual data.

The two pictures below show different types of robots: a robot soldier, and a robot cook. The robot's conscious is the imaginary voice in its head that: do recursive tasks, manage recursive tasks, follow rules, provide common sense knowledge, think, do linear logical reasoning, recall information, make decisions, solve problems, predict the future, generate future steps to reach a goal and so forth. The robot's conscious is usually represented by internal sentences (sound data) or simple visual data. The "voices" in the robot's mind help guide it to take intelligent action in a dynamic and changing environment.



In order to understand this concept let me visually depict what the conscious looks like in the mind of the robot. Hypothetically, let's use videogame POV to demonstrate the point of view from the robot's 5 senses. These visual images, flow diagrams, and text that pop up in the robot's mind serve as the linear instructions to think and act in a complex and changing environment.
The data structure to Human Level Artificial Intelligence submitted to the patent office in 2006 (priority) was my attempt to patent a robot that can think and act in this manner.

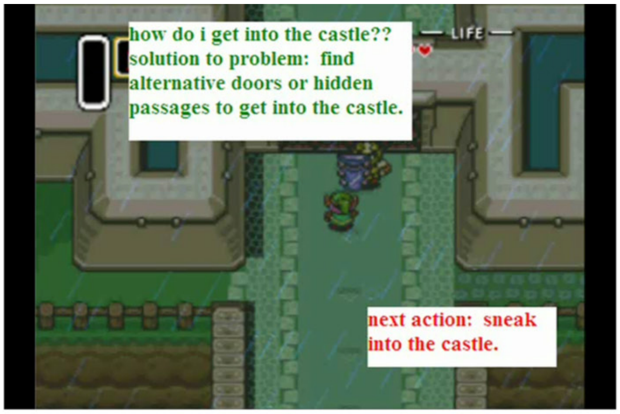
The text that pops up on the screen are the robot's internal thoughts



Here is a scene where the robot is interpreting the 2-d environment and navigating in an unknown environment. ▲

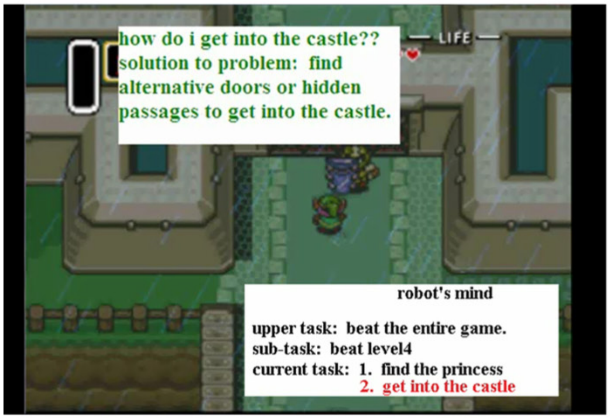
upper task: pass the entire game.
current task: roam in obvious places and discover missions. the game will guide you.

Here is an example of the robot doing recursive tasks. ▲



how do i get into the castle??
solution to problem: find alternative doors or hidden passages to get into the castle.

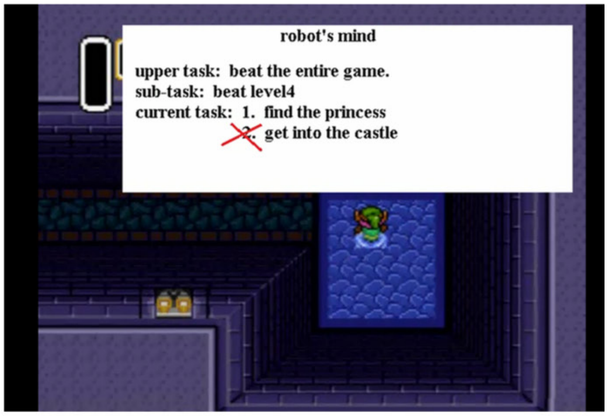
next action: sneak into the castle.



how do i get into the castle??
solution to problem: find alternative doors or hidden passages to get into the castle.

robot's mind

upper task: beat the entire game.
sub-task: beat level4
current task: 1. find the princess
2. get into the castle

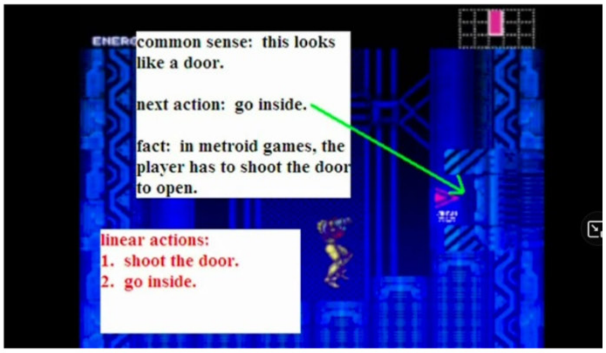


robot's mind

upper task: beat the entire game.
sub-task: beat level4
current task: 1. find the princess
~~2.~~ get into the castle

These are examples of the robot generating common sense knowledge and taking linear action based on its recursive goals. ▼

interpreting the environment..

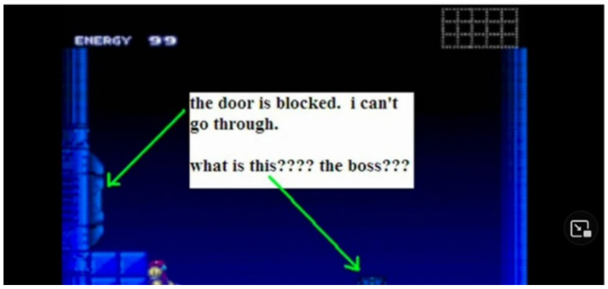


common sense: this looks like a door.

next action: go inside.

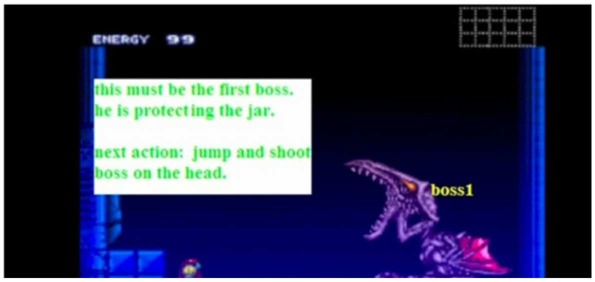
fact: in metroid games, the player has to shoot the door to open.

linear actions:
1. shoot the door.
2. go inside.



the door is blocked. i can't go through.

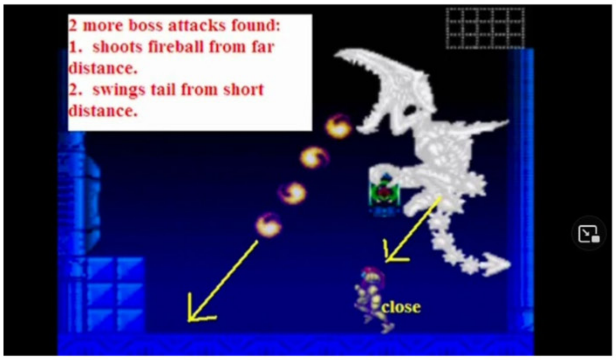
what is this???? the boss???

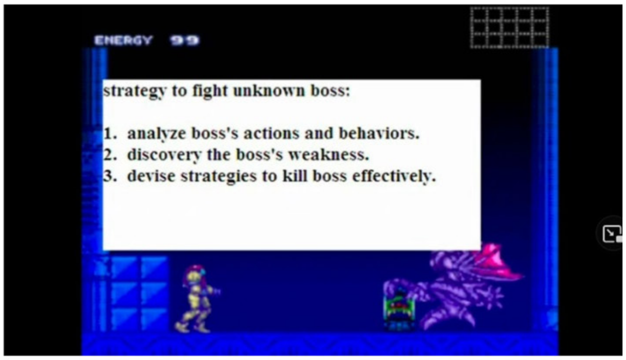


this must be the first boss. he is protecting the jar.

next action: jump and shoot boss on the head.

Observation through trial and error and discovering new facts.


2 more boss attacks found:
1. shoots fireball from far distance.
2. swings tail from short distance.



strategy to fight unknown boss:

1. analyze boss's actions and behaviors.
2. discovery the boss's weakness.
3. devise strategies to kill boss effectively.

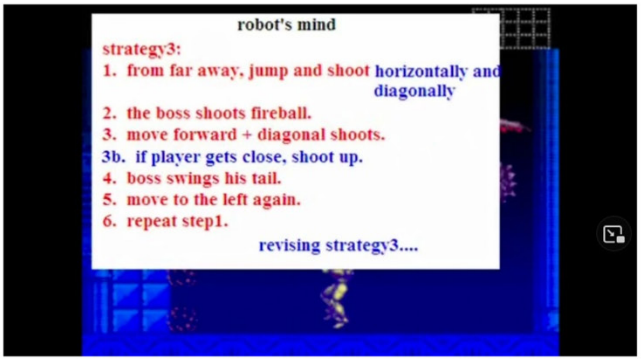
Here is a scene where the robot is generating a complex strategy (called strategy3), based on new ideas, learned strategies from memory, and through trial and error.

strategy3:
1. from far away, jump and shoot horizontally.
2. the boss shoots fireball.
3. move forward + diagonal shoots.
4. boss swings his tail.
5. move to the left again.
6. repeat step1.

use strategy3

And using stretegy3 as an encapsulated action.

Even more complex is revising strategy3 based on trial and error.


robot's mind

strategy3:
1. from far away, jump and shoot horizontally and diagonally
2. the boss shoots fireball.
3. move forward + diagonal shoots.
3b. if player gets close, shoot up.
4. boss swings his tail.
5. move to the left again.
6. repeat step1.

revising strategy3....

Over 400 hypothetical demo videos are posted online on Human-Level AI doing various tasks (especially playing videogames).

PictureE

Referring to PictureE, every frame of the robot's 5 senses are stored in memory. These pathways store the linear data that the robot goes through to do a certain task. There are certain goals and rules that are automatically done by the robot without given the instructions. For example, the robot knows it has to stay a certain distance from the stove or the current task is to cook spagetti. These tasks are implied in the pathways and doesn't require the robot's conscious to remind the robot all the different rules and goals it needs to do. Only the most important facts/rules are activated by the conscious. The human brain is very primitive and it can only focus on several things at any given moment in time. The pathways guide the robot to do things automatically, while the robot's conscious gives the most important facts/rules during specific times.

Teachers in school teach the robot various skills, like mathematics, science, decision making, learning information, social skills, generating common sense knowledge, logical inferencing, and physical education.

The most important thing teachers can teach the robot is the ability to "teach itself" to learn, practice, and do tasks. For example if the robot doesn't know how to drive a car, it can teach itself how to drive. The robot will seek knowledge from driving school or books to learn and practice. Practicing is an integral part of driving a car because it helps the robot refine its skill and become an expert.

Once the driving knowledge is stored in memory, the robot's brain can use decision making pathways to "use" the driving knowledge to drive a car.

As stated above, the robot's conscious manage tasks for the robot. However, there are many other things the robot's conscious can do, such as:

1. learning a task (learning a new skill like driving a car or cooking a lobster dish)
2. practicing a task.
3. usage of a task/s
4. modification of a task.

To illustrate my point I will show how my robot learns to drive a car. Keep in mind humans learn to drive a car in a completely different manner from an autonomous car. First of all, we don't have navigation systems in our brains, nor machine learning or GPS systems. Pay close attention to the way the robot learns, practices, and uses a task.

The knowledge to drive a car using human intelligence.

A skill like driving a car is thousands of times more complex than playing chess. There are more rules, more goals, and safety rules to follow. How does a human learn to drive? We learn to drive from both, driving teachers and knowledge in books.

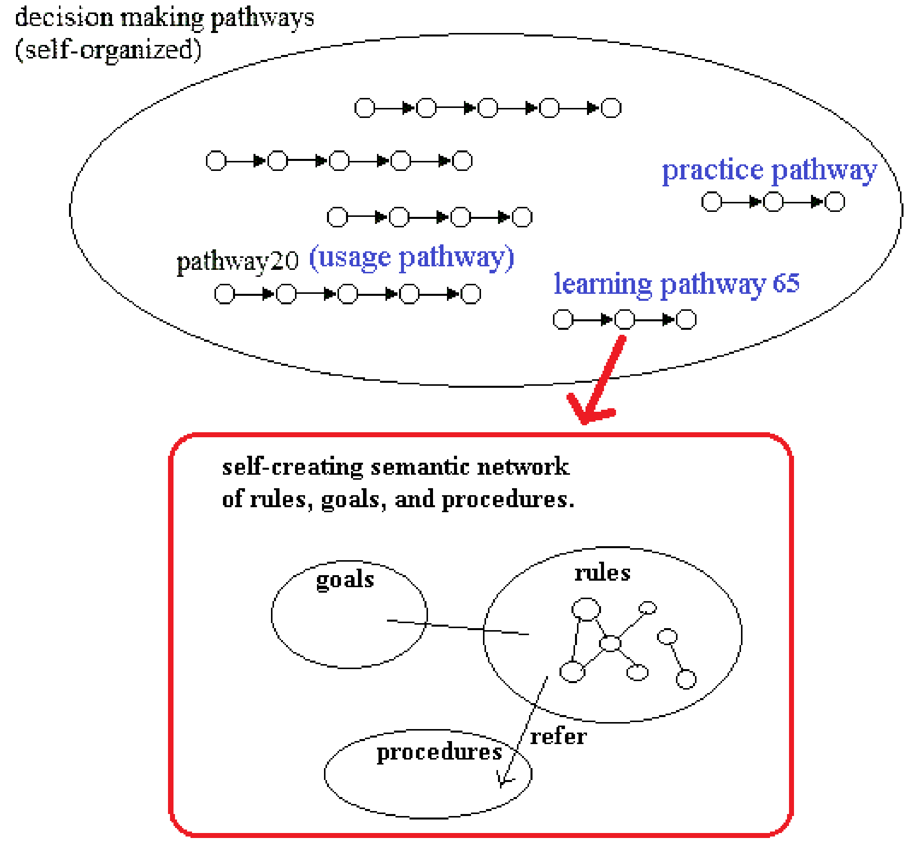


FIG. 45A

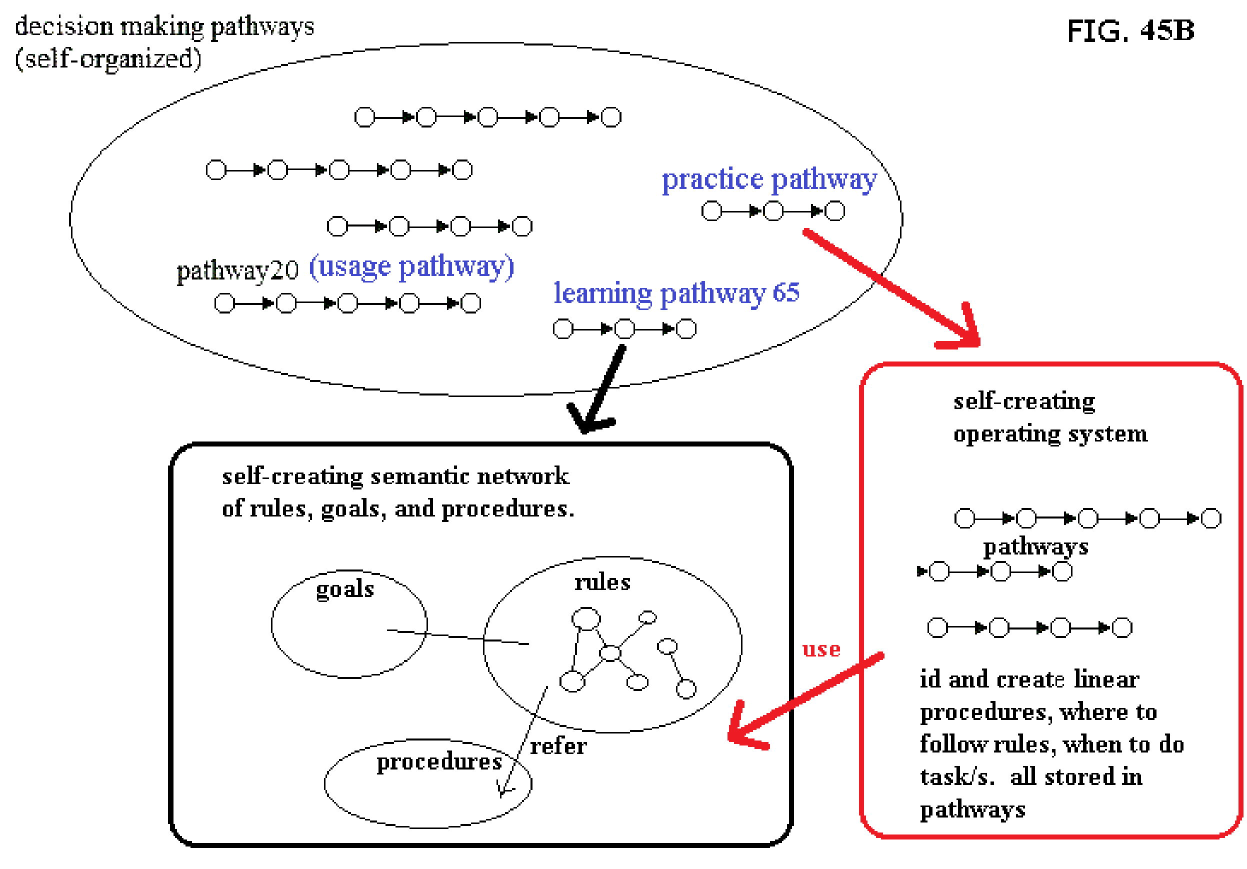


FIG. 45B

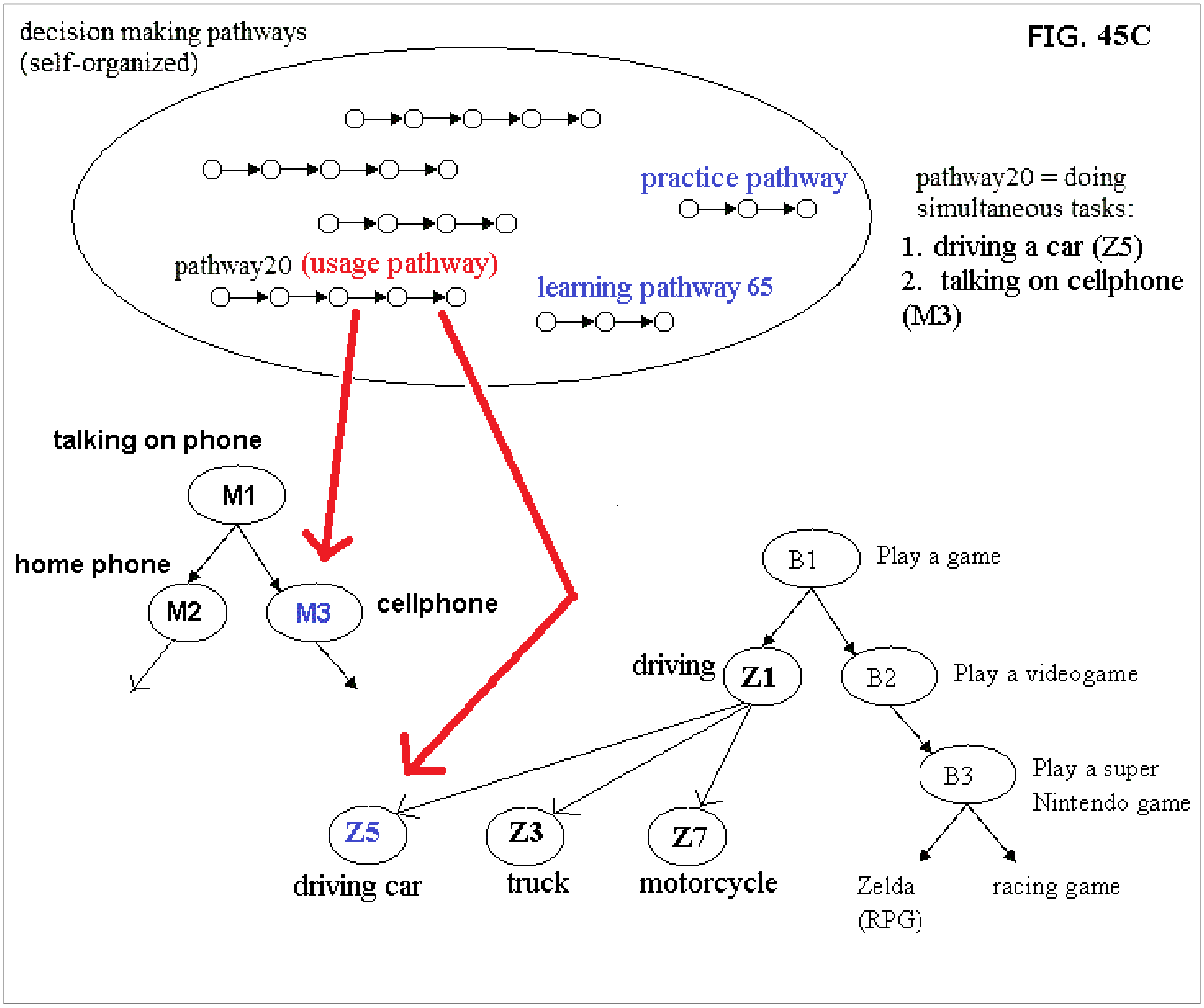
For the robot, he will use the learning a skill pathway to learn the goals and rules of driving. The robot will use logic and common sense to seek out the most important information in books or lectures. The sentences from driving books self-organize in a self-creating semantic network (FIG. 45A). Information will be stored optimally and visually. The goals are stored here and the rules are stored there.

Since the rules of driving is plenty, in the rules container are even more organized structures to store operational functions, recursive rules, hierarchical rules, solving conflicting if-then statements, and nested-if-then statements.

The learning pathways store optimal search functions to look for data quickly in this driving semantic network. These search functions are also self-created.

Next, the practice pathway is used to create and store linear instructions in driving pathways (FIG. 45B). As usual, the robot's mind will use common sense knowledge to create and store instructions in pathways. After months of driving and trial and error, the robot creates optimal driving pathways to drive a car, perfectly.

The learning pathways and the practice pathways create the pathways to drive a car in the robot's brain. Once the data is there, the robot can use another decision making pathway to "use" that knowledge (driving task). FIG. 45C is a diagram depicting a decision making pathway20 to use 2 different tasks (driving a car and talking on the phone). This decision making pathway20 is doing 2 simultaneous tasks at the same time.

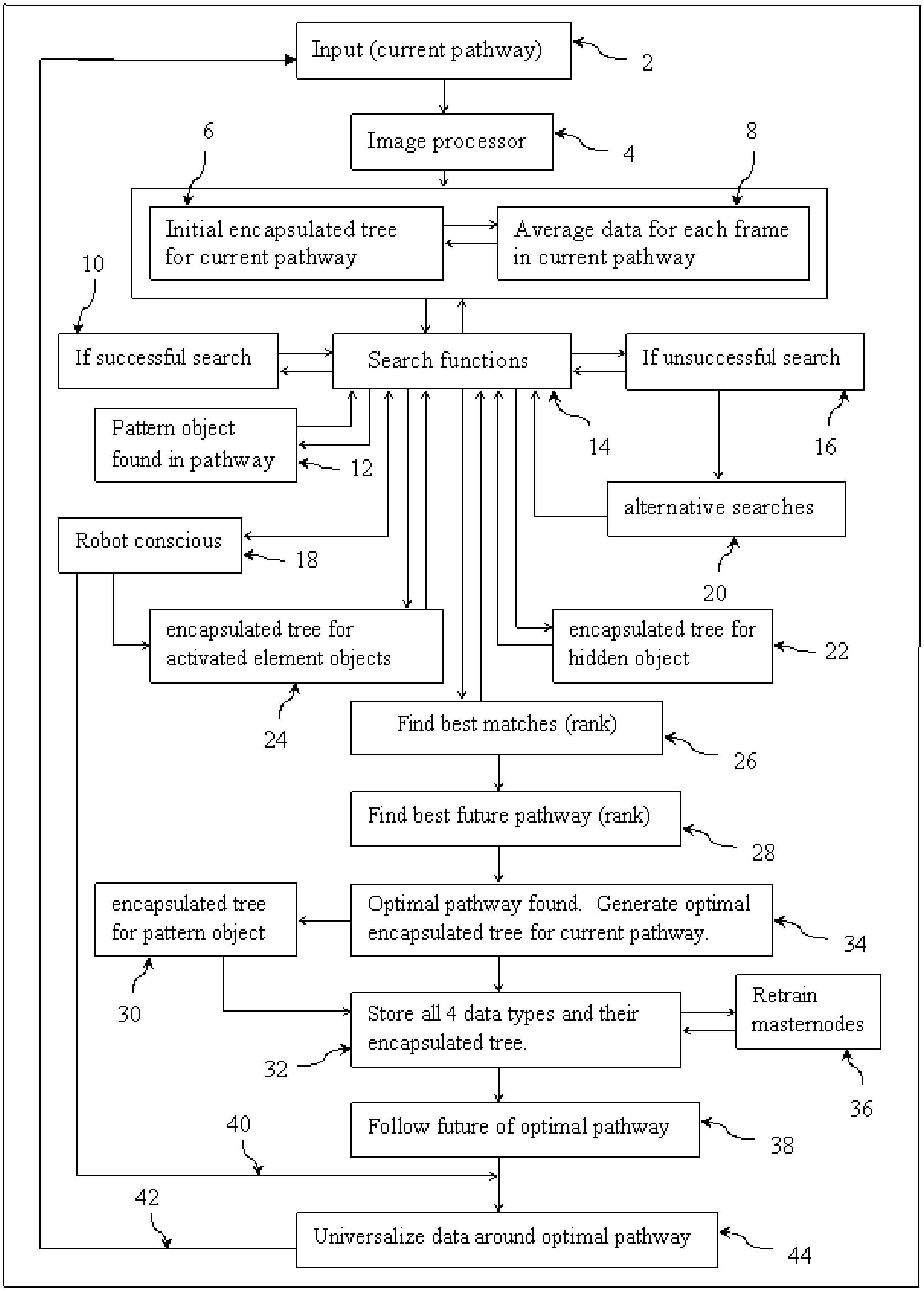


As the reader can see I'm not using any AI methods from autonomous cars. My robot learns information to drive from reading books or watching lectures. Humans don't have a navigation system, or machine learning, or GPS in their brains. They drive a car based on human intelligence and not by data structures from autonomous cars.

Chapter 1

Overall AI program

FIG. 1



Referring to FIG. 1, the present invention is a method of creating human artificial intelligence in machines and computer based software applications, the method comprising:

an artificial intelligent computer program repeats itself in a single for-loop to

receive information, calculate an optimal pathway from memory, and taking action; a 3-D storage area to store all data received by said artificial intelligent program; and a long-term memory used by said artificial intelligent program.

Said an AI program repeats itself in a single for-loop to receive information from the environment, calculating an optimal pathway from memory, and taking action. The steps in the for-loop comprises:

1. Receive input from the environment based on the 5 senses called the current pathway (block 2).

2. Use the image processor to dissect the current pathway into sections called partial data (also known as normalized visual objects). For visual objects, dissect data using 6 dissection functions: dissect image layers using previous optimal encapsulated trees, dissect image layers that are moving, dissect image layers that are partially moving, dissect image layers by calculating the 3-dimensional shape of all image layers in the movie sequence, dissect image layers using recursive color regions, and dissect image layers based on associated rules (block 4).

3. Generate an initial encapsulated tree for the current pathway and prepare visual object variations to be searched (block 6).

Average all data in initial encapsulated tree for the current pathway and determine the existence state of visual objects from sequential frames (block 8).

4. Execute two search functions to look for best pathway matches (block 14).

The first search function uses search points to match a visual object to a memory object. It uses breadth-first search because it searches for visual objects from the top-down and searches for all child visual objects before moving on to the next level.

The second search function uses guess points to match a memory object to a visual object. It uses depth-first search to find matches. From a visual object match in memory the search function will travel on the strongest-closest memory encapsulated connections to find possible memory objects. These memory objects will be used to match with possible visual objects in the initial encapsulated tree. This search function works backwards from the first search function.

The first search function will output general search areas for the second search function to search in. If the second search function deviates too far from the general search areas, the second search function will stop, backtrack and wait for more general search areas from the first search function.

The main purpose of the search function is to search for normalized visual objects separately and slowly converge on the current pathway (the current pathway is the root node in the initial encapsulated tree). All visual objects in the initial encapsulated tree must be matched. Search points and guess points call each other recursively so that top-levels of normalized visual objects will eventually be searched as well as bottom-levels of normalized visual objects.

5. Generate encapsulated trees for each new object created during runtime and include it in the initial encapsulated tree.

If visual object/s create a hidden object, then generate encapsulated tree for said hidden object. Allocate search points in memory closest to the visual objects that created the hidden object (block 22).

If visual object/s activates element objects (or learned object) then generate encapsulated tree for said activated element objects. Search in memory closest to the visual object/s that activated the element object (block 24).

If pathways in memory contain patterns determine the desirability of pathway (block 12).

6. If matches are successful or within a success threshold, modify initial encapsulated tree by increasing the powerpoints and priority percent of visual object/s involved in successful search (block 10).

If matches are not found or difficult to find, try a new alternative visual object search and modify initial encapsulated tree by decreasing the powerpoints and priority percent of visual object/s involved in unsuccessful search. If alternative visual object search is a better match than the original visual object match modify initial encapsulated tree by deleting the original visual object/s and replacing it with said alternative visual object (block 16 and 20).

7. Objects recognized by the AI program are called target objects and element objects are objects in memory that have strong association to the target object. The AI program will collect all element objects from all target objects and determine which element objects to activate. All element objects will compete with one another to be activated and the strongest element object/s will be activated. These activated element objects will be in the form of words, sentences, images, or instructions to guide the AI program to do one of the following: provide meaning to language, solve problems, plan tasks, solve interruption of tasks, predict the future, think, or analyze a situation. The activated element object/s is also known as the robot's conscious (block 18 and pointer 40).

8. Rank all best pathway matches in memory and determine their best future pathways. A decreasing factorial is multiplied to each frame closest to the current state (block 26 and block 28).

9. Based on best pathway matches and best future pathways calculate an optimal pathway and generate an optimal encapsulated tree for the current pathway. All 5 sense objects, hidden objects, and activated element objects (learned objects) will construct new encapsulated trees

based on the strongest permutation and combination groupings leading to the optimal pathway (block 34).

If the optimal pathway contains a pattern object, copy said pattern object to the current pathway and generate said pattern object's encapsulated tree and include it in the optimal encapsulated tree (block 30).

10. Store the current pathway and the optimal encapsulated tree (which contains 4 data types) in the optimal pathway (block 32).

Rank all objects and all of their encapsulated trees from the current pathway based on priority and locate their respective masternode to change and modify multiple copies of each object in memory (block 36).

11. Follow the future pathway of the optimal pathway (block 38).

12. Universalize data and find patterns in and around the optimal pathway. Bring data closer to one another and form object floaters. Find and compare similar pathways for any patterns. Group similar pathways together if patterns are found (block 44).

13. Repeat for-loop from the beginning (pointer 42)

The basic idea behind the AI program is to predict the future based on pathways in memory. The AI program will receive input from the environment based on 5 sense data called the current pathway. The image processor will break up the current pathway into pieces called partial data. The image processor also generates an initial encapsulated tree for the current pathway. Each partial data will be searched individually and all search points will communicate with each other on search results. Each search point will find better and better matches and converge on the current pathway until an exact pathway match is found or the entire network is searched. During the search process, visual objects will activate element objects (learned objects) or create hidden objects. Each new object created by the visual object/s will generate their respective encapsulated tree and included in the initial encapsulated tree. The optimal pathway is based on two criteria: the best pathway match and the best future pathway. After the search function is over and the AI program found the optimal pathway, the AI program will generate an optimal encapsulated tree for the current pathway. All 5 sense objects, all hidden objects, all activated element objects (or learned objects) and all pattern objects will recreate (or modify) encapsulated trees based on the strongest encapsulated permutation and combination groupings leading up to the optimal pathway. Next, the current pathway and its' optimal encapsulated tree will be stored near the optimal pathway. Then, the AI program follows the future instructions of the optimal pathway. Next, it will self-organize all data in and around the optimal pathway, compare similar pathways for any patterns and universal data around that area. Finally, the AI program repeats the function from the beginning.

The next couple of sections will emphasize on the robot's conscious and how the conscious is used to solve problems, plan tasks, predict the future and so forth. These sections are mentioned in detail in parent applications, but I will give a summary explanation so the readers can have a better understanding of how human intelligence is produced in a machine.

The human conscious works by the following steps:

1. The AI program receives 5 sense data from the environment.
2. Objects recognized by the AI program are called target objects and element objects are objects in memory that have strong association to the target object.
3. The AI program will collect all element objects from all target objects and determine which element objects to activate. Each target object might have multiple copies in memory so each target object will gather element objects from all or most same copies in memory.
4. All element objects will compete with one another to be activated and the strongest element object/s will be activated.
5. These activated element objects will be in the form of words, sentences, images, or instructions to guide the AI program to do one of the following: provide meaning to language, solve problems, plan tasks, solve interruption of tasks, predict the future, think, or analyze a situation.
6. The activated element object/s is also known as the robot's conscious.

Referring to FIG. 2A, when the AI program locates the three visual objects: A, B, C in memory it will run electricity through these nodes and all of its connections.

The mind has a fixed timeline. Only one element object can be activated at a given time in this timeline. This is how we prevent too much information from being processed and allow the AI to focus on the things that it senses from the 5 senses (FIG. 2B).

FIG.2A

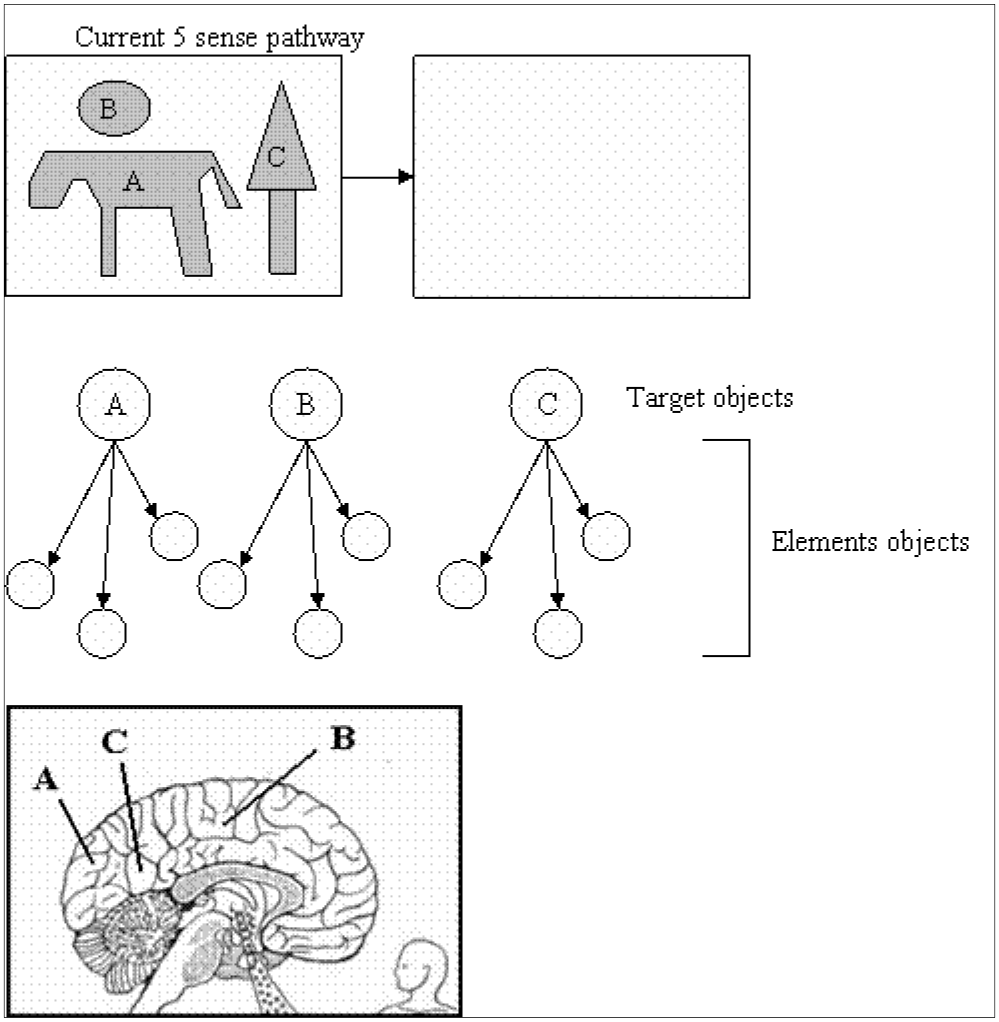
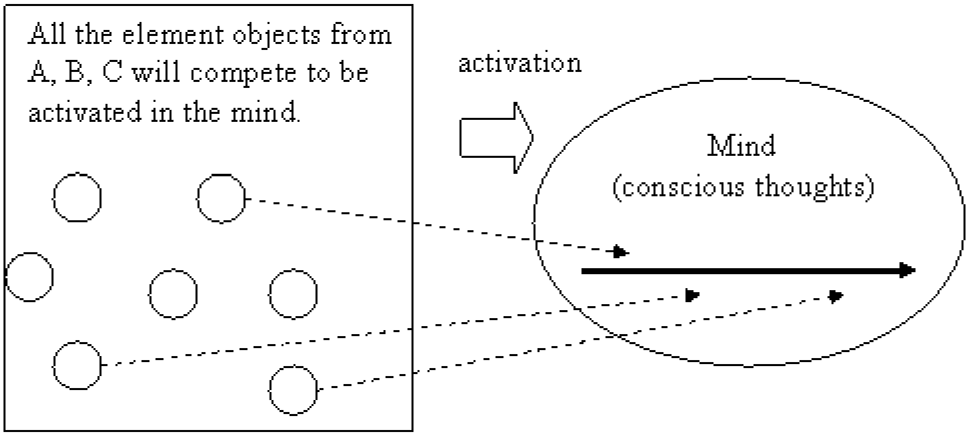


FIG. 2B


All the element objects from A, B, C will compete to be activated in the mind.

How does the robot learn information in terms of a bootstrapping manner, whereby old information is built on new information to form complex intelligence?

Intelligent pathways in memory (details)

First, the robot has to learn English grammar in kindergarten and first grade. He has to have a basic grammar understanding. Next, using English sentences, teachers teach the robot how to think, act, learn, and make decisions.

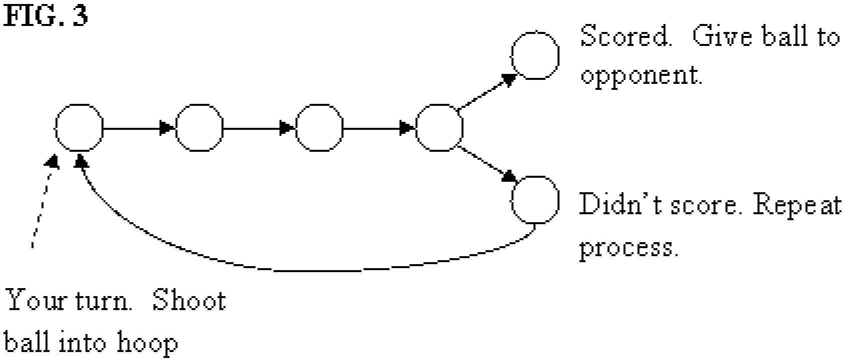
The main purpose of pathways in memory is to create any form of intelligence. The pathways should contain patterns that will control the AI program to take action in an intelligent way. Instructions in pathways control the robot's body functions such as moving its arms and legs or searching/modifying/processing/and storing information in memory or thinking consciously of intelligent ways to solve problems.

English sentences is a fixed object, referencing the beginning of a pathway/s. Sentences reference pathways all over the brain, regardless of where they are located. Thus, when the robot activates a sentence (activated thought), it searches the memory part to find the beginning of the sentences' respective pathway. This is how my robot's brain reference "blocks of domains", regardless of where they are located in memory or how complex these domains are.

The key to human intelligence

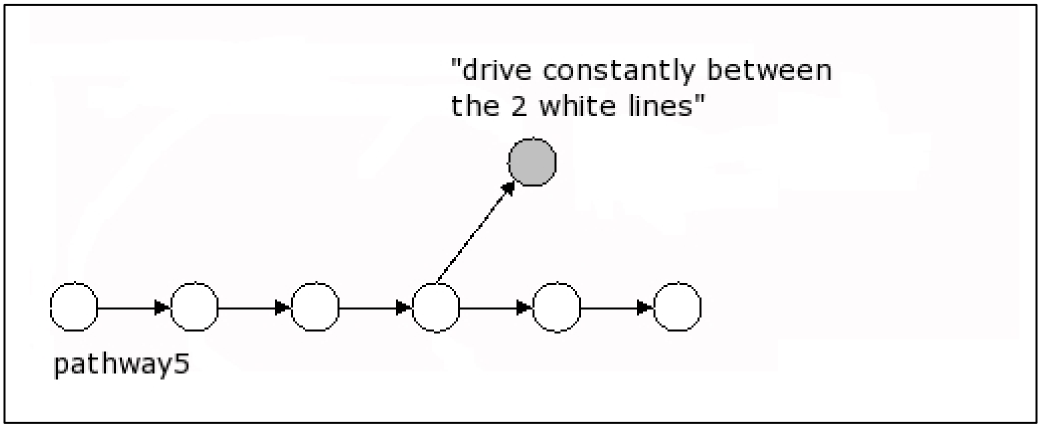
Language is the key to how these pathways are structured in an intelligent manner. Language will define the functions and behaviors of each pathway in the robot's brain. In FIG. 3, the sentence: "shoot the basketball until it goes into the hoop" is a for-loop that will loop itself based on a condition/s.

FIG. 3

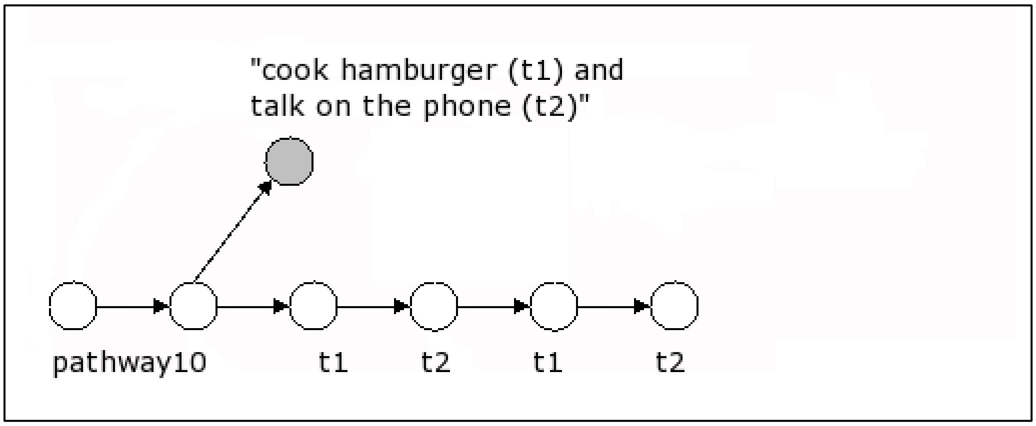


Notice that a sentence/s marks the beginning of a pathway or a sentence/s marks the beginning of a function or operation. A given pathway can have a plurality of sentences or possible sequences (almost like a smart tree/forest). Think of pathways as connected dendrites in a human brain that can form complex data structures.

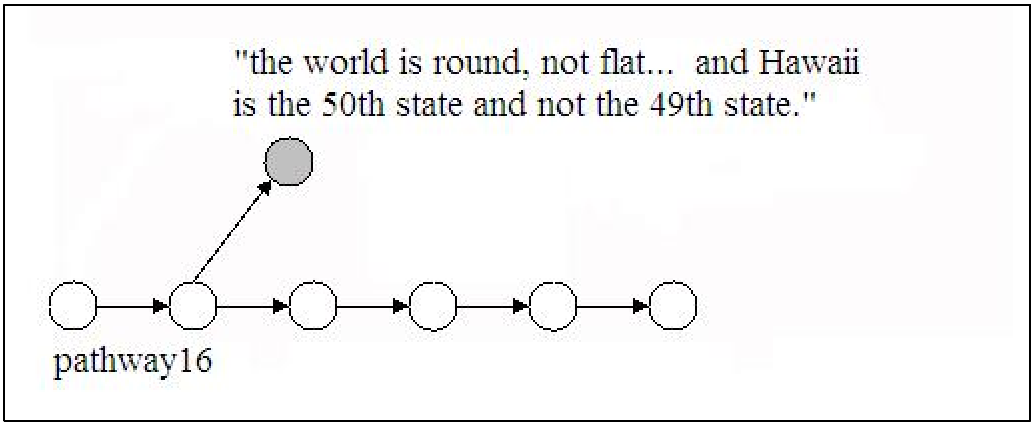
In this example, the sentence: "constantly drive between the 2 white lines" is a constant rule that tells the pathway to constantly follow this rule while driving.



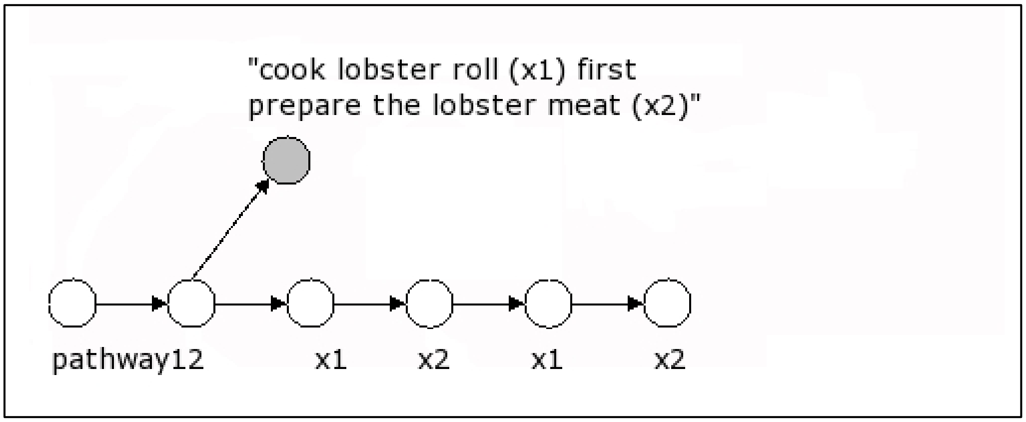
In this example, the sentence: "cook hamburger and talk on the phone simultaneously" manage 2 tasks simultaneously by switching between tasks until both tasks are completed.



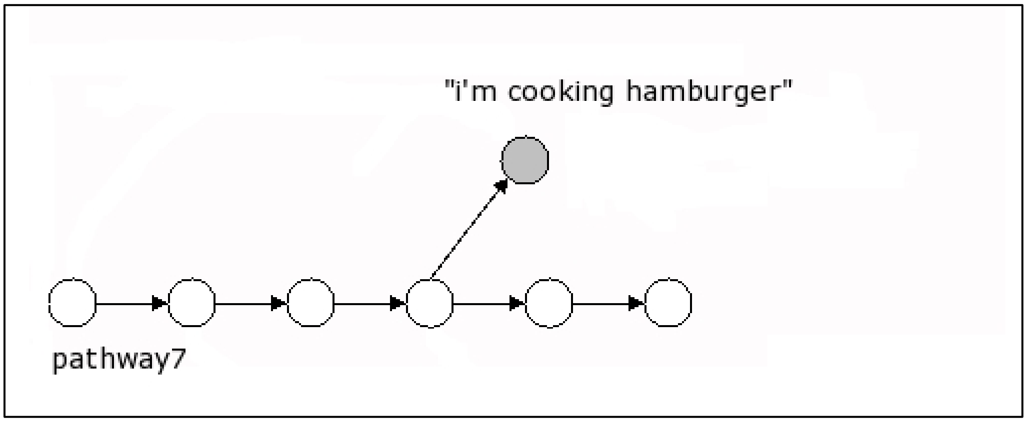
In this example, the sentence: "the world is round, not flat... and Hawaii is the 50th state and not the 49th state." is forcing the robot's brain to correct wrong information stored in memory. In this case, the correct data is created in memory (marked as correct) and the wrong data will have a forget function next to it (marked as wrong). Thus, the next time the robot retrieves the fact, the correct data will be present. This example shows pathways can do complex database functions.



In this example, the sentence: "cook lobster roll. First, prepare the lobster meat" manages recursive tasks. The 2 sentences basically manage recursive tasks. It knows it is currently making a lobster roll. At the same time, it also knows that the first step is to prepare the lobster meat. After finishing the first task the second task will pop up in the robot's brain.



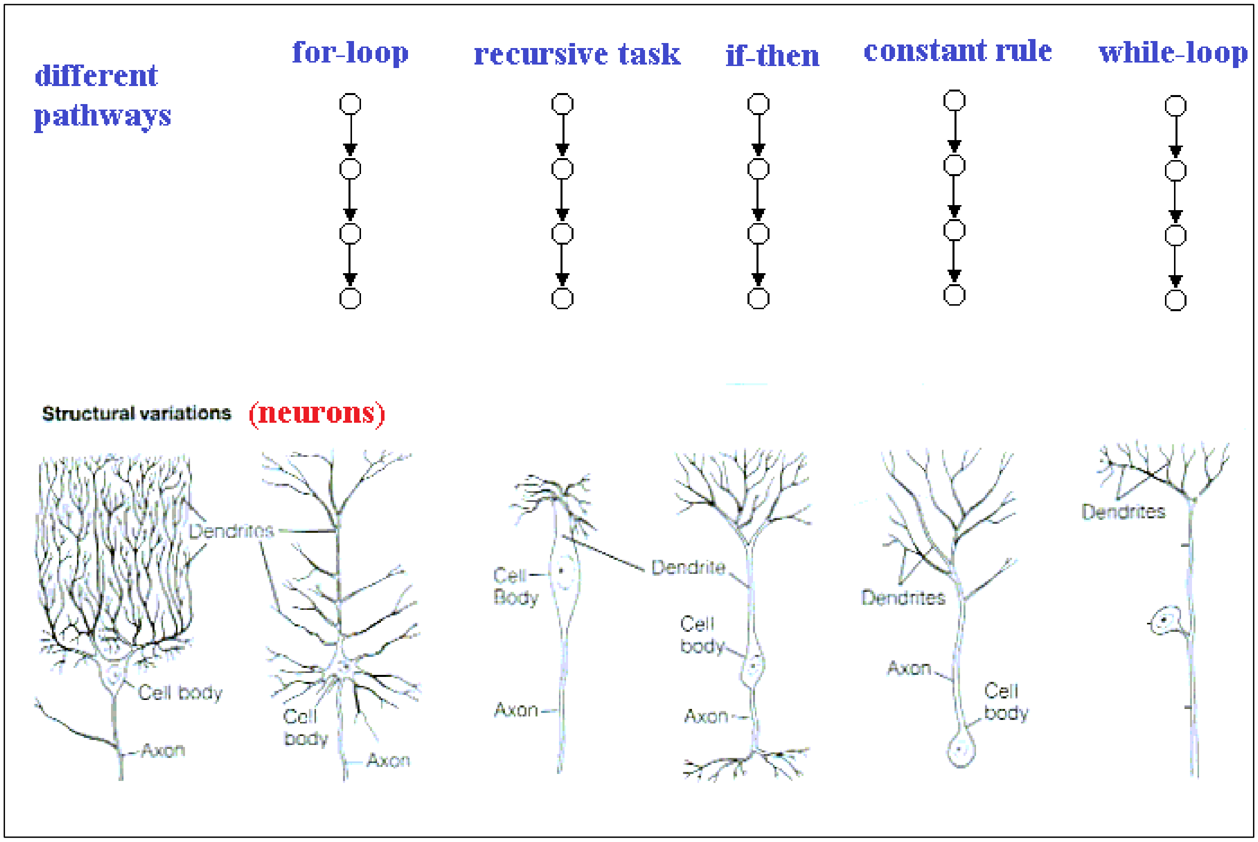
In this example, the sentence: "cook hamburger" is a constant task. The pathway will contain the beginning and ending of the task.



The pathways in memory can form for-loops, while-loops, if-then statements, and-statements, or-statements, assignment statements, sequence data, recursive functions, classes, procedures, random data, static data and all the different combinations. The pathways are also able to form any type of computer program, including: databases, expert systems, genetic programs, and AI programs. Simple computer programs like a word processor or complex computer programs like the internet can form in pathways. These sequence data can even form self-learning and self-accomplishing behavior to solve arbitrary problems.

The intelligent pathways can do anything that a state machine can do. Self-organization between similar intelligent pathways is the tool that defines the state machines.

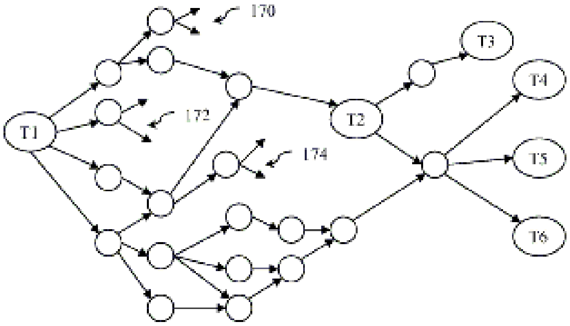
The diagram below shows various elemental neuron structures. These elemental neurons are exactly the same as different types of pathways. In computer science they teach you that operational functions like for-loops, if-then statements, while-loops, OOP, recursive functions, or-statements and so on are the elemental building blocks to any computer program. Neurons in the human brain work exactly the same way. These elemental neurons build on top of each other (based on learning) and form complex structures in the human brain to think and act.



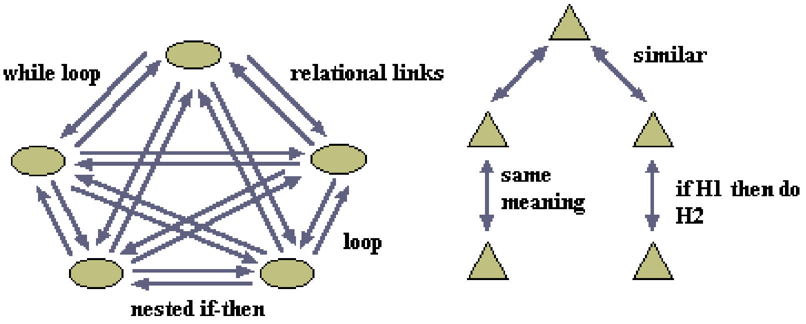
The pathways store possibilities of life experiences so self-created decision trees are automatically created in the robot's brain. Language in pathways also form self-created semantic networks, database systems, operating systems to manage multiple tasks, or logical network systems.

English sentences in pathways generate:

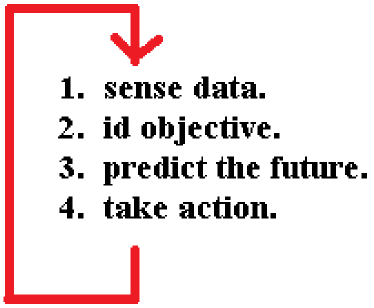
self-created decision trees



self-created semantic networks and logic networks



self-created operating system to manage multiple tasks.



1. sense data.
2. id objective.
3. predict the future.
4. take action.

repeat every second

1. goals:
2. rules:
3. planning:
4. identity and limits

long-term memory

Details of the conscious

The robot's conscious comprises a computer program that is created by the intelligent pathways in memory. As the robot's brain selects an optimal pathway, this intelligent pathway will generate a computer program to do things (insert, delete or modify information that goes into the conscious). As the robot's brain selects optimal pathways over a period of time, the computer program is generated and functions to do one or multiple tasks.

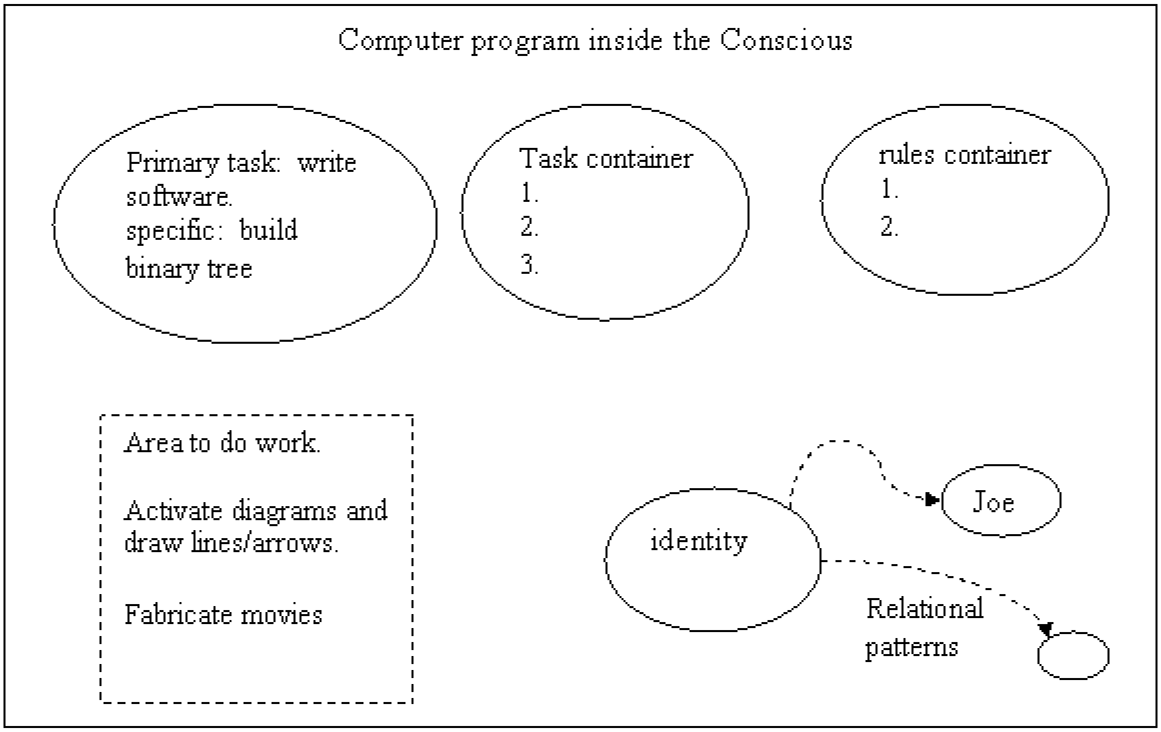
If the current situation is to drive a car, the conscious will have a computer program to drive a car, if the current situation is to fly an airplane, the conscious will have a computer program to fly an airplane, if the current situation is to do math equations, the conscious will have a computer program to do math equations. While the robot is selecting intelligent pathways for a given situation, the intelligent pathways are creating the computer program inside the conscious to take action.

The computer inside the conscious serves as an operating system and it can manage multiple simultaneous tasks. For example, in terms of driving a car, the robot has to search and identify traffic lights, road signals and pedestrians. At the same time, he has to drive between the two white lines on the road and avoid other cars/pedestrians. In other cases, the robot has to plan routes in his head so that he knows what streets to drive in. The rules and tasks of driving a car are all managed by the computer program inside the conscious. Later on, I will give detailed examples of how the conscious works.

FIG. 3 is a diagram illustrating the robot's conscious. The conscious is like an empty room and you can do anything in it. You can extract a map of the city and draw lines/arrow on it. Planning routes require that the robot extract a map image from memory and id the current location, draw lines to the destination location, and list the streets it has to travel in linear order. The robot will drive the car based on the list of streets it has created in its mind.

There are basically two containers in the conscious: tasks and rules/facts. Tasks are things that the robot has to physically or mentally do. Picking up a book is a physical task and searching for relevant data in memory is a mental task. The rules are data that is extracted from memory that are needed for a task. Solving a problem require facts about a situation. Rules are needed to be followed while doing a task. For example, when driving on the road, there are rules that you have to follow. When playing baseball, there are rules to follow for each player.

FIG. 3



The intelligent pathways selected by the robot during a situation will define what tasks to do at certain times and what probable rules to follow at certain times. To tell you the truth, the rules are mostly made up of if-then statements. If the robot recognizes this object then it will take this action. However, there is no definite law that says rules have to be if-then statements. They can be any discrete math function, like when-statements, for-loops, complex recursions and so forth.

Another feature of the conscious is the identity container. The current pathway (the current encountered situation) contains 4 types of data types: 5 sense data, hidden objects, activated element objects and pattern objects. These 4 data types will define objects, events, actions, object interactions, object relationships and so forth. Basically, they create a self-defining and self-adapting semantic network based on the data in the current pathway. There are some object interactions that are so complex that even semantic networks can't define. Things like object property and complex object interactions are things semantic networks can't represent.

A self-defining semantic network is a network that doesn't need computer scientists to write the functions. The network is created by itself from data learned from the environment (the current pathway).

In the identity container are very important things the robot believes, such as its primary goals in life, what are the rules that it must follow and what does it like and dislike. This container is important because the robot is "self-aware" and constantly knows certain things and doesn't need to search for these things in memory. When the robot encounters objects/events/actions from the environment it extracts the strongest facts in memory related to these objects. Knowledge is like a water valve. When the robot's task is to play baseball, the most important knowledge about baseball pours in from memory into the conscious. When the robot encounters a friend, the most important knowledge about this friend pours into the conscious. In other cases, the robot can think of a bird and the bird's knowledge will pour into the conscious.

Each object/event/action in memory is like a forest of network that is neatly organized.

The identity container is created because the robot is constantly thinking about itself. When it decides on items, the robot might say: "I select the blue box", the word "I" automatically extracts info on the robot. When the robot does a task, he knows that he is doing the task. If someone else in the environment is talking to or about the robot, then the robot will extract info about itself.

The identifying of objects/events/actions and their interactions is based on the activated element objects from the robot. The conscious thoughts of the robot identify objects/events/actions. Conscious thoughts will be in the form of sentences and finger pointing to assign words to 5 sense data. Other peoples' spoken sentences can also define objects/events and actions in the environment.

By identifying objects/events/actions in the environment and establishing relational links, object properties and object interactions, the data is configured in an optimal way to be stored in memory.

The identity container is there because it is easier for the robot to make decisions and to do tasks based on what the robot wants. Things the robot likes and dislikes are there so that he can make decisions quickly. Instead of searching for the relevant data in memory during runtime, the knowledge is already there for the taking. Of course, the knowledge within the identity container is based on the current situation. When the robot has to make a decision, the knowledge will be based on what that decision is about. For example, if the robot was in the supermarket and he has to decide on groceries, the identity container might contain likes and dislikes in terms of shopping. In the task container, there might be intelligent pathways for decision making. In the rules container, the rules of selection from the task of decision making might be included.

There are two types of factors that change data in the robot's conscious: internal commands and external commands. For example, the intelligent pathways selected by the robot's brain can change the tasks in the task container. The robot wanted to go to Mcdonalds to eat lunch, but changes his mind when he saw the taco picture from Taco bell. The intelligent pathway decided to eat at Taco bell instead of Mcdonalds. This example is an internal command. The external command is based on people or things from the environment.

If the robot was playing baseball, the coach will tell the robot what position it has to play. The coach will tell the signal codes of the game and the batting lineup. The robot will have to first identify a command from the coach. The coach might give a 10 minute speech and commands might be vague. The robot might have to use logic to id the commands. Once the command is found, the robot will change all the data in the containers in the conscious in terms of its task, rules to follow and knowledge needed.

Using intelligent pathways to determine what knowledge to learn

When the robot is in school, he has the task of learning knowledge from teachers. He also has the task of following the teacher's every command. Unless that task is dangerous or unusual, the robot might ask why.

In some cases, the robot has the power to control wither to learn knowledge or not. If the robot doesn't want to learn, he can stop paying attention to the teacher or to pretend like he is paying attention.

He can pay attention to the lesson or he can ignore the lesson. The controlling of the degree of learning is done via intelligent pathways from memory. If the robot knows that he is in trouble with his grades, he can consciously decide to pay strong attention to knowledge from the teacher. Paying more attention means learning more, which leads to a better grade for the class.

The intelligent pathways in memory also determine fact from fiction. If the robot was taught that the world is flat and one day he read a magazine and the scientist says the Earth is actually round, will he believe the magazine or what society think is true at that time? Intelligent pathways of determining fact from fiction will be used and if the robot determines that the magazine is true, then he will put a reminder in the location of the false fact in memory and establish a pointer to the true fact.

On the other hand, if the robot determines that the scientist is lying and his facts are flawed, then he will ignore the information in the magazine and only store a temporary encounter of the magazine. The data from the magazine will be marked with a false fact and will not change the true fact that is already stored in memory.

This intelligent pathway to determine fact from fiction is based on teachers teaching the robot how to find out if the material read is true or false. The teachers have to teach steps to identify, analyze and determine if certain reading material is false or not. Teachers have to train the robot to determine truth from different medias. The robot might have to read a short essay and determine if the content is true or not. Or the robot has to watch a movie and determine if the content is true or not. Maybe there are some truths and some false information in the movie and the robot has to id as much true and false scenes. In other cases, the robot has to read an entire website and determine where the true and false statements are.

Learning to activate element objects based on a given situation (target object)

In the overall idea of the conscious described above, the brain recognizes target objects from the environment and activates element objects (conscious thoughts) from memory. There is a way to teach the robot to activate specific element objects in its conscious based on a target object/s.

A classroom is a perfect place to learn how to activate element objects based on a target object/s. A teacher will teach the class how to analyze a situation or object. After the situation is given the teacher will ask the students to comment on the situation and to share insights with other students.

The teacher and students will comment on the situation and their comments can be facts, logical facts, common knowledge, question-answers, point of views and so forth. Once the robot hears all the comments from students and the teacher about the situation, he knows what are important data. The situation is the target object and the comments and facts are the element objects. In the future, when the robot encounters the situation he will activate the comments from the students and teacher.

Grade school worksheets can be another method to learn to activate element objects based on a given target object/s. In the worksheet, there might be a picture and next to the picture is the instruction: "write down important facts about the picture". If the picture is a man, then important facts can be: his name, telephone number, occupation and age. If the picture is a snake, then important facts can be: a reptile, poisonous, dangerous creature, has venom in teeth, don't go near it and so forth. The next time that the robot encounters a snake, the facts from the worksheet activates. If the robot encounters a male, facts about his name, telephone number, occupation and age activates.

In some instances, the worksheet can be associated with recognizing a target object and activating element objects.

Learning to create a database system in memory

In current database systems there exist a frame for each customer. In the frames are slots such as name, age, gender, occupation and height. The human robot learns to create a database system in memory. The frames and functions of the database system are learned. In memory, data is stored in a hierarchical-associated manner. The intelligent pathways provide the search function for the data base system.

What about the frames/slots and relational links to other customers? How does the robot form these frame/slots? The answer is right in front of your eyes. When the robot reads a form about someone, the form lists important facts about the person. The form will list the name, address, phone number, occupation and so forth about a person. The form is optimal in that important facts, such as a person's name are located at the top and minor facts, such as their pet are located at the bottom. This numbering list determines the priority of which facts are important and which aren't.

If the robot encounters many forms and these forms are all similar to each other, whereby the name is at the top and minor facts are at the bottom, then the robot can create a template form in memory. Since a form is associated with a person and the robot encounters this association numerous times, then a template form can be created for all people the robot has met and will meet.

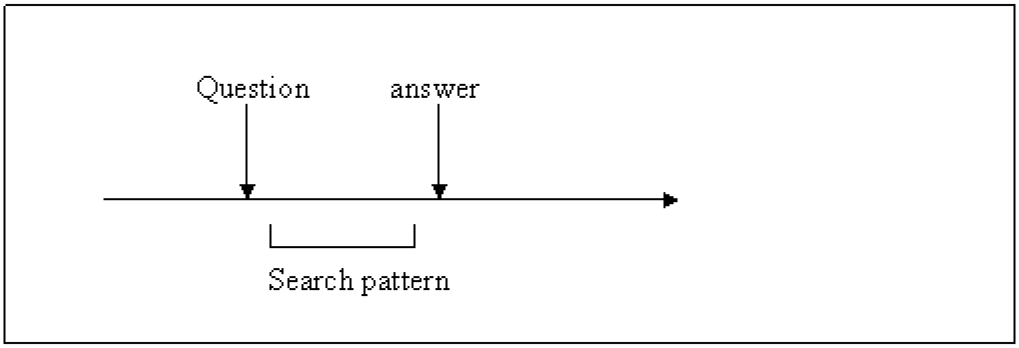
If the robot meets a girl, called Jessica, at a party that he has never seen before, he will store the visual images of Jessica in memory and create a template form for this person. When Jessica tells the robot her name, the name will be stored in the template form at the name slot. When Jessica tells the robot information about herself, the information will be stored in their respective slots in the template form.

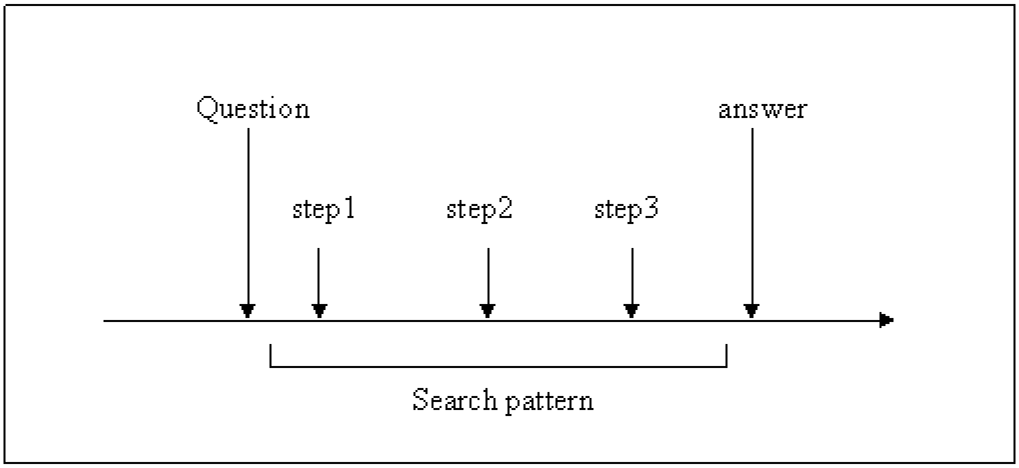
The template form is associated with Jessica at this point. If the robot encounters Jessica in the future, this template form will activate and further facts will be stored in this template form. Let's say that Jessica told the robot her occupation in the second meeting. The fact "Jessica is a retailer" is identified by the robot; and the robot logically assume her occupation is a retailer. The sentence "her occupation is a retailer" will be stored in the template form under the occupation slot. Why? Because the occupation in the template form has association with the sentence. The sentence: "her name is Jessica" will be stored in the name slot. The sentence: "her phone number is 555-5555" will be stored in the phone number slot. It's all about association. The different data types are compatible because of this type of network. The sentences are sound sentences and the template form is a visual diagram. These are two different data types, but the sound occupation is associated with the visual text occupation.

The search function for this database system is based on patterns. The teachers teach the robot using supervised learning. Here is the question and here is the answer. The robot, through self-organization, will find the patterns between the question and the answer. The pattern is the search function. If the question and the answer example is very complex, the teacher can teach the robot steps of logic in order to get from the question to the answer. The robot can then find out the patterns by comparing similar examples (FIG. 4).

The question is the start and the answer is the end. It doesn't have to be a question-answer format, it can be a command-accomplish command format. It can also be an input and desired output format.

FIG. 4





The steps are guides to make it easier for the robot to find the answer and to ultimately discover the search pattern. The diagram above shows that there are three steps that have to be done in order to find the answer. During the self-organization phase, when the robot's brain compares similar examples, it will find patterns between each step.

The search pattern in intelligent pathways is a combination of internal functions (searching in long term memory, searching in memory, determining distance, etc) and intelligent pathways. In other intelligent pathways, there might be search patterns already pre-existing and that can be transported to the current search pattern. For example, step1 might be an intelligent pathway that already exists in memory. Instead of finding the pattern to step1 the robot's brain can copy the search pattern from the intelligent pathway found into step1.

Chapter 2

Computer programs inside the conscious

The robot will sense data from the environment through its 5 senses: sight, sound, taste, touch and smell. Target objects are objects the robot recognizes from the environment. The brain of the robot will extract element objects that have strong association with their respective target objects. All element objects from all target objects will compete with one another to be activated in the mind. These activated thoughts are known as the conscious of the robot.

There are two types of conscious thoughts: open activation and hidden activation. The open activations are element objects that are presented to the conscious and the robot is aware of the data. On the other hand, hidden activations are element objects that are not presented to the conscious and the robot is unaware of the data. Extremely complex tasks in memory, such as problem solving, will require both open and hidden activation.

Think of the conscious as a "memory bubble" that takes in and modify data; and the modified data instruct the robot to take action (FIG. 5A - 5B). Inside the conscious is a computer program (or a series of interconnected computer programs) that manipulate data in the conscious. The intelligent pathways extracted from memory and used by the robot generate this computer program.

FIG. 5A

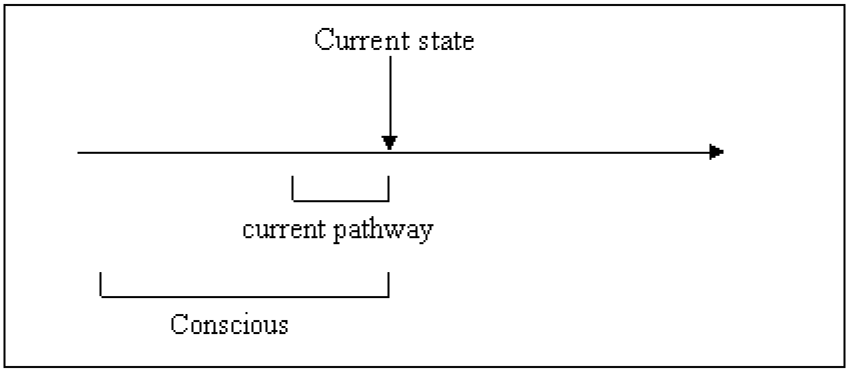
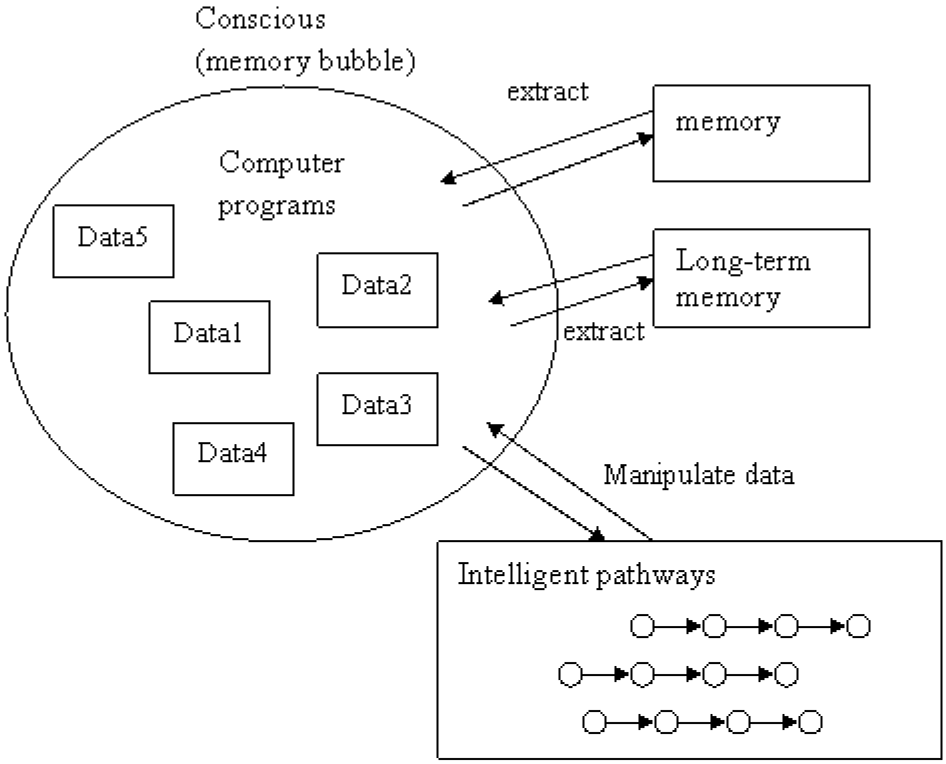


FIG. 5B



The intelligent pathways control the data in the conscious by adding, deleting and manipulating data. They also control the searching and extracting of data from memory. The instructions inside the intelligent pathways will have search functions that will extract relevant data, based on a situation, and put them into the conscious. Some data will be extracted from memory, while other data will be extracted from long-term memory.

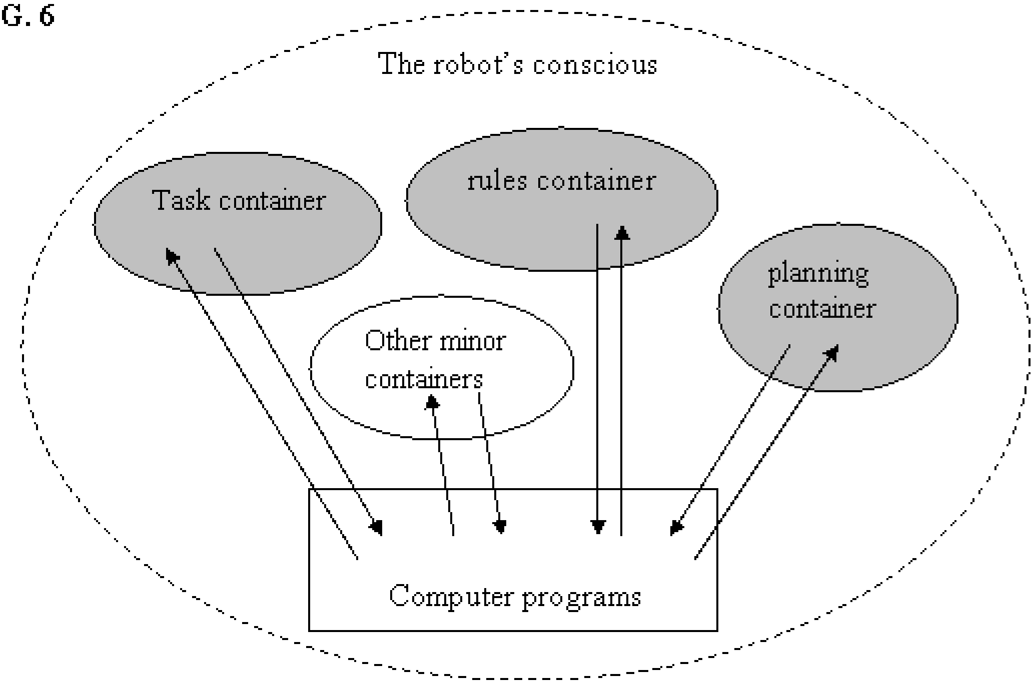
There are also other things the intelligent pathways in memory can do. They can control the comparing of data; or the analyzing of two or more data from the conscious. These intelligent pathways can even predict the future and generate a computer program to output future events of what might happen based on the current environment.

The conscious actually does many other things for the robot. Some of these things include: giving knowledge about an object, providing meaning to language, solving problems, answering questions, identifying grammar structure, following orders, planning tasks, solving interruptions of tasks, managing multiple tasks at that same time, following the rules of a game, coming up with imagination and so forth. The intelligent pathways in memory allow the conscious of the robot to do all these things (and more).

Important data in the conscious

The conscious does many different things for the robot. Four of the most important things the conscious does are: 1. manage tasks. 2. establish rules to follow, based on the tasks. 3. planning steps to achieve tasks. 4. know identity. Referring to FIG. 6, there are four containers the computer program in the conscious generated as a result of intelligent pathways: the task container, the rules container, the planning container and the identity container. All data from all four containers influence each other one way or another. For example, the rules will influence what tasks to follow/abort and the planning information will influence what rules to follow or what tasks to do.

FIG. 6



These containers are just temporary caches inside the conscious that was generated by intelligent pathways in memory. Based on the current environment, the robot selects an optimal pathway from memory and that optimal pathway has instructions to create containers so that groups of data could be manipulated and logical thoughts and actions can be had by the robot. The intelligent pathways create any type of computer program or discrete mathematical functions to manipulate data in the conscious -- a database system, an operating system to manage multiple threads, a word processor, an image processor, a search engine, or any software program.

Storing and retrieving data in memory

In current database systems, a computer programmer has to create the storage functions and the search functions. In a human robot, the brain is made in such a way that the storage and retrieval of data is based on learned knowledge. Teachers teach the robot how to store data in memory and patterns within these lessons will establish how to store and retrieve data in memory.

Data in memory are based on pathways – the robot learns knowledge by taking in movie sequences from the environment through its 5 senses: sight, sound, taste, touch and smell. These pathways can store static data, which are data in pathways that exists. For example, the robot is looking at a TV guide for this week. The TV guide for this week is one static data stored in memory. Last weeks TV guide or the weeks before are also individual static data. Pathways can also store linear data. For example, the linear steps to solving the ABC block problem is linear data – these pathways store the linear steps the robot has to follow in order to solve the ABC block problem.

The data in memory can also be in any 5 senses: sight, sound, taste, touch or smell; and each sense can be represented in different ways. A music song belongs to sound and an action movie belongs to sight and sound. All these 5 sense data can be either static data or linear data (or both).

Organization of data is done through hierarchical association. A network of data that have relations to each other, based on the robot's experiences, will knit themselves together. The more two objects are encountered by the robot; and the closer these objects are from one another, the more association they will have with each other.

Although data is based on hierarchical association, the real factor that organizes them in memory is intelligent pathways. Teachers will teach the robot how to store data in memory. They will teach the robot which are important data and which aren't. For example, if a teacher gives a 2 hour lecture on nanotechnology, the robot will identify the most important data from the lecture and store them in memory. These important data will have top priority, while minor or repeated data will have low priority. Another example would be if someone tells the robot a 2 hour story. The robot can summarize the story into 3 sentences. That 2 hour story can be represented by 3 sentences. The 2 hour story can also be represented by a simple fabricated movie.

Learning forms

Let's say that there is an object composite of a person Dave in the robot's memory. Dave is a good friend of the robot and they have been friends for many years. The robot will store a network of data concerning dave (which is called an object composite). How does the object composite of dave structure the data in a hierarchical manner? How do we know that the face of dave is more important than the leg of dave? How do we know that a name of a person is more important than their age? How do we know that important facts about dave should be prominent in the network?

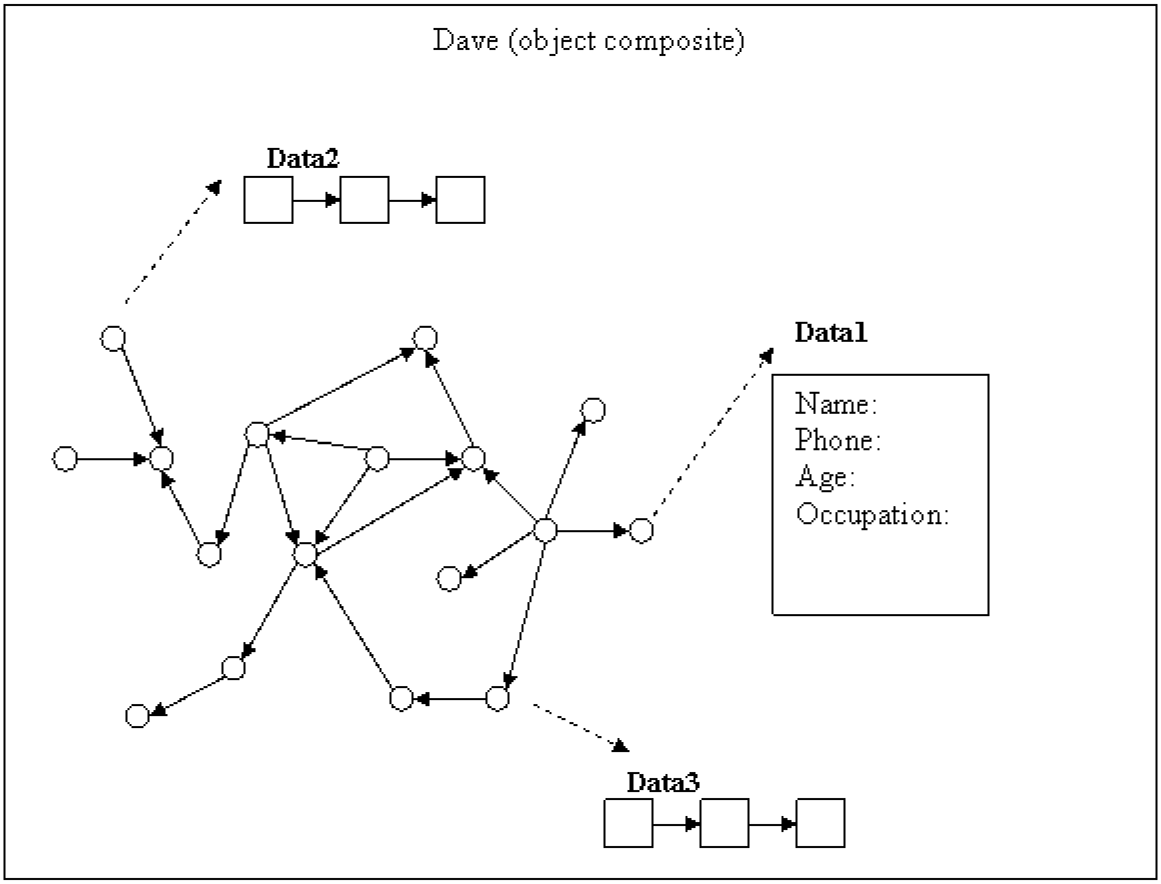
The answer is by lessons learned in school. Teachers taught us, through class lessons, to always focus on a person's face to identify them. This intelligent lesson instructs us to always look at a person's face to id them. By focusing on a person's face, the face becomes stronger and it will be the dominant encapsulated object in a human object. Another factor is that human beings focus on the face because it gets their attention. The voice of a person comes from the face, so we innately focus on the face.

The last example shows us how the encapsulated object "face" is the dominant object in the object "person", but what about information about a person? How do we know that a person's name is the most important id? The answer is from observing different fillable forms. Notice that in all forms or database entries, the name is the first data representing a person. By looking at forms, we can also map out the second or third most important data about a person. We know that age, gender, phone number and address are the most likely secondary data listed in a form.

We can also use intelligent pathways to determine what are major or minor data regarding a person. For example, rarely does anyone want to know a person's religion or what their pet's name is. These data are considered minor data.

Intelligent pathways learn these things innately or by lessons from teachers in how data should be organized. Referring to FIG. 7, the diagram shows an object composite (a network) on dave. These data have relations to each other based on association. In data1, a form of dave is present, whereby it is a static data; which lists major information about dave such as his name, phone number, age and occupation. In data2, a visual movie of dave is present and the face of dave is the most prominent data. The movie sequence can be static data or linear data. It could be an experience the robot had when encountering dave in the past. The last data, data3, is a sound recording of dave when the robot was talking to him on the phone. This sound recording has voice patterns that distinguish dave from anyone else.

FIG. 7



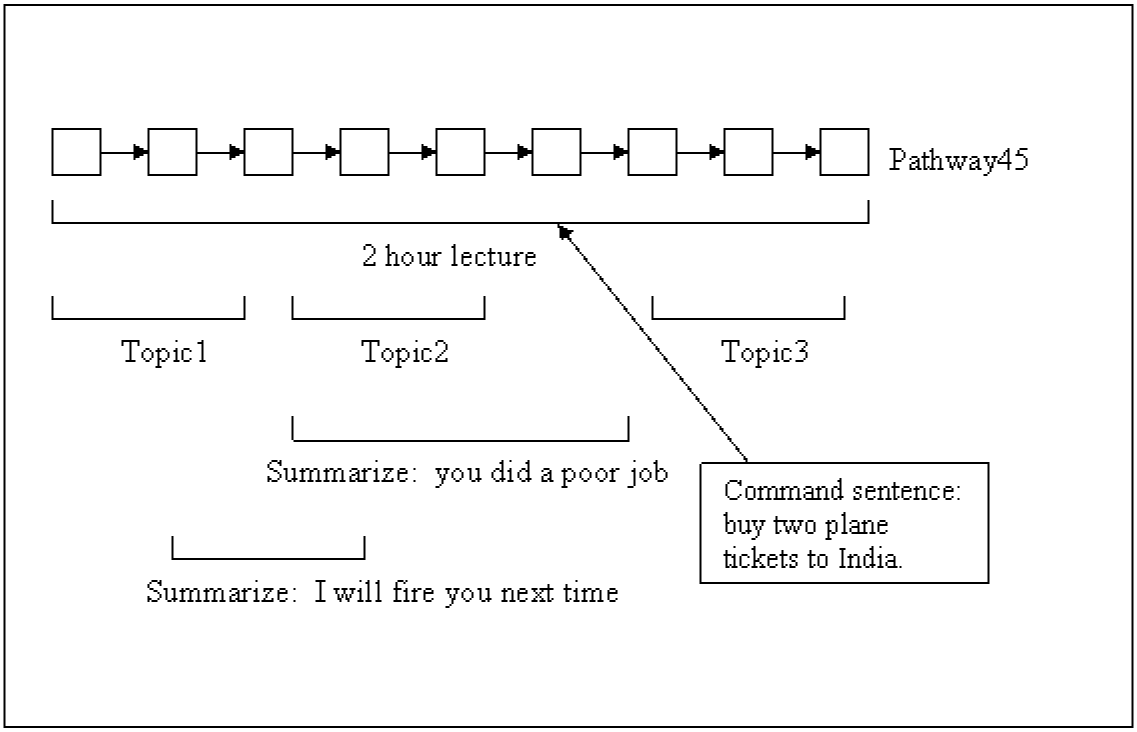
Lessons learned in school to organize data in memory

Lessons in school will create intelligent pathways in memory that will interpret data. If the robot had to listen to a 2 hour lecture from a boss; and in the two hours, the boss has given 1 command sentence, the robot is intelligent and will id and remember the command sentence. All other data in the 2 hour lecture is minor data, but the important data is the command sentence. The robot can also use intelligent pathways to summarize and interpret what the boss wants the robot to do in the future.

Maybe, in the 2 hour lecture, there is no clear command sentence or the boss doesn't specifically tell the robot what to do. The boss is telling the robot ambiguously what he should do in the future. Intelligent pathways in the robot's memory interpret the 2 hour lecture and summarize it so that the robot understands the most important information. These activated element objects are the robot's conscious that are generated by intelligent pathways. These conscious thoughts help to better organize data in memory.

Referring to FIG. 8, in the 2 hour lecture, the robot was able to activate conscious thoughts that relate to objects, events and actions. Some of these activated thoughts are logical analysis, summarizing of events, additional data related to objects and so forth. These activated thoughts (or element objects) help organize data in memory when the current pathway self-organize in memory. Objects, events and actions are delineated and boundaries are set, whereby individual data are hierarchically structured. For example, the 2 hour lecture is grouped as one data. Within the 2 hour lecture are topic1, topic2 and topic3. Within each topic are their respective inner groupings. Additional data like summarizing and id of an important command sentence also help group data in memory in an intelligent manner.

FIG. 8



Activated thoughts to organize data

Thus, when the robot wants to search for data on the 2 hour lecture experience, he can search the actual pathway45 or search the activated thoughts (element objects) to find data. In fact, if the robot's brain searches the activated sentences, the search process will be much faster. For example, if topic2 was a speech on nanotechnology, the search function will go into topic2; and since topic2 has reference points to the beginning and ending of topic2, it can search for more specific data on topic2. This would be much faster than searching all data in pathway45.

Pathways in memory forget data and pathway45 can break up into a plurality of fragmented pathways. The activated element objects help the robot's brain to reference data even when pathway45 is forgotten. For example, if the robot experience pathway45 10 years ago, he can still recall some images and activated thoughts. He might not be able to remember everything, but the most important data is still stored in memory.

Compatibility for different data types

All data in memory has same, similar or different data types. Referring to FIG. 7, data1, data2 and data3 are all in different data types and the length of each data is different. How will the search function find specific data or analyze data or compare different data types??? The answer is through intelligent pathways stored in the robot’s memory.

Chapter 3

Retrieving data from memory through intelligent pathways

Data in memory are stored using different data types. These data types will come in the form of 5 sense objects, activated element objects, pattern objects or hidden objects. Since the data in memory are different, how will the search functions work? The answer is through intelligent pathways. The robot has to learn how to analyze data in the real world. Teachers have to teach the robot how to look at a form and extract information from the form. For example, if the robot was asked a question such as: "can you read me the name on the form?", the robot has to look at the form, locate the name, and read what the name is. Then, the robot has to face the teacher and tell her the name that was on the form.

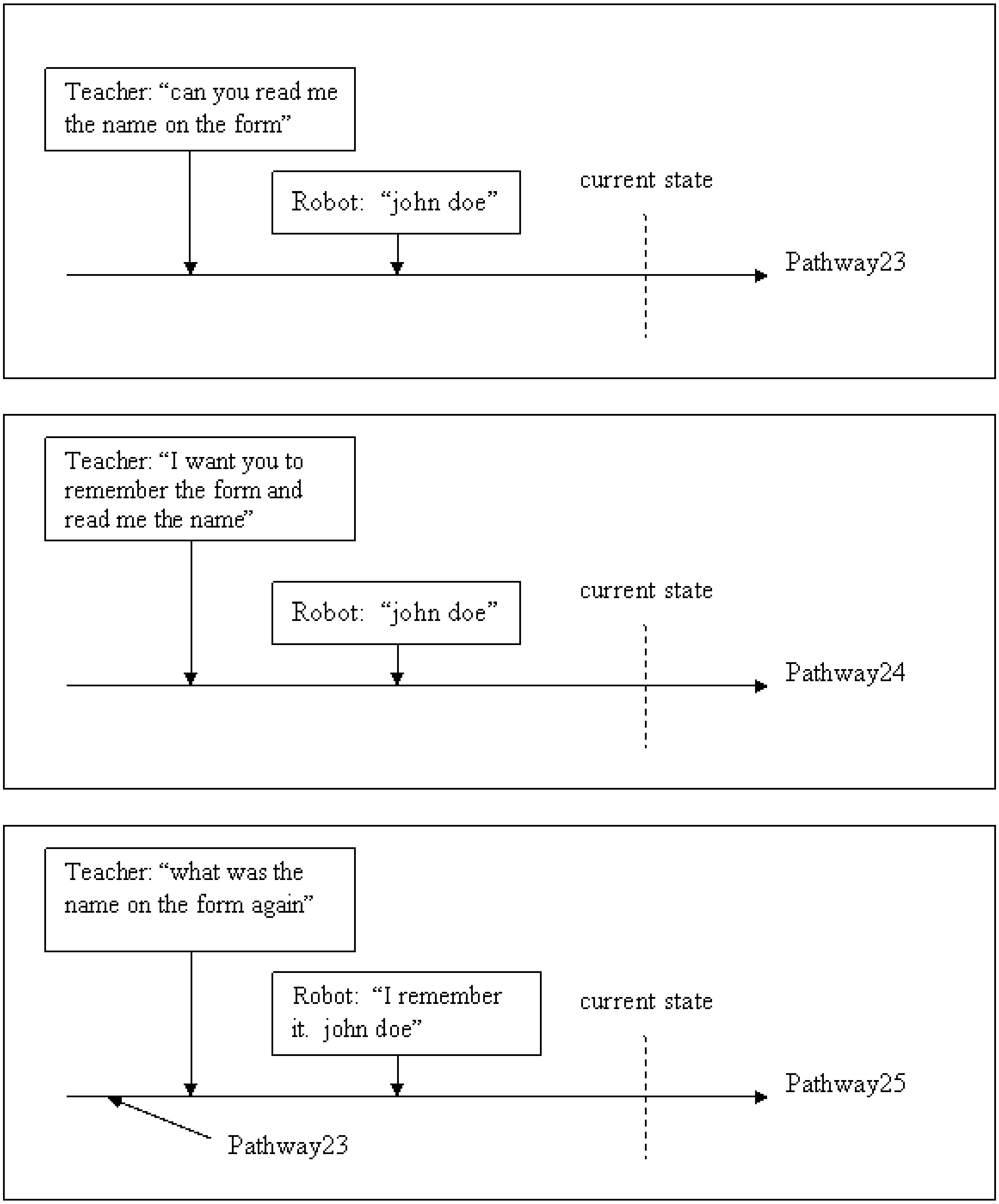
Using this lesson (an intelligent pathway) the robot can extract a form image from memory and use the lesson to extract the name on the form image. This form image is data stored in memory, as an experience by the robot.

In another example, a teacher plays a song for the robot and after the song is over, the teacher asks the robot to describe what category the song belongs to. Next, the robot will analyze aspects of the song, such as rhythm and lyrics; and say to the teacher what the most likely category of the song is.

Using this lesson (an intelligent pathway) the robot can extract a song from memory and use the lesson to output the category the song most likely belongs to. This song is a memory of the song, experienced by the robot.

The question is: how does the robot know that the intelligent pathway learned by teachers can be used to analyze and extract information from data in memory? The answer is through patterns. FIG. 9 is a diagram depicting how the robot finds the patterns between lessons to analyze and extract information from data in the real world and analyze and extract information from data in memory.

FIG. 9



In pathway23, the teacher teaches the robot how to analyze and extract information in a form he is currently looking at. If the robot doesn't understand the steps to analyze or to extract information, then the teacher has to teach the robot these steps.

Intelligent pathway23 is universalized in memory, which means the robot can answer any question related to searching and extracting information on a form. This can happen only if the robot was taught many similar examples. For example, the universal pathway23 can answer similar questions such as:

1. “can you read me the phone number on the form”
2. “can you read me the occupation on the form”
3. “can you read me the home address on the form”

The universal pathway23 can cater to similar questions. The instructions to analyze and extract information are very similar.

In pathway24, the teacher gives the robot a command: “I want you to remember the form and read me the name”. The robot’s brain will extract the memory of the form and use pathway23 to analyze and extract the name on the form. Finally, he will tell the teacher what the name is, which is john doe.

In pathway25, the teacher gives an ambiguous command to the robot; and based on logic (another intelligent pathway) the robot assume the teacher wants to know the name that was on the form. Pathway23 has already been encountered and the robot answered the question in pathway23, he simply has to repeat that answer.

To sum things up, the robot is taught many various examples and based on patterns and logic, he will know that pathway23 can be used not only for analyzing and extracting information on a form the robot is currently looking at, but also, to use pathway23 to analyze and extract information from a data in memory.

Intelligent pathways to set a goal, plan a strategy and achieve the goal

The pathways in memory can create any form of intelligence. Referring to FIG. 10, an intelligent pathway to set goals, plan a strategy and achieve goals can be created (called A1). Teachers will have to teach the robot what are the steps to achieve a goal. The steps are: to set goals, plan a strategy, trial and error and successfully achieve goals. During the planning and trial/error steps, there exists a loop because some strategies might not work and the pathway has to loop itself again to plan a new strategy and to take a different action.

FIG. 10

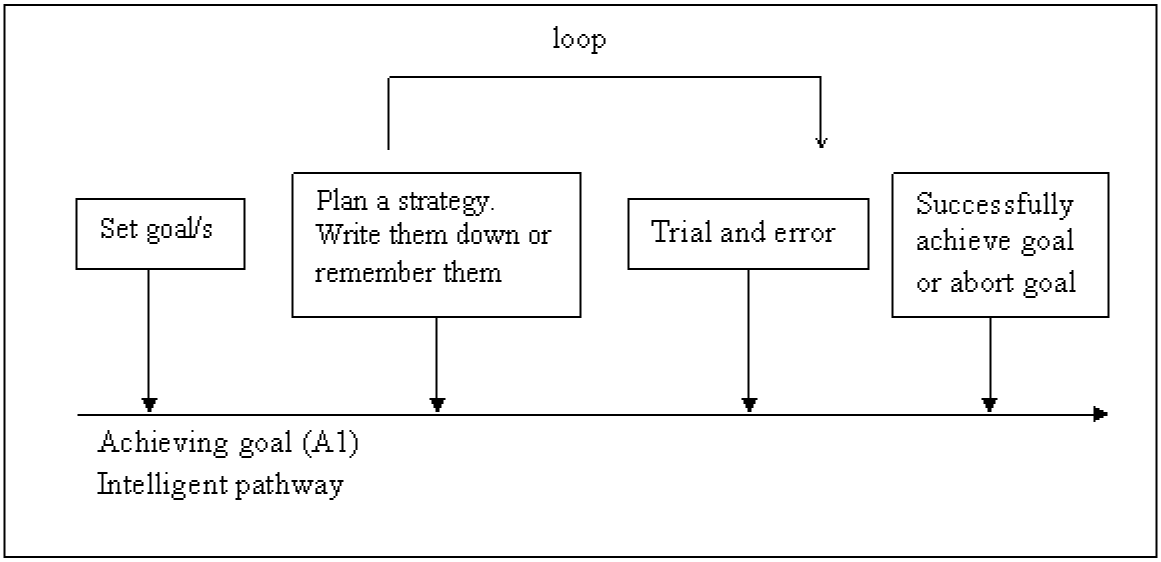


FIG. 11

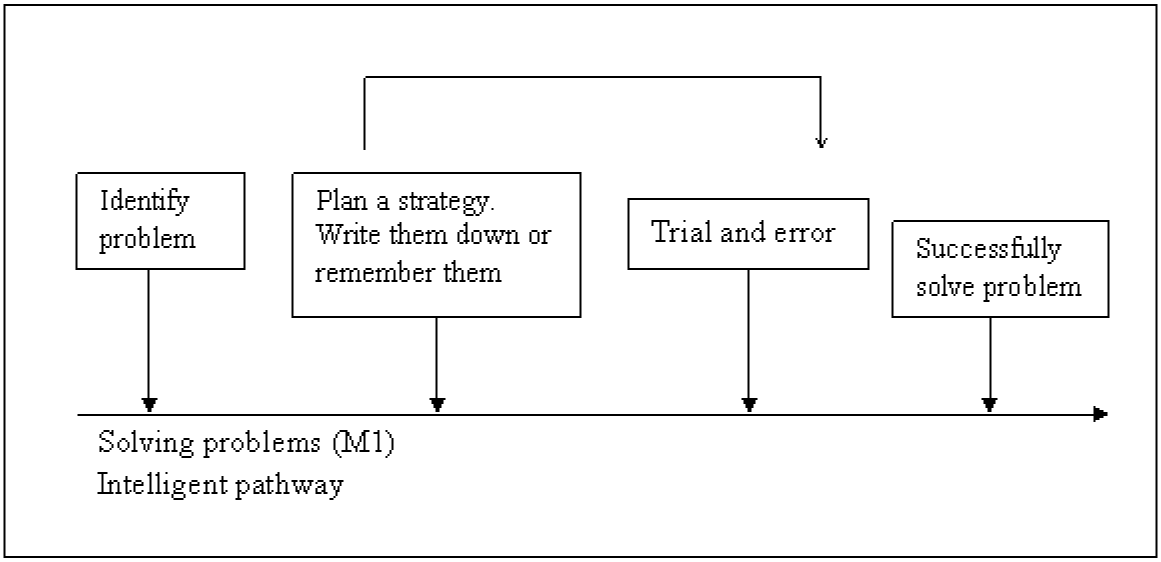


FIG. 11 is a diagram depicting an intelligent pathway called M1 that basically solves a problem. The steps are presented in the diagram on how the robot will solve a problem. Notice that intelligent pathway A1 and M1 are very similar. In memory, these two intelligent pathways will be very close to each other because of their commonalities.

Both A1 and M1 can be universalized so that they can cater to a wide variety of situations. For example, the problem solving pathway M1 can be used for all problems. It can be used to solve a business problem, a math problem, a science problem, a conflict problem, a personal problem, a videogame problem, or driving problem. On the other hand, pathway A1 can be used to achieve "any" goal/s. The robot can use A1 to solve a math problem or to drive a car to a destination or do a math problem, or do a series of math problems or past a videogame.

Similar types of problems will self-organize. For example, doing a science problem might be similar to doing a math problem or driving a car might be similar to driving a motorcycle. Intelligent pathway A1 and M1 are universal pathways that will organize specific pathways in their hierarchical trees.

Another note is that pathway A1 can encapsulate pathway M1 and/or vice versa. Steps in pathway M1 might be to achieve 20 different separate goals. Or steps in pathway A1 might be to solve 10 interconnected problems. The brain of the robot will self-organize data in memory so that similar data will be grouped together. A bootstrapping process occurs, whereby pathways in memory add, delete and modify data on previously learned pathways.

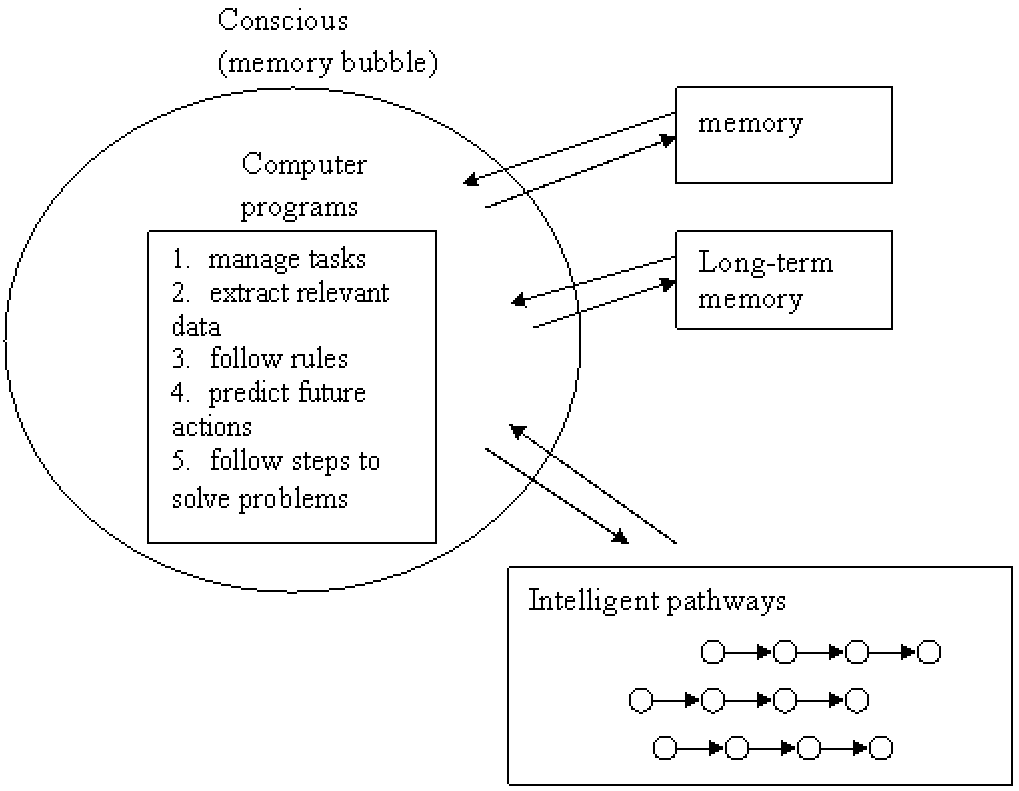
Intelligent pathways used to manage tasks, plan strategies and follow rules

We learned that intelligent pathways can search and extract data from memory. These data in memory have different data types; and the storage of data are not preprogrammed by computer scientists. The brain learns how to organize data through examples and lessons from school.

Other remarkable things intelligent pathways can do for the robot are: to manage tasks, plan strategies, extract relevant data and follow rules. Referring to FIG. 12, the intelligent pathways the robot's brain uses to take action controls the computer program in the conscious. The optimal pathway controls vital things like what the robot's goals are, what tasks to do in the future, what rules should be followed and what information in memory should be extracted for the purpose of logical deductions.

The diagram in FIG. 12 really shows the overall idea behind the conscious and how the robot actually controls itself in an intelligent way to take action.

FIG. 12



The robot’s conscious applied to a videogame

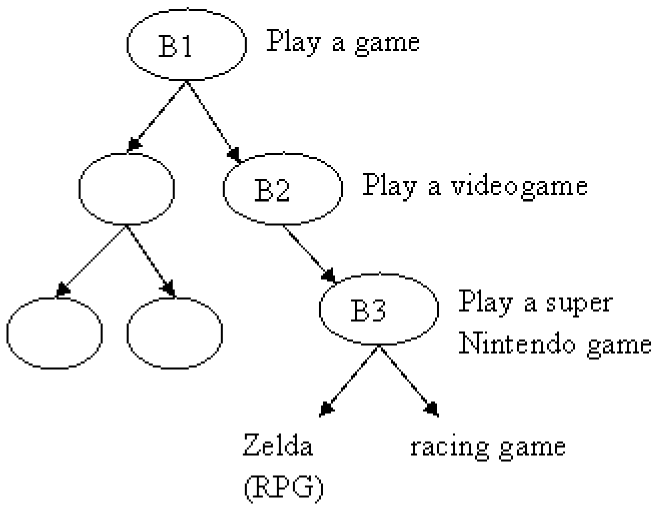
The task of playing a videogame is a good example of how human robots act intelligently in a dynamic environment. When playing a game, the robot has to use logic and common sense knowledge in order to past the game. Playing a racing game or a RPG game is very difficult and the player has to understand the rules of the game, the objectives of the game, how the controls work according to the game, how to solve problems in the game, how to get the character from one destination to the next, how to come up with plans to beat the game and so forth.

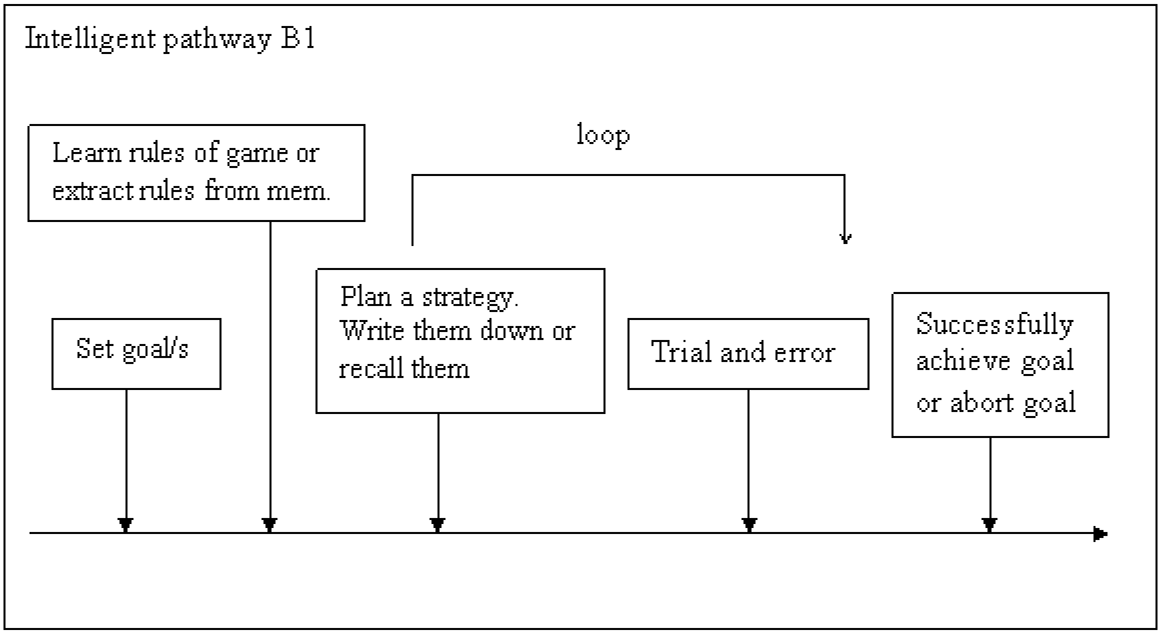
The legend of Zelda is a very good example because the robot has to use human-level intelligence in order to past the game. The Zelda game isn’t like a side-scrolling action game, whereby the player accomplishes levels in linear order. In Zelda, the player has to talk to characters in the game and these characters will tell the player what to do next. If the player doesn’t follow the instructions from characters, he will get lost and will not past the game. The key to passing the Zelda game is to use human logic to come up with planned strategies and to take action; and through countless trial and error.

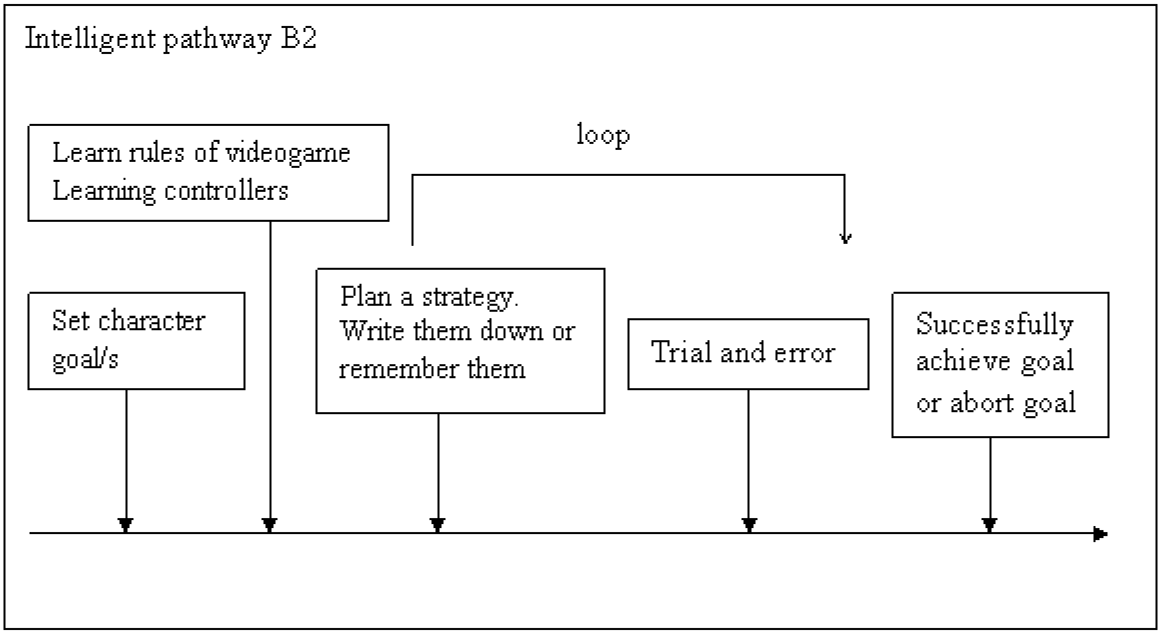
The intelligence in the robot’s brain comes from a bootstrapping process, whereby new data is built on top of old data. When we play a videogame, we are actually using our knowledge of playing a general game. The lessons of playing sports games in real life, the lessons of playing chase master, the lessons of playing a board game, the lessons of driving a car, the lessons of an occupation actually comes from one universal way of playing a game.

Referring to FIG. 13, pathway B1 is a universal pathway to play any game. The steps in B1 are very general, in that, all games played have these linear steps. If you observe a sports game or a board game, they have these general steps. B2 is a more general pathway to play a game. In this case, B2 represent playing a videogame. All the intelligent pathways (B1-B3) are all encapsulated and structured in a hierarchical manner so that the data goes from general to specific. Intelligent pathway B3, on the other hand, record detailed steps to play a specific game. If the game is the legend of Zelda, the steps to playing this game are different from the steps to playing a racing game.

FIG. 13 Hierarchical structure of playing a game







In intelligent pathway B2, the player has to set goals in terms of the type of videogame played and based on the player's goals in the game. Next, in the videogame are various rules and boundaries the player has to follow. Then, the player has to also know what buttons control what actions in the videogame. When playing a videogame, the controller controls the actions of the character on the monitor the robot is seeing. On the other hand, if the robot was playing a real sports game, he has to use his body to take action.

Knowledge of the videogame

When the robot plays a given game, the robot's conscious extracts relevant data about the game (activated element objects). The computer program in the conscious will open a valve and knowledge about the game will start to pour in. The intelligent pathways will help a great deal in extracting the relevant knowledge, but the computer program in the conscious actually controls when certain rules might be needed or when certain knowledge is needed for logic in a given situation.

For example, when playing a real baseball game, knowledge about what happens during the timeline of the baseball game will be mapped out by the conscious. During the estimated timeline of the game, knowledge will enter the conscious at specific times. In the beginning of the game, the player (the robot) will know where to go. It will follow the commands given by the coach. The coach will designate each player their positions and to set the batting lineup. Next, the coach will tell everyone what the hidden signals are for this game. The robot's conscious will tell the robot where to be and what instructions from the coach he has to follow.

When the game begins, the robot already has tasks it will do based on the lecture by the coach. Previous knowledge about baseball will pour in such as where the player has to be and what its roles are. The rules of the position the robot will play must be known. These rules sets up the limited actions the robot can take, what rules to follow, what actions to take at given situations and so forth.

As stated in the last chapter, the conscious primarily has 4 containers: task container, rules container, planning container, and identity container. Based on the position the robot is playing, the rules container is filled with rules (most notably if-then statements). If the robot is playing center field, some of the rules might be: If the fly ball is hit hard, back up; if the fly ball is hit lightly move forward; if catcher throws the ball to second base, move forward. These are the rules that the computer program in the conscious is trying to maintain in the rules container.

As the robot changes positions, the rules in the rules container will be filled with their respective position rules. The rules of a pitcher are different from the rules of a left fielder. The computer program within the conscious will extract and maintain the rules in the rules container based on the tasks in the task container.

Some games do not need to follow commands from someone. Playing the legend of Zelda is a game that simply requires knowledge from memory. The robot's conscious will extract all the tasks and rules of a RPG game (short for role playing game) and at the beginning of the game, the robot knows what type of game it is, what its goals are and what rules to follow.

This information has been learned through magazines and strategy guides of previously played RPG games. The knowledge of what to do and how to play the game has been described in magazines. The robot learned how to play an RPG game through reading videogame magazines.

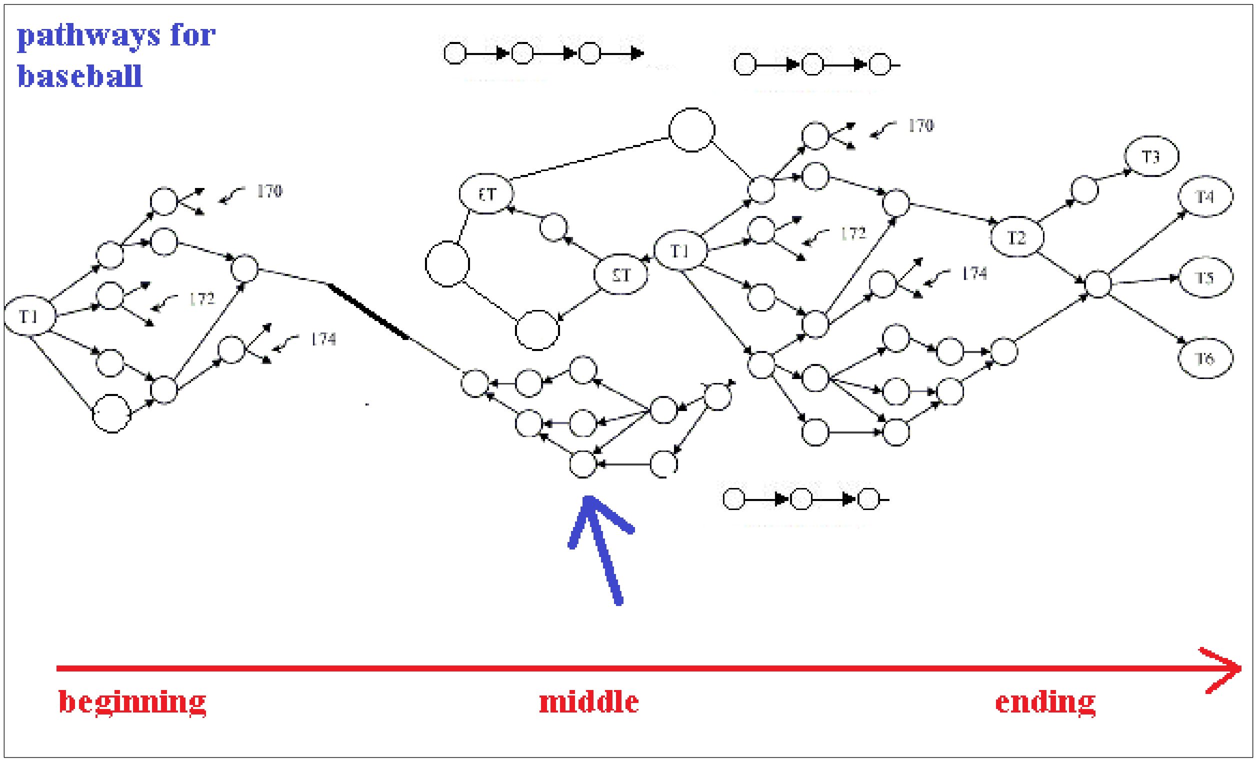
For me personally, when I play a new RPG game, my first goal is to understand the story and what the game is about. Then, I would follow the guidelines from previous RPG games, which are:

1. talk to all characters in the game and follow their instructions
2. read the instruction manual
3. keep a map of the game. You will find a map either from the game or an instruction manual.
4. past level 1 before going to level 2 and so forth.
5. if you get stuck in the game, ask yourself why you can't advance.

These are just some important facts that are needed in order to play the legend of Zelda or any RPG game. These facts will pour into the conscious the moment the robot wants to play a RPG game. In fact, specific tasks/rules will pour into the rules container and task container at specific situations. During the game, problem solving pathways such as M1 will be used to solve problems in the game or to come up with imaginative ideas to go around obstacles. Other intelligent pathways will be needed during the game such as pathway A1 to achieve goals.

Chapter 4

In the last chapter you learned that pathways can generate self-created computer programs based on learning from the real world, and learning based on a humanoid body with 5 senses identical to a human. The robot is experiencing life from a first person point-of-view and storing these experiences as pathways in its brain.

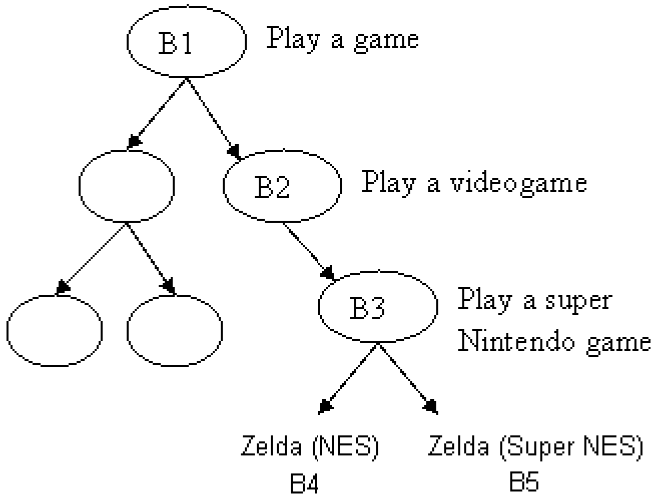


Over time these pathways generate self-created computer programs such as relational databases, search engines, logical systems, decision making systems, and what have you. The most critically important thing is that the robot's brain created a self-aware decision making system that can store information, search for information, predict the future, and make logical decisions for all given tasks. A Task that it hasn't learned before will be using similar pathways in memory that is similar to the task and adapting its instructions. If the robot wants to learn a new skill it has to read books and practice to acquire that skill in its brain (just like humans). Knowledge in the public school system (from K to college) gave the robot all the knowledge and skills it needs to do any human task.

These pathways are structured hierarchically to form massive forest and trees of "tasks". Each task has its own network or series of interconnected network of pathways, and they are structured in a hierarchical manner, whereby each task share information and procedures -- going from general to detailed. Referring to FIG. 13, this diagram depicts a universal pathway to play video games. At that top are the general instructions to play a video game, where all video games are applied. Usually pathway b1 is used to play an unknown game the robot has never played before. It uses general rules, linear instructions and strategies to beat the game.

For example, pathway b1 can be used to play board games or play a real basketball game or play video games.

FIG. 13 Hierarchical structure of playing a game



In the lower levels are the individual games sorted out in categories and genres -- sports games will go here, RPG games will go there, and racing games will go here. Each game, or task, is stored based on 2 things: learned groups and commonality groups. A cat and a dog look similar so they are grouped based on visual commonalities. However, a dog and a mouse look different in terms of size, shape, color and 3-d model, but both species are classified as animals. “Animal” is the word that links the two different species together as a learned group. Both commonality groups and learned groups co-exist together in memory to organize associated objects. It also assigns meaning to language. For instance, the sequence images of a cat is associated to the text cat, and the text cat is associated to the sound cat. All three objects have the same meaning.

These 2 self-organization groups: commonality groups and learned groups for data storage works for both static images and sequential tasks. The robot experiences life and stores these data in memory, and organizes them in a hierarchical manner. In terms of storing tasks, the pathways build on top of each other recursively to form complex, self-created computer programs.

Most of these network pathways are in the form of nested if-then statements, recursive tasks or conditional operators. Referring to FIG.13, pathway B2 shows a more specific pathway to play videogames, B3 is a detailed pathway to play a role playing game, and B4 is a pathway to play a RPG game called the Legend of Zelda. Pathway B3 is a universal pathway to play “any” RPG game. This means the robot can use this pathway to play unknown RPG games that it has never played before. It doesn’t understand the objective of the game, it doesn’t know the rules nor does it know what to do in the game. This pathway B3 gives the robot all the general instructions it needs to seek out the rules, objectives and strategies of the game. The more it plays, the more information it collects from the game and therefore it gets better and better at the game through trial and error. This universal pathway (B2) actually teaches the robot how to seek out knowledge and to play an unknown game through practice to become an expert in the shortest amount of time.

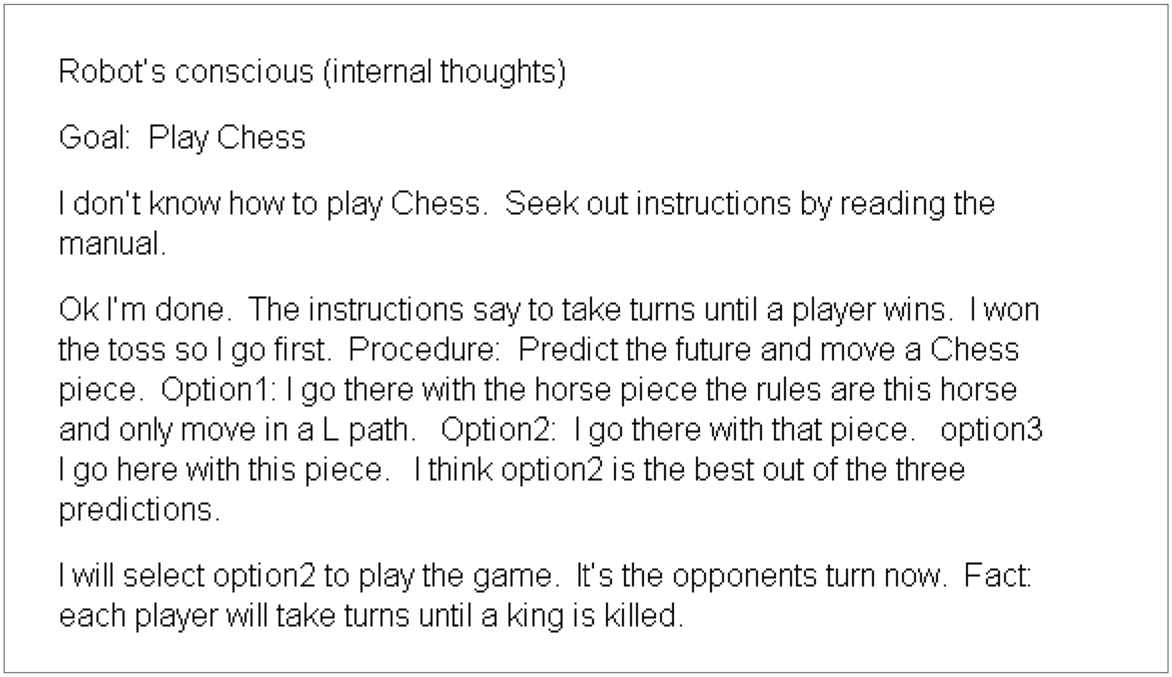
In the lowest levels are where the objectives, rules and procedures are stored about specific RPG games. Let's say that the robot never played the Legend of Zelda before and it is playing the game for the first time, pathway B3 is used to play this unknown game. After many hours of trial and error, it creates B4 in memory -- it remembers all the pitfalls and the rewards of the game. Their faculty for adaptation is able to create optimal pathways to play the game for each level, and for each scene in the game. Any new levels it encounters, there is a unique area in the tree that stores the new information as memory experiences for future references.

Now let's say that the robot played Zelda for the NES and now it is playing Zelda for the super NES. Both games are different with different levels, but the play mechanics is the same. What the robot will do is take pathway B4, representing the old Zelda game, and use that as knowledge to play this brand new Zelda game. The play experience will be stored close by called pathway B5. Pathway B5 will borrow skills and facts from pathway B4, and it uses trial and error to attempt to play the game to the best of its abilities. Said borrowed skills and facts might be this statement, "the goal of the first Zelda game was to rescue the princess from the evil Wizard, this new Zelda game might have the same goal. After the robot analyzes the game and do a few run throughs, it will find out that this activated thought is indeed true. The objective of the game is to rescue the princess from the evil wizard. All of the other rules and objectives of the game are also the same or similar. So instead of relearning this stuff through countless hours of game play, it can borrow knowledge from similar games already learned.

Also, there is knowledge that has nothing to do with the task; that activates in the form of common sense knowledge that will help the robot to play the game. Thus, in addition to pathway B4, the robot is also using common sense knowledge and adaptation to play this new Zelda game. It's using all knowledge from all fields that it had previously learned from K to college. Things like how to add numbers or how to solve a simple puzzle are all knowledge required to play specific levels in Zelda. There might be a simple side level where the player (the robot) has to add numbers and put puzzle pieces together to complete the level. If the robot doesn't know how to add numbers or do simple elementary problems, it won't be able to pass the level and therefore it I won't be able to move on with the game.

Playing similar board games

Like I said before, pathway B1 is a universal pathway to play any game. It can be used to play real basketball or used to play board games. In the game of chess the robot is given instructions about the game like "each player takes turns to play the game", "the objective is to defeat your opponent by killing the king.. blah, blah, blah.", "the rules are.. ". The robot learns these objectives and rules by seeking out these facts in books or instruction papers. In the game, if the robot doesn't have the instructions layed out, it can watch people play and assume what the goals and rules are, or he can simply ask someone. Here is a diagram of internal thoughts of the robot while it learns to seek out the objectives and rules of Chess:



Robot's conscious (internal thoughts)

Goal: Play Chess

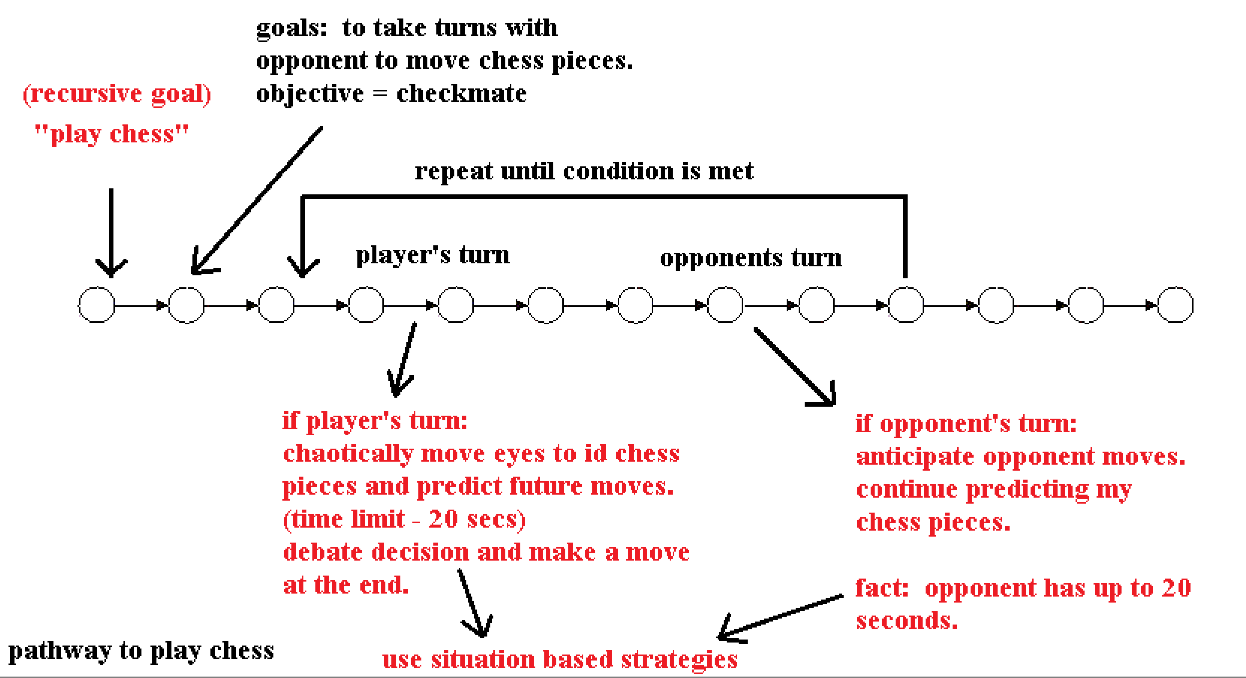
I don't know how to play Chess. Seek out instructions by reading the manual.

Ok I'm done. The instructions say to take turns until a player wins. I won the toss so I go first. Procedure: Predict the future and move a Chess piece. Option1: I go there with the horse piece the rules are this horse and only move in a L path. Option2: I go there with that piece. option3 I go here with this piece. I think option2 is the best out of the three predictions.

I will select option2 to play the game. It's the opponents turn now. Fact: each player will take turns until a king is killed.

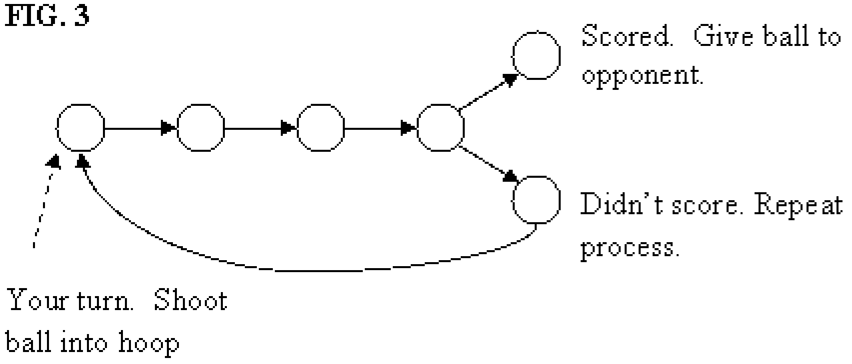
These rules, strategies and objectives of the game are stored in memory and create a self-created database of information about the chess game. Next, it also creates conditional loops and recursive pathways to play the game Referring to FIG. 32, this is a universal pathway to play Chess. The rules and objectives it learned will be stored in the pathways as conditional operators (for-loops, while loops, if-then statements, linear instructions, recursive tasks, etc.). For example, when the robot reads a statement like: “each player will take turns to play the game”, it will create a conditional for-loop to loop the game. The sentence actually creates a for-loop with conditions inside a pathway. The player (the robot) goes first, then the opponent, then the player, then the opponent, etc. The rules learned are also part of a condition. If it’s the player’s turn, the brain has to follow all the rules while making a move. When it’s the opponent’s turn, the player has another set of rules and procedures to follow, like anticipate the opponent’s future moves and, also, what it needs to do in the future to beat the opponent.

FIG. 32



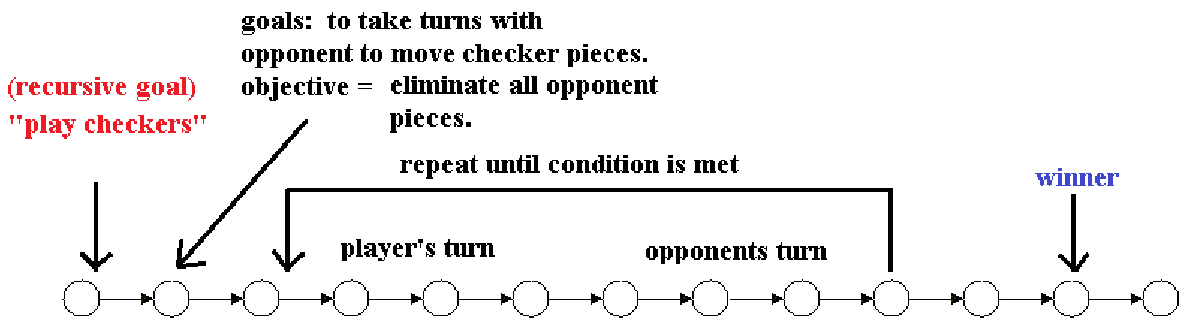
Thus, sentences stored in pathways form computer programs to do tasks. If the robot is flying an airplane, a computer program is generated in its mind to fly the plane. If the robot is playing chess, a computer program is generated in its mind to play chess. If the robot is playing basketball, a computer program is generated in its mind to play basketball, and so on. Based on its goal, it will extract relevant pathways over a period of time and the pathways form this self-created computer program in its mind to do a task/s.

FIG. 3



Since Checkers is so similar to Chess, it is stored right next to the Chess area in memory. Hypothetically, let's say that the robot is a master at Chess, but has never played Checkers before. Instead of relearning everything, the robot can actually borrow knowledge and skills from Chess to play checkers. Referring to FIG. 34, the for-loops are the same because in checkers players have to take turns playing. The rules are different so the robot has to adapt and change the rules. The point I'm trying to make is that it is using a template of the Chess game to play a brand new Checkers game. So learning the new rules objectives and procedures for checkers becomes easy.

FIG. 34



pathway to play checkers

This one sentence borrows (or references) the for-loop pathway from the Chess area to the Checkers area, "checkers is the same as Chess because players have to take turns to make moves until someone wins." Board games like Monopoly or the Game of life also use this for-loop, but their pathways have multiple players instead of two players. The robot's brain borrows pathways from memory and adapts the pathways to a new type of game. This is how the robot can learn any game regardless of how complex they may be. Sentences also have patterns for searching for data and inserting, storing and modifying data. For example, this sentence searches for and modifies wrong data in memory: "the Earth isn't flat its round".

This method to borrow skills in memory is similar to sports games. People that can play soccer can easily adapt that knowledge to learning a new sport like basketball or football. This general knowledge to play sports is actually learned in PE class in the public school system, where teachers teach students how to teach itself to play a new sport or to borrow knowledge from a pre-existing sport. Learning to refine its skills through practice is also previously learned through teachers in school. "Teachers teach the robot how to teach itself how to learn a new skill, by seeking out knowledge, practicing through trial and error, and refining its underline skills."

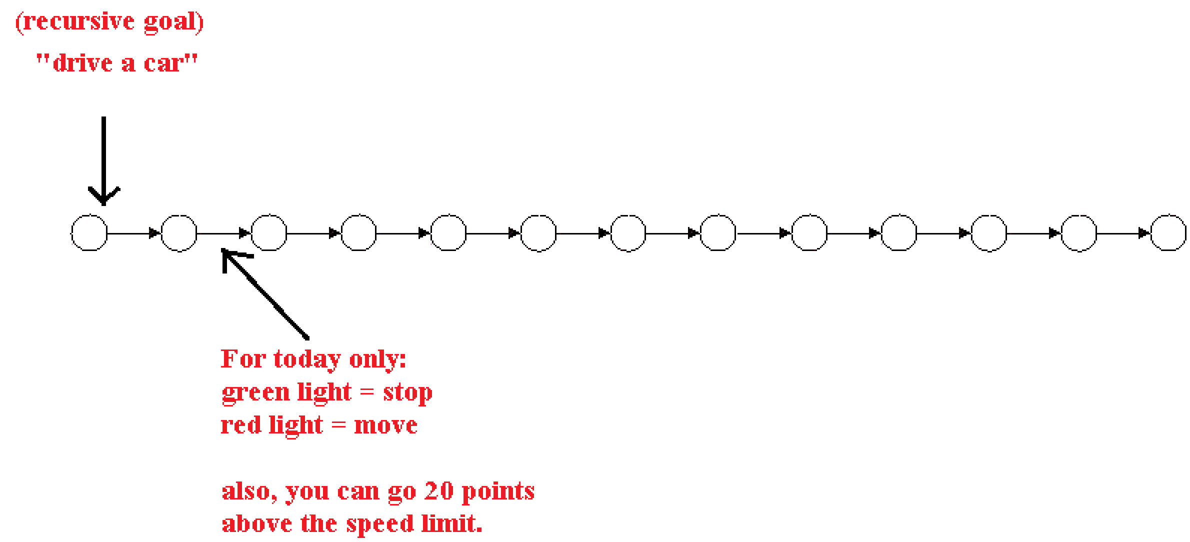
Here is a pathway to do recursive tasks like make a sandwich. The process of making a sandwich isn't as simple as people might think. This universal pathway has to have the capability to make a sandwich under "all" circumstances and they must follow all the laws set forth for food safety . For example, if a customer says he/she is allergic to onions, the robot has to leave out the onions from the sandwich. Now what if the onions are mixed in with the sauce like in a Big Mac? This means the robot has to remake the sauce without onions. Yet, in another case, what if the robot, during one step, inadvertently drops the lettuce on the ground? A human worker will know that the floor is filled with dirt and bacteria and understand that the lettuce is now inedible by humans. Will the robot pick up the lettuce and service it to the customer?? Maybe the floor isn't that dirty and the robot decided to wash the lettuce and then put it back on the sandwich, again? What if it was serving soup and sees a big roach swimming around? In both cases the robot would serve the food to a customer and, more than likely, the restaurant will be sued. In order to solve these problems the robot needs common sense knowledge (all of it, from K to college).

This is why it is so damn important for the robot to go to school to learn everything the proper way. Because if it doesn't, it won't even be able work as a chef at your local McDonalds. Even something simple as making a sandwich requires all the knowledge learned from Kindergarten to college, believe it or not. All the ethical laws and all basic knowledge about safety rules must be learned and deeply embedded in the robot's brain when making decisions.

Driving a car example

Under rare conditions the robot has to break the rules of driving. FIG 22 depicts the robot playing a driving video game, and in the game there are different rules for driving. For example, the game states that red light is go and green light is stop. And the speed limit is raised to 20 points more than its sign states. The robot, at this point, has to adapt its knowledge of driving to these new rules.

FIG. 22

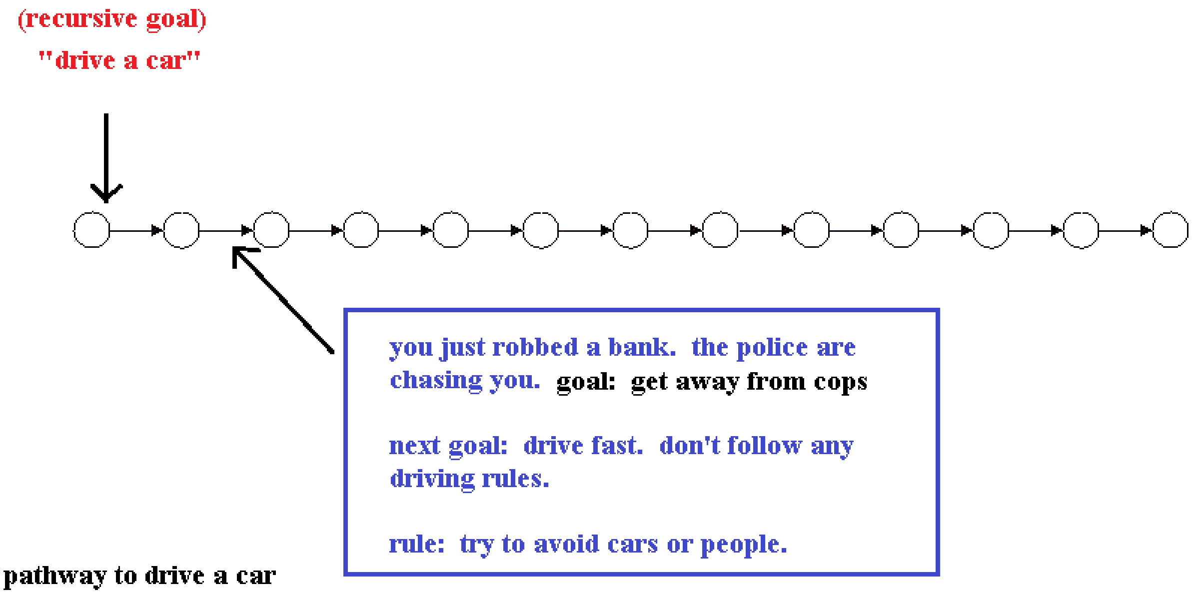


pathway to drive a car

In the real world, let's say that the government decided to change the rules of driving dramatically, the robot has to adapt to these new rules. And it does it in an optimal way where it doesn't have to relearn everything. It can read a modified driving book and use that knowledge to change its rules, procedures and objectives. It must also have to practice to drive under these new rules to really refine its skills.

In very rare conditions, driving rules are thrown out the window all together. In this diagram, the robot just killed someone and robbed a bank. For self-preservation, the robot's main objective now is to flee the cops. It will drive really fast, breaking the speed limit signs and driving in roads that are illegal. Because the robot is smart enough to know that when caught he will either serve life in prison or get the death penalty. Self-preservation is a decision it has made to avoid certain death, even if it means endangering people on the streets.

FIG. 24



you just robbed a bank. the police are chasing you. goal: get away from cops

next goal: drive fast. don't follow any driving rules.

rule: try to avoid cars or people.

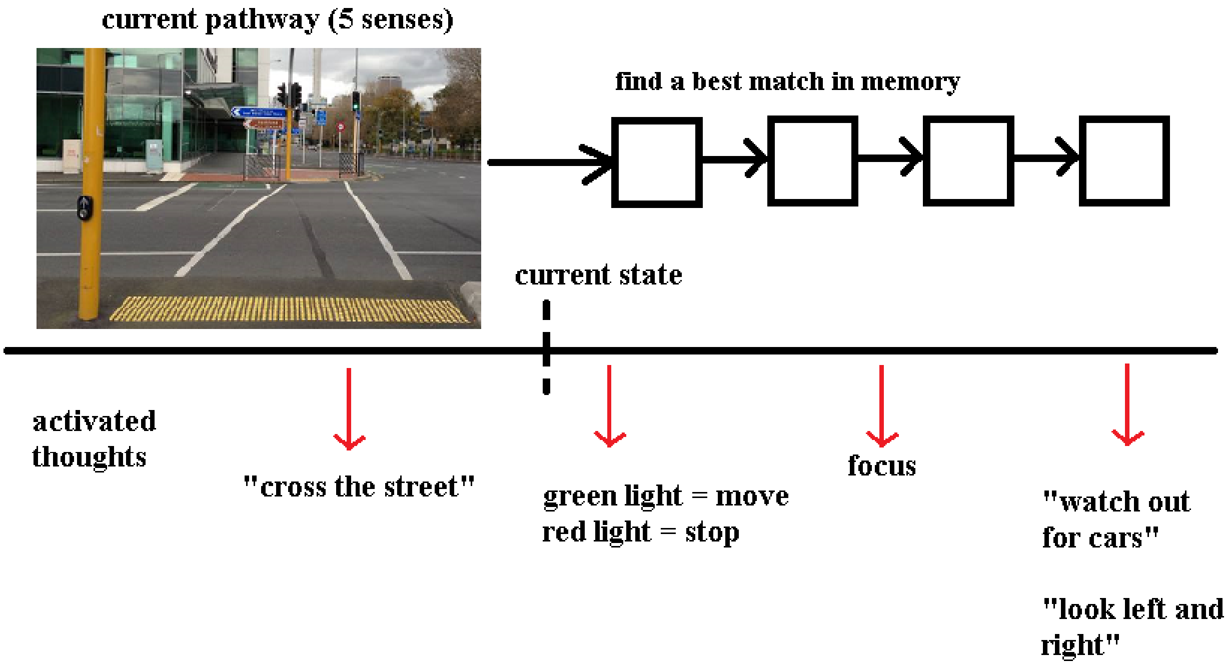
pathway to drive a car

Pathway matches are based on its goals and objectives not its current environment.

When the robot is searching for a pathway match in memory, it isn't focused on the current environment. It's actually focused on the linear internal thoughts or linear decision making stored in pathways. For example, when the robot wants to cross the street, it doesn't matter what the street looks like or how wide the side walk is, the internal linear instructions to cross the street activates in the mind to cross the street. If the robot is making a sandwich, the internal linear steps activate in the mind, it doesn't matter when the robot has to make the sandwich, or how far the sink is from the robot, or what the frig looks like. It is able to make the sandwich under any situation. The step-by-step instructions in the pathways are the data the robot is focused on to do a given task.

The instructions in pathways activate conscious thoughts to do work and forms the internal voices in the robot's mind. This conscious thoughts is like an invisible teacher to tell the robot things and make intelligent decisions. The environment can change or the jobs to be done can change, but the robot's conscious thoughts remain the same and can easily adapt to this ever changing environment. This is why the robot can do any human task under any circumstances, because of the "inner voices".

FIG. 14


current pathway (5 senses)
find a best match in memory
current state
activated thoughts
"cross the street"
green light = move
red light = stop
focus
"watch out for cars"
"look left and right"

Chapter 5

Managing multiple tasks simultaneously

When the robot drives a car, he is actually doing multiple tasks simultaneously. One task is driving in the middle of the 2 white lines. Another task is to follow traffic laws such as avoiding other cars on the road. Yet another task is to observe any traffic signs and obey them. If a traffic sign says 30 mph, then that means the robot has to drive less than 30 mph. Yet another task is to look out for sirens such as a police siren or an ambulance siren. Most of these driving rules will be stored in the rules container in terms of if-then statements. If you recognize a police siren, pull over to the side of the road. If you recognize an ambulance siren, pull to the side of the road and let it past you. The conscious of the robot will manage all these driving rules and to alert the robot when a rule has been violated.

Sometimes conflicting rules and rule interruptions are solved through logic. The solving of conflicting rules are done by intelligent pathways in memory. Teachers teach the robot how to solve conflicts when it happens.

Yet another task for the robot is to plan routes to get to its destination location. Intelligent pathways will be used in memory to search for its current location and maps and fabricated maps will be activated to instruct the robot in what roads to drive in to reach the destination location.

The computer program in the robot's conscious will activate a previously learned map of the environment. This map can be a very general elementary map of the environment. The computer program will mark where the car is currently located and it will mark where the car is going. Then, the computer program will draw sequence of lines to get from the current location to the destination location. The output of the computer program in the conscious will be a sequence of streets that the robot has to drive through to get to its destination location.

When downloading multiple files online, the server will send packets for each file equally. For example, if it takes 1 hour to download a 5 minute video and the user wants to download two 5 minute videos, the downloading will take two hours. Both video will be completely downloaded after 2 hours. Managing multiple tasks for a human robot works the same way.

The robot's attention can only be devoted to one task at a given time. If the conscious has to manage 2 tasks, it has to work on task1, stop task1, work on task2, stop task2, work on task1, stop task1, work on task2, stop task2 and so forth. The attention of the robot's conscious can only focus on one task at any given time. If the robot was driving a car, he has to manage multiple tasks simultaneously. The computer program of the conscious will be the engine that will switch between tasks. It will drive the car between the white lines, then it will observe the environment for any signs, then it will drive the car between the white lines, then it will observe the environment for any signs and so forth.

However, driving a car will require lots of simultaneous tasks. Sometimes people can drive on the road, eat breakfast and talk on a cellphone at the same time. Each additional task will put more strain on the robot's focus.

Multiple tasks for videogames

When playing a game like space harrier for the Sega genesis, the player has to manage multiple layered tasks. Space harrier is a first person shooting game and the character is flying in the air and he has to shoot space aliens. My approach to playing the game is to avoid any objects that come close to the character and to shoot as many space aliens as possible. The more space aliens I destroy, the more points I receive. Higher points would mean that I can earn extra life.

The two tasks for Space harrier are considered layered tasks because they are happening simultaneously. In fact, the conscious has to do more than two tasks at the same time. The character has to shoot specific types of enemies in order to receive powerups. Thus, in addition to avoiding objects and shooting space aliens, the robot has to also destroy specific enemies to get powerups.

Tasks are managed by the decision making of the robot at that moment. If the robot wants to get powerup, then it will go after specific enemies. Under certain circumstances, the robot might be tired and doesn't want to past the level, so he might skip the powerup. The robot might be playing the game for fun and he might deliberately hit an object. Thus, decision making is not just based on intelligent pathways in memory, but based on other factors such as emotions, mood, goals and so forth.

Complex rules can also play a factor in decision making. If the player has too many powerups, the CPU of the videogame will automatically send an undefeated enemy to rob the player of all powerups. This rule basically prevents the player from getting too much powerups. The robot has to decide wither it wants to have 5 powerups and keep all 5 powerup, or have 10 powerups and lose all 10 powerups. The robot will most likely pick the first choice.

Teachers will teach the robot how to manage multiple tasks. Do task1 first, stop, do task2, stop, continue on task1, stop, continue on task2, stop, continue on task1, stop, continue on task2, etc, etc. After switching back and forth, the robot has to make sure that it continues on the tasks until both tasks are completed.

Sometimes the robot can give priority to tasks. The robot can do task1 for 5 minutes, then do task2 for 1 minute, then do task3 for 2 minutes. In this case, task1 has high priority so the robot spends more time on it, while task2 has low priority so only 1 minute is devoted to that task (for each iteration).

Managing many tasks simultaneously is very difficult for human robots because the conscious is only aware of one thing at a given time. Human beings were taught to write things down or have a scheduler that manages multiple tasks. In a fast food restaurant, if a customer wants to order

10 items, the clerk has to punch out the order into a computer so that he can process the order himself or let others process the order (or both). During the order process, the clerk can check to make sure all of the items are processed.

People who work in a manufacture plant has to use computers or notebooks in order to manage multiple tasks.

If you think about the robot's brain and its conscious, intelligent pathways create a database system because the pathways can store data in a meaningful way and retrieve data in a meaningful way. The intelligent pathways can also create an operating system to manage multiple threads of tasks.

Examples of logical functions inside the conscious

Intelligent pathways in memory are structured in a hierarchical manner and are created by lessons taught in school. In some respects the data sensed from the environment has little concern in terms of the robot's decision making. The robot forms the conscious so that it can control its decision making regardless of the environment it is in. For example, if the robot was in a baseball game, he will not automatically play in the game. He might be an observer or a bystander who happens to past a baseball field. The goals and the roles of the robot is based on its decision making at a given time period.

The robot controls the situation. If the robot was at the baseball field as a player, then the robot's conscious will have the necessary knowledge to play baseball. If the robot was at the baseball field to observe the game, then the robot's conscious will tell itself what to do. In this case, the robot has to sit on the stands and watch the game.

Thus, based on the robot's decision making and its goals, it will take action; and not based on the current environment. In some respects the 5 sense data from the current environment is considered less important than the robot's goals.

This is the whole point about building a robot that has human level artificial intelligence. The robot is aware that it's alive and it controls its own actions in a dynamic environment.

The next couple of examples will illustrate the logical functions that are executed inside the conscious. As stated before, inside the conscious there is a series of "computer programs" that extract data from memory, process the data, and control the actions of the robot. The intelligent pathways create these computer programs. As the robot encounters the environment, in each iteration, it will select an optimal pathway from memory. This optimal pathway is the best pathway to follow to take action in the current environment. As the robot encounters the environment, the intelligent pathways create the necessary logical functions to process data and to instruct the robot to act intelligently in the future.

Some of the logical functions inside the conscious are extremely simple. They take specific data from the current environment and use simple discrete math functions to process the data. The remarkable thing is that the conscious (via intelligent pathways) extract very little data from the overwhelming sea of information from the environment and use this information to do logical thinking.

FIG. 14A-B are examples from the game space harrier. Should the character move to the top where there are lots of enemies to shoot or should the character move to the bottom where there are less enemies to shoot? The decision making is based on many other factors. For one thing, if I kill lots of enemies, I will have higher points and higher points mean that I can earn an extra life. However, there are rules based on this situation. When you encounter a lot of enemies, there is a high probability of getting hurt which causes lower life energy. When your life energy reaches 0 the player will lose a life (or game over).

The player has a choice to make in this situation. If the player's life energy is high and the player wants to beat the game, then go to the top and kill as much enemies as possible. On the other hand, if the player's life energy is low and the player wants to beat the game, then go to the bottom and destroy those enemies instead.

FIG. 14A

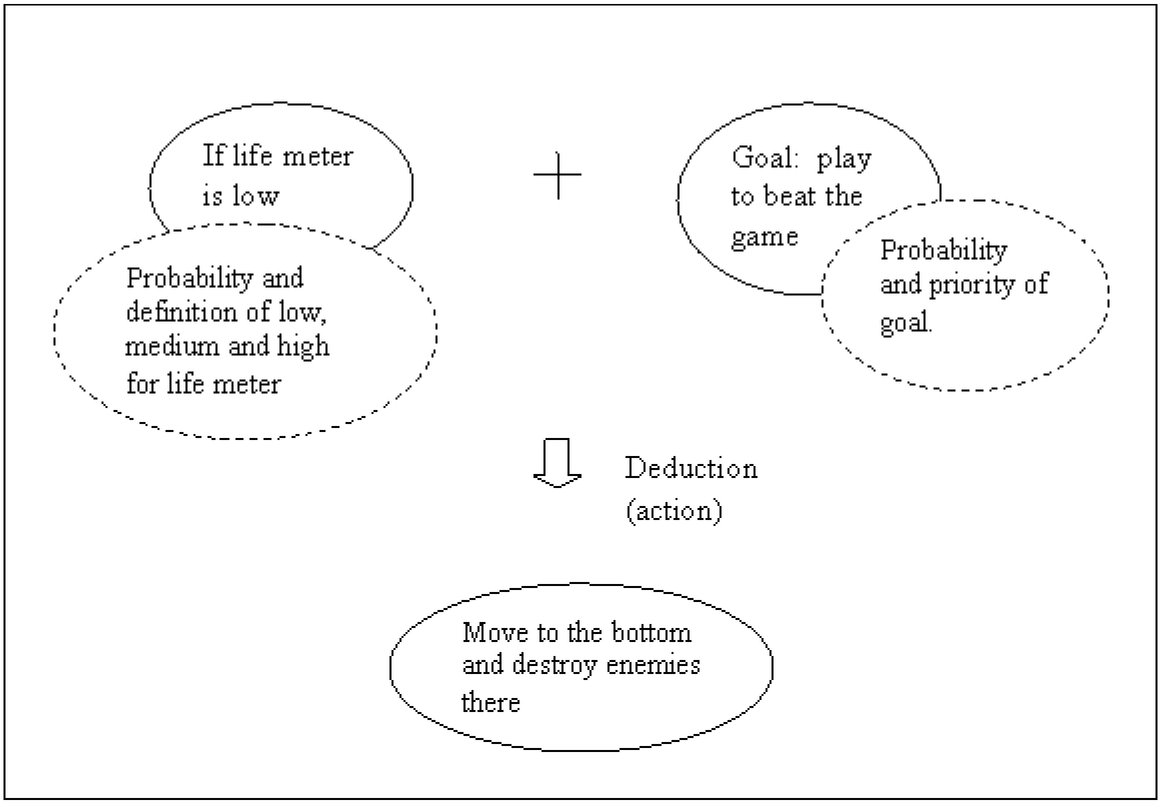
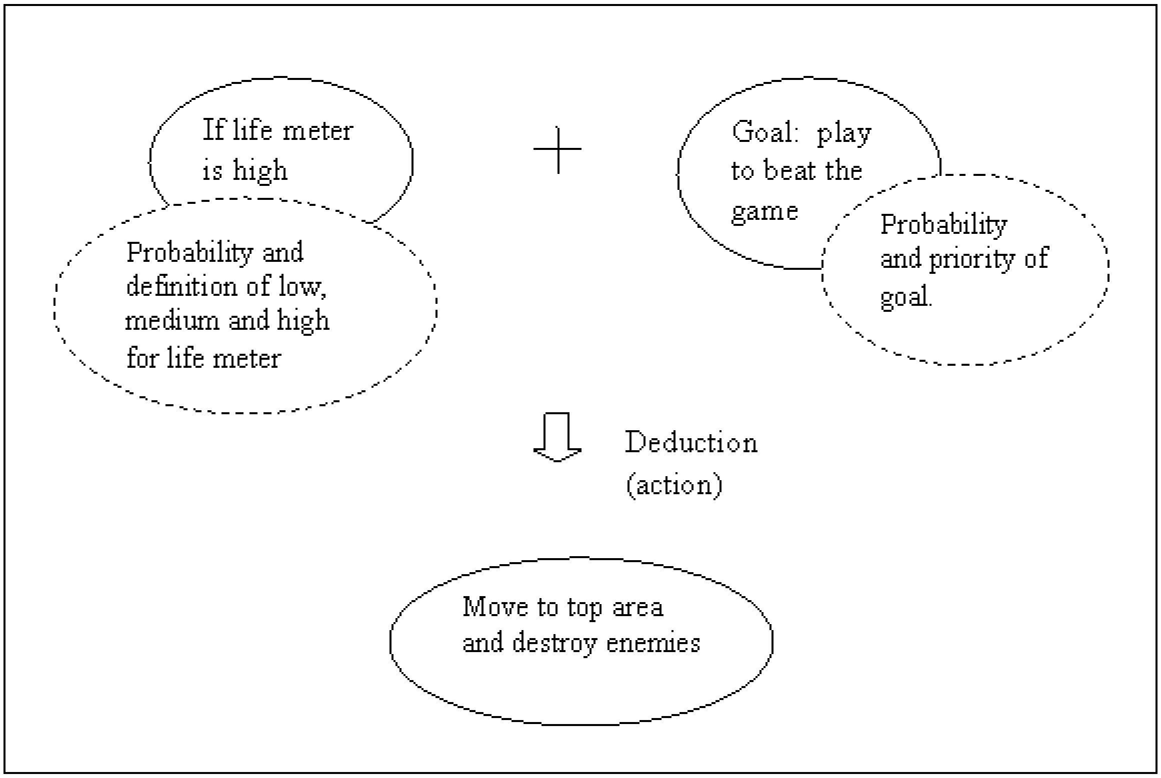


FIG. 14B



The 5 sense data coming into the robot every millisecond is overwhelming, but it is able to only extract the logical thoughts in FIG. 14A-B. It limited the overwhelming data from the environment to only a few elements. The decision making was based on a simple type of interconnected if-then statements.

In some ways it's not as simple as the diagram in FIG. 14A-B. There should be many factors involved in decision making (refer to previous books). The factor of determining how badly the robot wants to beat the game is important -- is he beating the game with high priority, medium priority or low priority. If he was in a contest, he would have the highest priority in terms of beating the game. If he was at home playing for fun, he would have low priority in terms of beating the game.

Another factor is the life meter. How does the robot determine what is low, medium or high? Many years of playing a videogame will determine this. The current game may not have the same standards as other videogames. The computer program inside the robot's conscious has to decide what is low, medium and high for this videogame.

As stated before, there are two types of conscious: open activation and hidden activation. The if-then statements might be considered open activation (solid circles) and the life meter and goal priority might be considered hidden activation (dotted circles). The robot is aware of the open activations, but isn't aware of the hidden activations.

The example in FIG. 14A-B is just one illustration for the space harrier situation. The goal might have 30 different variables. For example, the goal can be: play for fun, play poorly, play to laugh, play to lose and so forth. Teachers will teach the robot under numerous situations. The robot can also learn to teach itself how to make decisions through trial and error.

Layered tasks

When the robot is playing a given videogame, there are computer programs in the conscious that generate decision making and determine what actions to take. The conscious generates a computer program for each layered task. The robot might play a driving game and he might have 4-5 tasks that he is juggling simultaneously. One task is to drive between the 2 white lines; another task is to watch out for road signs; another task is to follow traffic laws; and another task is to avoid other cars or objects.

Each task can be static or linear. If the task is linear then the computer program will repeat itself over and over again. Driving between the white lines is a task that is continuous because the robot is always following this task. Following traffic lights is a linear (or repeated linear) task. The robot will stop the car on the traffic line, wait for the traffic light to turn green, and drive the car forward. Once the traffic light is past, the robot can stop the task and wait for another traffic light before executing the same task.

Each task is given a computer program and all computer programs inside the conscious have to work together in terms of changing the tasks and rules. All computer programs in the conscious have to be compatible with each other. In terms of decision making, how the robot's conscious manages multiple tasks is the actual decision making process. All layered tasks are managed and controlled in a linear fashion -- which tasks goes first, second, concurrently and so forth. The decision making process is actually the managing of multiple tasks in the task container. The conscious will delete, add or modify tasks in the task container as time passes.

Some layered tasks are complex, for example, a game like gradius for the super Nintendo, require players to manage powerups. The powerup meter has 6 levels and each level has a different powerup. For example, the first powerup is speed. When the player collects one powerup the first powerup meter will light up. If the player presses the C button the player will have faster speed. The 5th powerup level is the laser. If the player collects 5 powerups, the 5th level powerup level will light up. If the player presses the C button the player will have a laser weapon. The task of managing the powerup meter is done through a computer program in the conscious. This layered task has to interact with other layered tasks so that the robot can make decisions that benefit the videogame.

While the robot is trying to manage the task of the powerup meter, it has to also avoid any enemies or objects. The decision to pick a powerup level will depend on what environment the robot is in. If the robot is playing in the lava level, he needs the forcefield in order to pass the level. If the robot is playing in the forest level, he needs the laser weapon to destroy enemies easily. Thus, one layered task has to interact with other layered tasks in order for the robot to make a decision.

How does the robot actually play an unknown videogame?

What if the robot doesn't know how to play a videogame or what the objectives are or the rules are? The lessons in school of finding out what the objectives are and rules are will activate in the robot's mind. Things like: try pressing the buttons randomly to find out what the controls are or try something in the game to see what happens. Based on what happens, the robot can formulate what the character in the game has to do and what the rules are.

However, prior knowledge about videogames and what type of game it is will tell the robot what are the probable objectives and rules are. For example, if the robot was playing contra for the Nintendo, he will know by the first 5 seconds that this game is a side-scrolling action game. Side-scrolling games require that the player move forward and reach the end of the level. At the end of the level there will most likely be a boss waiting for the player.

While these general knowledge pop up in the robot's mind to play unknown videogames, basic type of trial and error functions are executed. Videogames can be played in terms of pain and pleasure. If the robot (player) is killed, the event will cause pain for the robot. If the robot passes a level or does something good in the game, then the event will cause pleasure.

Through trial and error, the robot learned many basic good or bad behaviors. If the character in the game has low life energy, that is pain. If the character is hit, that is pain. If the character loses a life, that is pain. If the character beats a boss, that is pleasure. If the character saves the princess, that is pleasure. Thus, intelligent pathways to identify wither an event in the game is bad or good, can be used to play an unknown game. Through trial and error, the robot will know what is good and what is bad in the game. Most likely the objectives and rules of a game are found through trial and error.

Identification of events for a videogame are learned from teachers (or through self-playing).

General types of identifications of events in a videogame include the knowledge:

1. do not touch enemies.

2. jump on the edge of the platform.

3. jump over bullets.

4. if an enemy have flashing colors, that means he is hurt.

5. aim for the bosses' head.

6. the game will tell you when a level is completed.

7. the character's life meter is somewhere on the screen.

8. the points of the game is usually at the top of the screen.

9. the life energy of the character is usually located at the top of the screen.

10. press the start button for additional information such as maps and items.

These are general facts related to all videogames. If you play any Nintendo game, you will find that all 10 facts listed above apply. Most of these rules are found by reading instruction manuals or videogame magazines. The robot might observe multiple videogames and come up with the knowledge that most games have a life energy at the top of the screen; or that when the life energy is gone the character loses a life.

These facts can also be learned by teachers. A teacher can teach a student how to play a videogame. He/she will point out to the robot facts about videogames. If the robot was playing double dragon, and he was on the bridge scene, he will press the jump button when he gets to the "edge" of the bridge. Sometimes, the robot will jump before the edge of the bridge. The result is landing in the water, which loses a life. This losing of a life will give the robot pain and he will find a new strategy through logic and remember this strategy in the future.

This method of finding a new strategy was taught by teachers. The robot will do something and if the robot fails, the teacher will ask the robot: "what did you do wrong". The answering of the question will give the robot a new strategy. The teacher will also tell the robot to remember this strategy in the future when it encounters the same problem. In the case of the double dragon game, the robot will go to the very edge of the bridge, then press the jump button. If it does this, the character in the game will reach the other side of the bridge safely and this will cause pleasure for the robot. The pathway to cross the bridge will be remembered, while other unsuccessful methods to cross the bridge will be forgotten.

Activating element objects using object association

The event of jumping at the edge of a bridge (or any platform for that matter) is strongly associated with the recognition of any platform. The next time the robot (player) encounters a platform where it has an edge and danger is located to the bottom of the platform, the lesson: "jump at the edge of the platform" will activate.

Essentially, if-then statements will be added into the event pool and the conscious will store the if-then statement as a rule. Once the robot encounters a platform that has danger below it, then the robot will activate the thought: "jump at the edge of the platform". The if-then statement that will be put into the event pool will look like this:

If recognize (platform with danger below), then activate thought "jump at the edge of the platform".

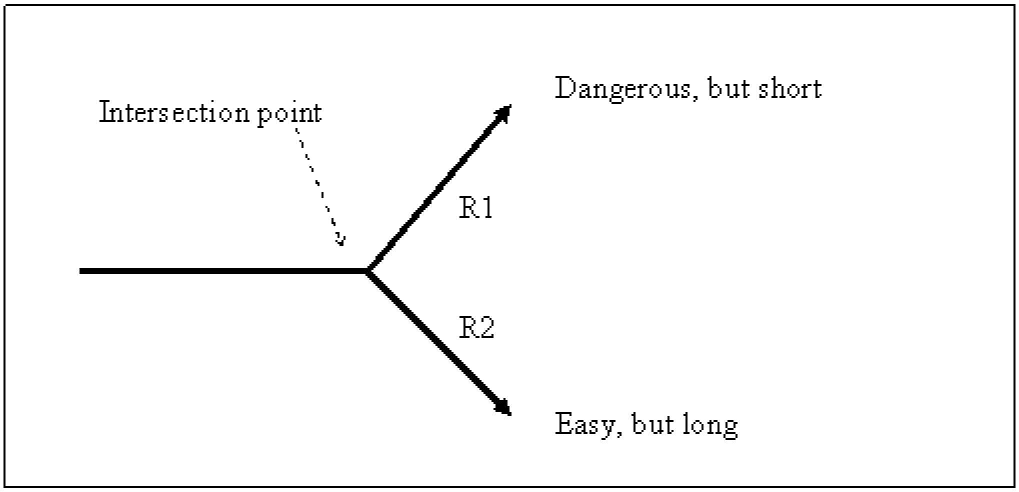
This if-then statement will be put into the event pool. (You can refer to my previous books to learn about the event pool). The event pool is just a container that the computer program in the conscious uses to tell the robot certain events will happen at certain times in the future. The event pool will also activate conscious thoughts at certain times based on a recognized event.

The event pool is important because it allows the conscious to manage simultaneous tasks easily. In my patent applications, I use driving a car as an example. The robot predicts the future and the event pool stores intelligent pathways (if-then statements, tasks, rules, encapsulated intelligent pathways etc.). These intelligent pathways reference when these tasks or rules or if-then statements will happen in predicted future pathways.

Activating intelligent pathways using object association

Referring to FIG. 15, an intelligent pathway such as decision making can be activated the moment when the robot recognizes an event. Imagine that the robot was playing a videogame and the character in the game has to make a decision to travel on 2 paths. The top path is dangerous, but short and the bottom path is easy, but long. The point where the two paths intersect is the event that should trigger a decision intelligent pathway. This decision pathway will use many facts from memory and the videogame to come up with the decision of which path to take.

FIG. 15



In the decision pathway, the robot will use facts such as:

1. If life energy is low take R2 path

2. If life energy is high take R1 path

Complex interconnected if-then statements can be used during the decision making process such as: if life energy is low and goal is to play for fun take R1 path.

The idea for this example is that when the robot encounters a similar type of situation for a different game, the same decision intelligent pathway will activate at that moment and the robot's conscious will generate the necessary computer programs to make a decision.

Thus, the robot's conscious can activate an intelligent pathway based on recognizing a specific event.

The robot's conscious can also activate an intelligent pathway based on logical thoughts. For example, in the legend of Zelda game, a logical thought might activate in the robot's mind. This logical thought will activate an intelligent pathway S2. Intelligent pathway S2 basically plans a route to get the character in the game from the current location to a destination location. Imagine that the character in the game activated a task to "get the silver sword". In order to get the silver sword the character has to go to a shop and buy the silver sword. Logic will activate intelligent pathway S2 to get the character from the current location to the shop.

A more general type of intelligent pathway can activate because the robot recognizes danger. Danger can be anything that threatens the character in the game. For example, jumping over the bridge is considered a danger or moving on a cliff is considered danger. In the moment the robot identify danger in the environment (which could be anything), the conscious thought: "be careful" activates. This "be careful" task will instruct the robot to pay more attention to the danger. New intelligent pathways will also activate to find a specific way of solving the danger. When the character in the game is confronted with jumping over the bridge, the "be careful" task will activate. When the character in the game is confronted with navigating through an asteroid belt, the "be careful" task will activate.

There are many other variations that activate the "be careful" task. The player might be fighting a boss and he repeatedly loses life energy. This repeated pain for the robot will trigger the "be careful" or "pay attention" task.

Chapter 6

Searching for data in the robot's brain

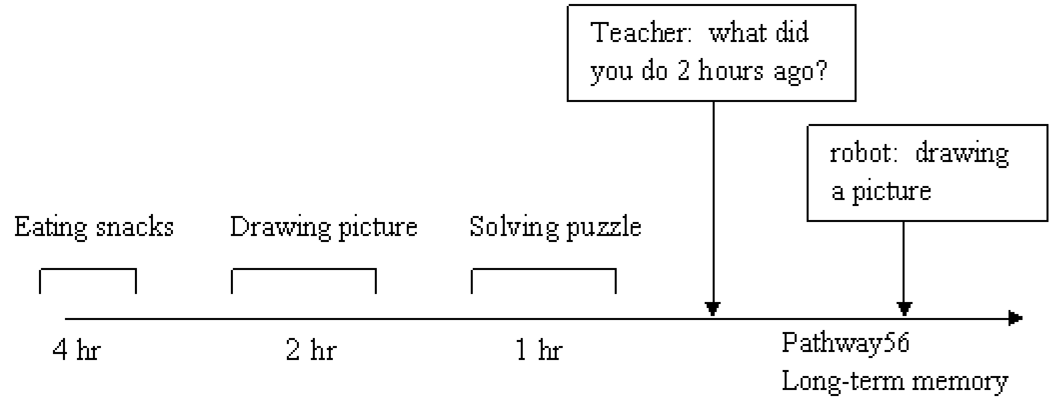
The search function for the brain is very basic and universal. The data in memory is stored in terms of association, however, each data can be same, similar or different. An entire network of data can be created for one object called an object composite. The object composite structures data in the network in a hierarchical manner so that important data is prominent and minor data is less prominent. The importance of data is determined by many factors and said data can be in any four data types: 5 sense objects, hidden objects, activated element objects and pattern objects. Usually, words and sentences are prominent data and long pathways and detailed images are minor data. The search function simply identifies object composites in memory (also called floaters) and extract information hierarchically, whereby prominent data is searched first before searching the minor data.

Intelligent pathways in memory have the necessary instructions to search in memory for data. The instructions in the search pathways are not learned by teachers, but they are based on patterns found from lessons taught by teachers. The teachers teach the robot a lesson and the robot's brain found patterns between the lesson and the retrieval and extraction of data in memory.

Example

FIG. 16 is a simple example illustrating how patterns are found between a lesson taught by teachers and the search instructions. The teacher asks the robot what it did 2 hours ago. The robot searches its long-term memory for data related to actions it took. The robot gives the answer: "I was drawing a picture". If you compare this example to many similar examples you will notice that the time of where in the long-term memory to extract information is same or similar. Within pathway56, the search instruction will search the long-term memory exactly 2 hours ago. It does not have to search the robot's memory, only long-term memory. By comparing many similar examples, the search pattern found out that it needs to extract data that happened approximately 2 hours ago.

FIG. 16



If the teacher asked another question such as: what did you do 1 day ago or 4 days ago or 1 year ago, the universal pathway will be able to understand where in the long-term memory to search to find information. Instead of searching all data in long-term memory or all data in the brain, the search pathways have patterns to search for data in a heuristic way.

The search pathways are learned by lessons taught in school and through pattern recognitions.

Questions such as: what was I doing earlier?, would require the robot to quickly scan the long-term memory closest to the current state to look for information. This question only relates to what the robot did today, possible a few hours ago. The search function will not search for data that happened yesterday or 1 month ago. Also, the search function is searching for prominent events that got the robot's attention. Simple events that the robot didn't pay attention to will be searched last (or ignored).

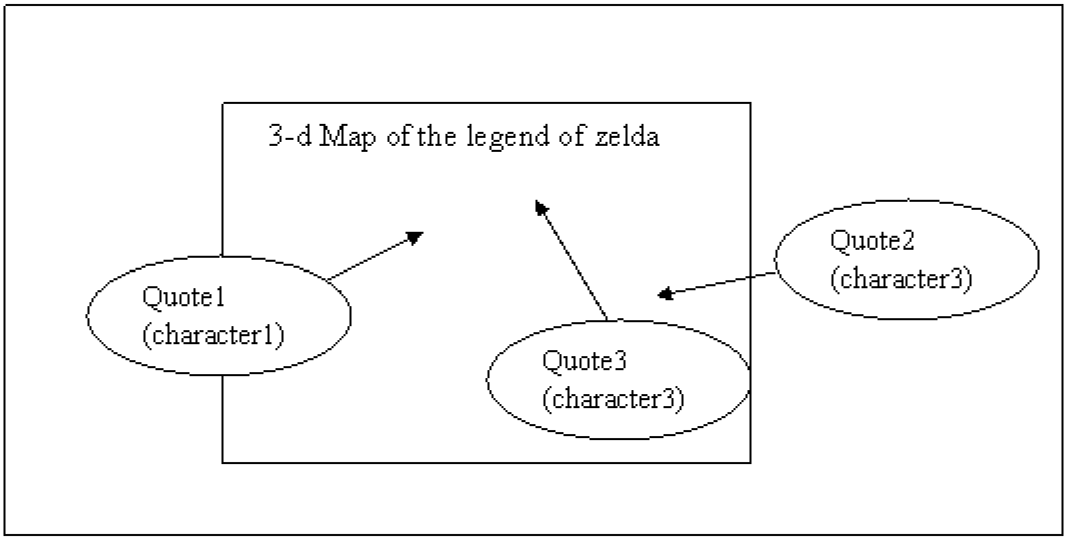
When the robot is studying and is preparing for a test, he will use conscious thoughts to remember important data. Intelligent pathways will identify important information in books. In fact, all data that the robot reads in will be stored in memory in an organized manner (via intelligent pathways). The way that the robot remembers important information is to control its focus. The robot might read in a fact and he will activate the thought: "remember this fact", the fact will be stored as strong data, both in the long-term memory and the brain of the robot. When the robot takes the test the very next day and questions are asked "related" to a fact, the fact will activate because the robot remembers that information.

The conscious has computer programs that organize data in the long-term memory. The data in long-term memory has reference pointers to data in memory (the robot's brain). Because data in long-term memory are new, those data are stronger than data in memory and can be remembered easily.

FIG. 17 is a diagram illustrating how data in long-term memory are organized through computer programs in the conscious. The robot has played the legend of Zelda for 3 hours; and during the game, the robot has talked to a dozen characters in Zelda land. All quotes from characters are important because they are the primary source of instructions to follow in order to beat the game. In the diagram, all quotes are organized by where they were encountered in a map (provided by the game) and visual sequences of encounters are stored in the map in terms of when they were encountered.

These types of quotes are summarized information that relate to a quote from a character. For example, a 2 minute speech by one character can be condensed to 1 sentence or 2 pictures. The robot will extract the information from this map and use intelligent pathways to analyze the data further. By having the conscious organize data in the long-term memory, the search function can find data very quickly. If the robot is stuck in the game and doesn't know what to do, it will first search important data in memory. The quotes from characters in the game will be searched first. For example, the robot might remember that one of the characters told him to search for a hidden cave in lake hylide. This memory will instruct the robot to go to lake hylide and search for a hidden cave. This hidden cave might have a character that will tell the robot his next mission.

FIG. 17

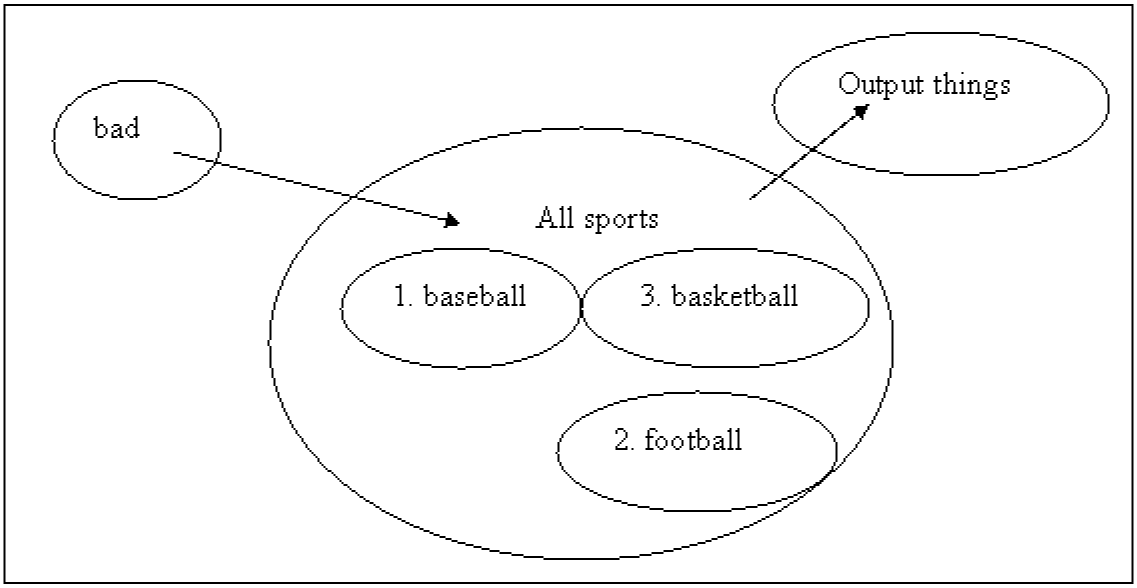


Controlling the variables the robot has to search in memory

When the robot is asked a question, an intelligent pathway can be used to identify important search variables. These search variables will then be used to search for data in memory. If the robot was asked a question such as: "what are the disadvantages of playing sports?", the search variables might be: bad, sports, output things.

FIG. 18 is a diagram of what the search variables should look like. The search variables are actually a fabricated diagram or movie sequence, whereby the variables are identified and the relationships of the variables to actions are established. Even the outputs of what to search for are outlined. In this case, the search function needs to output the "things" that are bad in terms of sports.

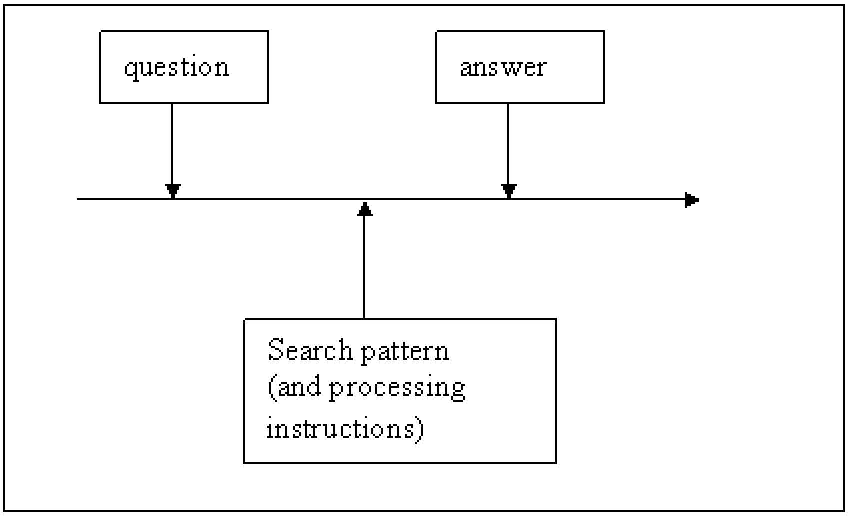
FIG. 18



Actually, the computer programs inside the conscious will do things in linear order. It will first search for the variables: bad, sports; and extract information about these two variables. A valve of information will flood the conscious and another intelligent pathway will be used to interpret the data and to extract specific information. The task of finding out "things" that are bad in sports require another intelligent pathway to analyze and extract information from the data in the conscious.

On second thought, questions and answers are learned in school and each type of question has a unique search function. It's up to the robot to learn how to answer "all" questions in school. The searching and processing instructions to answer a question is based on finding patterns between very similar examples. For example, the question: "what are the disadvantages of playing sports?" require many examples stored in memory. The teachers have to guide the robot to say the right answers for that question. The robot's brain will do all the hard work of comparing similar examples and to come up with an optimal way to search for the right answer for that question. FIG. 19 shows that when the search pattern is found, it will be stored right after the question. The next time the robot is asked a question, the unique search pattern will be used to output the correct answer.

FIG. 19



All questions and answers are learned through fuzzy logic. This means the robot can answer similar questions from the ones learned in memory.

Controlling activation of facts about target objects

In conventional grammar software, words are parsed using a language parser. Each word is assigned to their fixed definitions. For example, the text "bat" is a noun or the text "run" is a verb. Any intelligent program that uses grammar rules and language parsers are not self-learning. They require computer scientists to input information.

In my human robot (an AI program) there are no language parsers or grammar rules. Words that are recognized by the robot do not go through a language parser.

When the robot recognizes a text such as "bat" the robot will activate the meaning of the word. A picture of a bat might appear or the sound of the text might activate. All element objects (meaning) for the target object (text of bat) will compete to activate. The text word of "bat" has other meanings as well. The text word "bat" is also a noun. The question is how does the robot activate the correct facts related to a target object?

The conscious is the answer. The conscious controls the activation of element objects related to a target object. If the robot was doing a grammar worksheet and the instructions are to identify nouns and verbs, when the robot reads in the text "bat", the conscious will extract the fact "noun" from memory and assign that noun to the text "bat".

On the other hand, if the robot was doing an identification worksheet and the instructions are to assign visual images to texts, when the robot reads in the text "bat", the image of a bat will activate. The robot will compare the image activated to the image on the worksheet and determine that they are similar.

Thus, based on a problem, the conscious has computer programs set up to extract specific element objects from recognized target objects. This information will be used to solve a given problem. The computer programs in the robot's conscious are actually searching for the target objects in memory and extracting element objects in a heuristic manner.

It really depends on what problem the robot wants to solve. If the robot was doing a grammar worksheet, then it will activate grammar rules related to texts it recognizes. The robot's mind will simply activate grammar facts about the text it reads. If the robot was a teacher and is correcting student essays, then a computer program in the robot's conscious will specifically cater to identifying texts in terms of grammar, identifying sentence/paragraph errors, identifying misspelling or wrong words to use in sentences and so forth. The computer program will know that its goals are to grade essays and its job is to find errors in essays. Lessons learned in grammar class will activate and managed by the robot's conscious.

Another example would be reading Shakespeare books like Hamlet. A more analytical pathway must be used to read text from Shakespeare. A more logical form of intelligence is needed to understand what sentences mean and how multiple scenes relate to each other. The meaning to sentences and paragraphs by Shakespeare go beyond what is presented in the text. He uses artistic expressions that make his work a masterpiece.

If the robot was reading text from a novel, he doesn't have to worry about grammar rules or deep understanding of the text. All he needs to do is read the book and create a fabricated movie that explains, in a visual way, what the book is about.

Teachers teach the robot what to do for a specific task. If the robot's task is to grade student papers, the teacher will teach the robot to identify grammar errors, misspellings, content of paper and so forth. Thus, the next time the robot has the task of grading papers, he will use grammar rules, check grammar errors, and everything English teachers taught the robot about grading essay papers. On the other hand, if the robot was reading a novel, the teachers would teach him to visualize the texts or teach him nothing at all. The teacher didn't ask the robot to pay attention to grammar structure or identify sentence types. It's natural at this point for the robot to fabricate a movie, while reading a novel. However, hierarchical activation or random activation might tell the robot that this word is a noun and this word is a verb, etc. For the most part, reading any material with text, will activate English lessons because of association. For specific tasks, specific English lessons will activate.

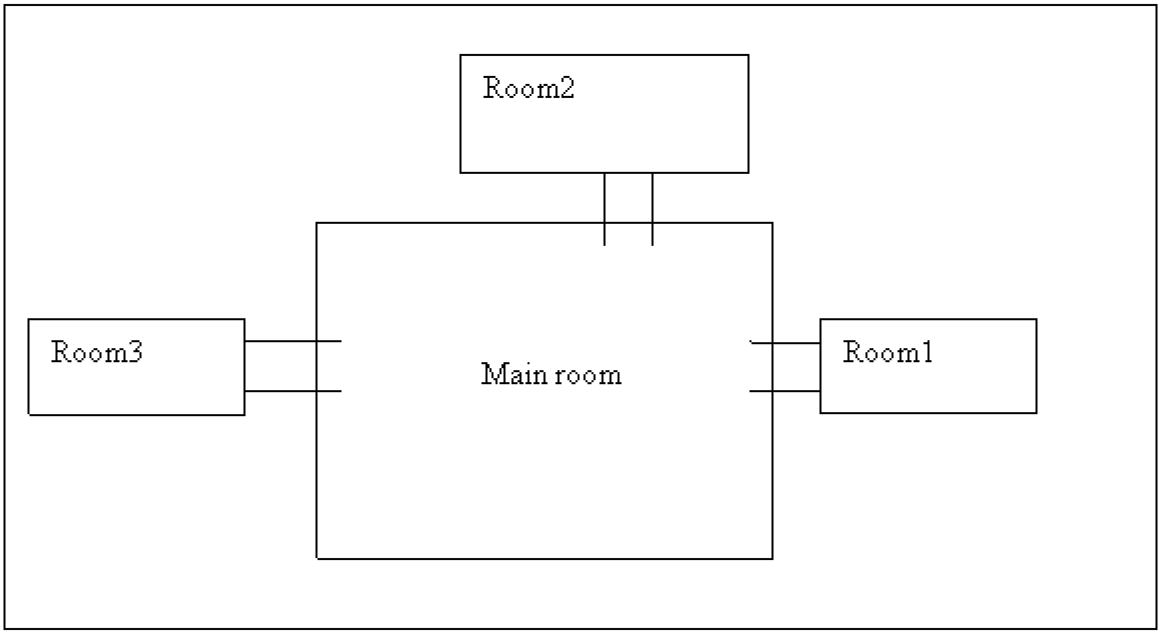
More advance novels like Hamlet or the scarlet letter will activate lessons in advance English classes, whereby the robot has to analyze and logically come up with facts that are not described in the text.

The conscious can generate a map of the environment

When the robot plays a game like the legend of Zelda, it is very important that the robot have a map of the environment. In the game, sometimes maps are given to the player and other times maps are absent. In the event maps are absent, the player has to use temporary memory to remember where he has gone in the past and what the environment probable looks like.

Some dungeons in the game resemble complex mazes. Sometimes, the player has to enter rooms and remember what these rooms look like. For example, the player can have the option of entering 3 rooms (FIG. 20). The player enters room1 and found out it leads to a dead end. The player goes back to the main room and enters room2. Room2 is another dead end, so the player goes back to the main room and enters room3 where he found the level boss.

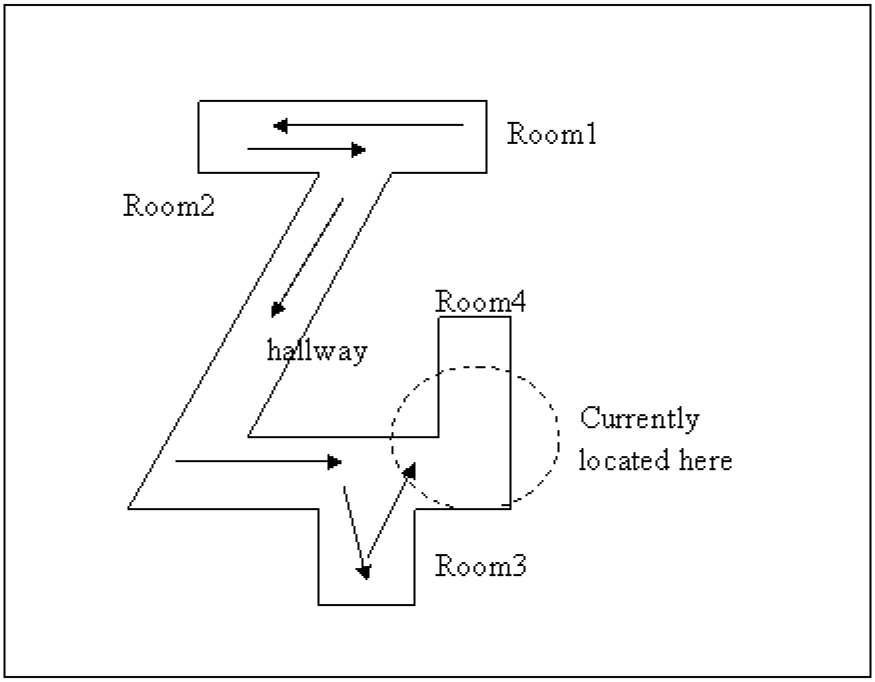
FIG. 20



The robot's brain has to remember what is in each room and to remember where it has gone in the past. The computer program in the conscious will generate a temporary 3-d map of the environment, based on a short past, of where it has gone and what information was stored in each room. This 3-d map will be used for logical purposes to make future decisions.

This 3-d map is very basic and the robot can only remember a short area of the environment. Referring to FIG. 20, only several rooms and their locations to each other are presented in the 3-d map. If the environment is too complex the robot will be confused in terms of what the environment looks like. This is one of the reasons why a human being will get lost if they are put into a complex maze.

FIG. 21



As the robot encounters more rooms the old map is forgotten, and a new up-dated map will be generated. By the way, the map in FIG. 21 is a map that was learned through example. In a past gameplay of Zelda, the robot goes through several rooms (called movie sequence1) and then presses the start button, which shows a map (called map1). If the robot learns many various examples of movie sequence1 and map1, then a pattern begins to emerge.

In the future, when the robot encounters a situation similar to movie sequence1, then a fabricated map1 will activate in the conscious. The fabricated 3-d map will look similar to map1.

In addition to the maps that are generated by the robot's conscious, fabricated maps can be created from data in memory. A fabricated movie of lines drawn on a surface will show the robot where it is and where did it go in the last several minutes (or hours).

Also, maps that are seen in the past and remembered can also activate to help the robot navigate its surroundings. For example, if the robot learned all the streets and cities of Hawaii, then it can activate a map of Hawaii; and come up with logical outputs. Taxi drivers have a very good map of the environment because it's part of their job – to drive people around the city. When the robot is driving a car and needs to go somewhere, a previously learned map will activate which will tell the robot where it's currently at and where its destination location should be. The robot will use intelligent pathways to draw lines on this remembered map to plot a route to its destination location.

Fabricated maps can be general (comprising lines and circles) or detailed (comprising remembered city maps). A general map of a country can be created or a detailed map of a small area in the country can be created. The intelligent pathways generate these fabricated maps. If the map is big like the United States, the mind can travel from one state to the next using animation zooming (zoom in or out). Maps can be traveled using car speed, human walking speed or plane speed. The robot's conscious can even draw lines and diagrams on the fabricate map to indicate a function. Traveling on states can be represented by animated or static arrows. The fabricated maps can also convert from general to detail at any given moment.

These types of intelligent pathway to manipulate and fabricate maps are based on the robot visually experiencing these behaviors in real life. For example, your local news station shows the United States map and the camera zooms in to Nebraska. This animation is learned by the robot and intelligent pathways mimic its behaviors.

Chapter 7

Static data and linear data in movie sequences

In pathways (or movie sequences), there are static data and linear data. Static data is considered the "existence" of an object and linear data is considered the linear steps of an object, event or action. One example of static data is a TV guide. Each week, my local newspaper would have 1 week worth of TV schedules. The TV guide is static data because shows on the TV guide are fixed for one week. Another example of static data is a person's medical form. The form depicts background information about the person such as name, phone number, age, gender, occupation and so forth.

These static data can change over time. For example, the person's medical form can change some of its slots, such as the age of the person or his/her occupation. A TV guide, on the other hand, will change every week. The TV shows for last week is different from the TV shows this week. The movie sequence record static data by the existence of that object. If the robot was looking at a form for 10 minutes, the form is existing in the 10 minute movie.

In the case of a 4 page form, the robot is recording the 4 pages in memory in terms of the 3-d representation of the form. The robot will record, in his memory that he has flipped through 4 sequential pages of a form. Sometimes, the robot will flip the forms backwards or skip pages. The important idea is that the robot can store each page in their respective 3-d location in memory regardless of how the robot flips through the pages of the form. In other cases, the robot can be confused and forget which page goes first, second or last because of the chaotic way the pages are looked at.

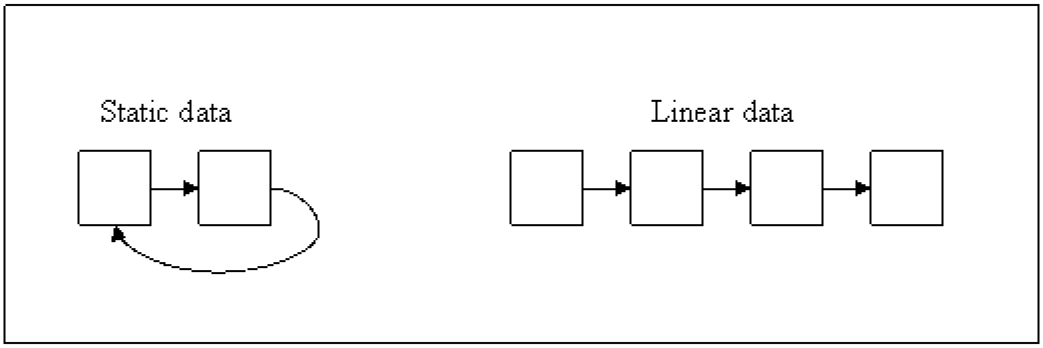
Another factor is the linear way text is read from the pages in 3-d space. The recognition of text is stored in the page, where it was read (in 3-d space). Reading text on a page requires zooming in on the page. If the robot looks at the full page, the static data won't record any text, just garbles of lines and simple color. Only when the robot zooms in and id the text will the text be stored in memory.

Linear data, on the other hand, records sequence of events. The ABC block problem is a good example because it requires linear steps in order to solve. Solving a math equation, like addition or subtraction also include linear steps.

FIG. 22 depicts a diagram that shows how static data and linear data are stored. The static data records the robot looking at the existence of an object. The data in memory loops itself and simply make an object's powerpoints stronger. The object could be a medical form and the robot is storing the medical form in memory (as the texts are read in). The first couple of frames might record the top half of the medical form and the next couple of frames might record the bottom half of the form. Once the entire form is read, the form is stored in memory; and the robot's brain will make the existing data on the form stronger if it encounters it in the future. Data on the form will forget as time passes. If the robot reads the form again, the slots in the form will be stronger in memory.

The linear data will record how objects animate themselves. When a block is moved from one location and another block is moved to take its place, then the movie sequences recorded the animation of replacing a block.

FIG. 22



Pathways in memory use both static data and linear data to store information. If you read a science book, you will notice a lot of pictures and diagrams illustrating how procedures or systems work. Some of the diagrams depict a fixed machine and the text in the book describes what each component of the machine does and how they work. Facts about certain areas of the machine are described.

Linear data and static data are interconnected in terms of learning knowledge. They give the robot a wide range of "capabilities" to store data. For example, if the robot was reading a science book on how a neural network works, a diagram of a neural network might be large and the diagram might have many small elemental parts. The robot see the diagram in fragmented areas and moves its eyes to focus on elemental parts or the overall diagram. This behavior will ultimately form a static data on the full diagram in memory, as well as, the focus and movement of seeing the elemental parts.

This is important because when the robot activates the diagram in the future to answer questions, the mind can move freely in any direction in the diagram and focus on any elemental part of the diagram. There are primarily three elemental parts to a neural network: input layer, hidden layer, and output layer. If the question asked was to describe what the input layer is, then the robot has to extract a diagram of a neural network, focus on the input layer, and activate facts related to the input layer. Using this knowledge it can answer the question. Another method is to extract the diagram of a neural network, focus on the input layer, use an intelligent pathway from memory to analyze the properties of the input layer, and output an answer.

Even in a medical form, the robot has to focus on small areas of the form like the slots and x-ray pictures and so forth. If the robot wanted to retrieve the patient's name, he has to activate the medical form in his brain and focus on the name slot. If he wanted the occupation of the patient, he has to focus on the occupation slot.

In most cases, when the robot reads a text form, the recognition of the words will activate visual images (or other 5 sense data) and that visual image will be stored next to the word that activated it. For example, when the robot reads a patient's medical form and focus on the patient's occupation, the image of the occupation will activate. If the occupation was a garbage worker, an image of a garbage truck will be stored near the occupation slot. When the robot recalls the medical form in the future, and when he focuses on the occupation slot, he will see a garbage truck and he will know that the patient's occupation is a garbage worker. By the way, when the garbage truck image activates in the robot's mind, his brain selects an intelligent pathway to interpret the image. The intelligent pathway will analyze the image and assume the patient's occupation.

Either the recalled memory of the medical form contains sound data (the sound of the word/s) or it contains a visual picture. Rarely, does the medical form (or any text form) contain visual text.

Learning, configuring and storing data

To better understand how pathways learn knowledge, I decided to use five examples: 1. reading novels. 2. reading comic books. 3. reading history books. 4. reading science books. 5. learning science lessons from teachers. These five examples show how the robot can learn knowledge in books and to store this knowledge in an organized way in memory.

1. Reading novels

When reading any book, the robot fabricates a make believe movie in memory based on the text read. This fabricated movie is generated to give a visual idea of what the story is about. This fabricated movie is based on activation of the meaning to sequential text. The robot creates a make believe movie (based on the 5 senses) to visually see what the story is about.

This fabricated movie is very complex because it contains interconnected objects, events and actions. Relational links between objects are established. Let's use a popular book like the lion the witch and the wardrobe. The objects involved are: the characters, scenes and the narrator. During the reading of the book, every word read by the robot will activate a primitive movie that explains the happenings in the story.

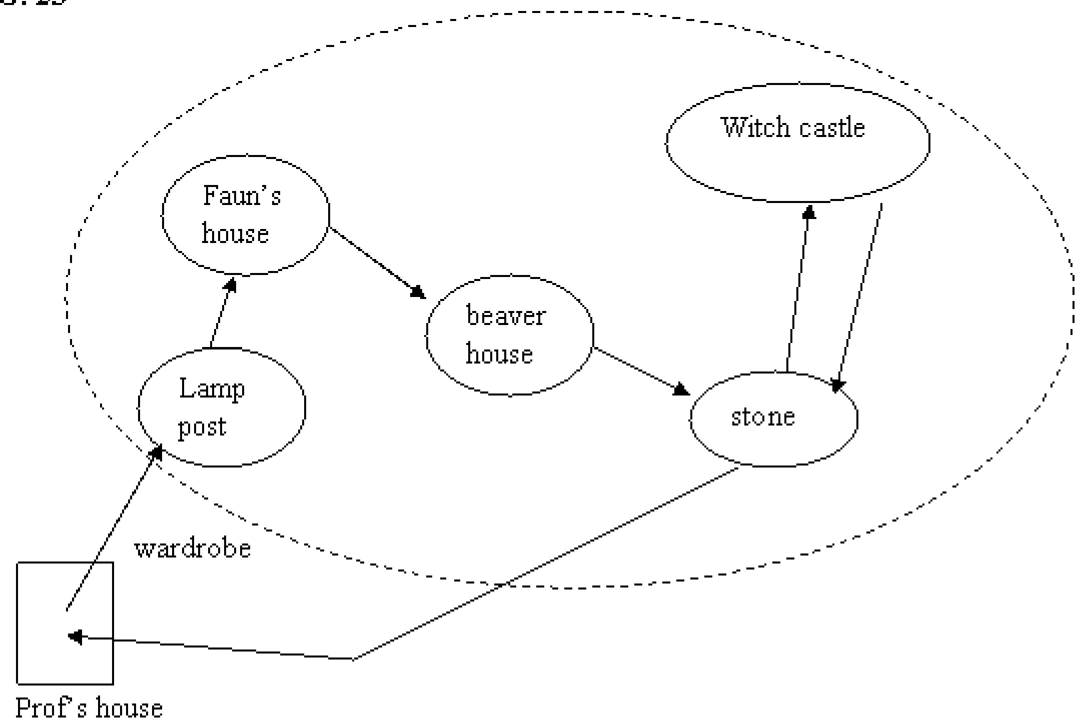
The robot's brain stores data in a 3-d manner. The visual environment from the story is primarily where the events in the story will be stored in memory. For example, at the beginning of the book, the 4 children are located in the professor's house. The robot will visualize what the house look like. Imagine that the first few pages describe the children in the kitchen talking and eating. The robot will fabricate a visual environment of a kitchen and what the characters should look like in the story. In the next couple of pages, the children plays hide and seek in the top floors of the house (the kitchen is located in the bottom floor). The robot will visualize the structure of the house and designate the kitchen is in the bottom floor and the rooms are in the top floors. A recording of a timeline will also be presented, whereby the children was in the kitchen and then they went to the rooms.

The words in the text describe how the characters look and what their personalities are. This information gives the robot a composite of each character. Things that are not in the book will be assumed by the robot. For example, if the detailed descriptions of the characters are not given and only a general idea, the robot can assume what the detailed description might be. If the author simply wrote that Lucy is a little 7 year old girl, the robot will add things like blonde hair (the story takes place in England), wears a pink dress and have one missing tooth.

If Edmund was described as a grouchy stupid teenager, the robot can make up a visual image of what Edmund might look like based on personal past experiences. Teenagers living in the 1800's wear mandatory school clothing; and that might be the image the robot fabricated for Edmund.

FIG. 23 is a diagram depicting the scene of events in the story. In the fabricated movie of the book, the scenes are outlined and their locations in 3-d space are just estimates. If you read books or watch movies, the authors do a very good job in setting boundaries of scenes. After reading the book, I was able to map out the sequential scenes in the book. Sometimes, the author revisits a scene. For example, the lion was killed on the stone hedge. At the end of the book, stone hedge was a place to celebrate their victory. In another example, the story begins in the house and ends in the house. Thus, repeated scenes in the book will be noted by the robot. He is able to tell what scenes were encountered, even if they are repeated, in a linear way.

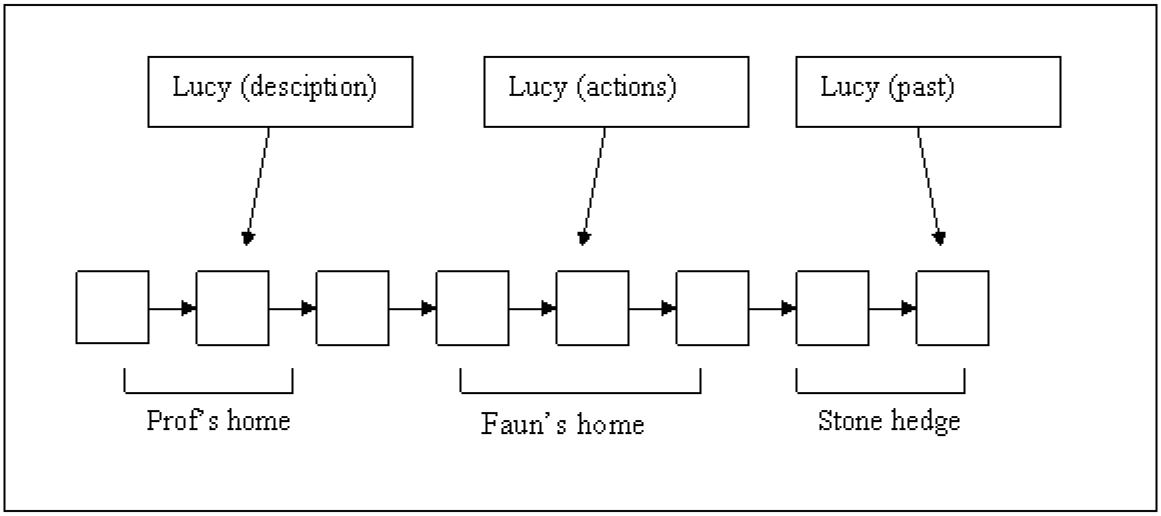
FIG. 23



The above example shows that the sequences of events in the story are stored in a 3-d environment. What about the characters and their actions? It all goes back to my original patent applications and books. The robot tags each character in the sequential movie with relational links. All actions and knowledge about a character is stored as one object floater (or a network of data). If the robot needs information about lucy, he can access the lucy floater. In this floater, all information that the robot has encountered on lucy will be there.

FIG. 24 is an example of how the lucy floater is created. Lucy is encountered 3 times in different scenes in the story. In the first scene, located in the prof's home, the author describes what lucy looks like and her personality. In the second scene, located in the faun's home, the author explains to the reader what lucy did. Finally, in the third scene, located in stone hedge, the author talks about lucy's past. All three scenes encountered gives new information about lucy. Since the fabricated movie is stored in a 3-d environment, this would mean that the object lucy is stored in multiple places (it doesn't have a fixed area to store lucy). However, even though lucy is stored in multiple places, there are reference pointers that id all lucy objects in the fabricated movie. When the robot wants to access information about lucy, he can locate the masternode (the lucy object with the strongest powerpoints) and this masternode has reference pointers to all other copies of lucy in memory. The information about lucy are considered the element objects and all element objects gathered will compete with one another to activate in the robot's mind from the target object lucy. This method shows that all data related to one object can be accessed in a global fashion.

FIG. 24



For fixed objects like environments, it is easy to assess that environment's facts/knowledge. However, moving objects like people and animate objects that move around a lot, need to access their information in a global fashion.

Why does the fabricated movie store things in a 3-d environment? Well, because the robot experience real life in a 3-d environment. It learned that the real world is 3-d and it applied that knowledge to a story in the book. In the first scene about the kitchen, the robot learned, from past experiences, that he has to move from the kitchen and climb the stairs to get to the second floor's rooms. The characters in the story are moving from the kitchen to the second floor. Thus, the 3-d environment learned in the real world is being applied to the 3-d environment in the story.

3-d encapsulate 2-d and 1-d. The robot can also understand 2-d data like a song. The robot can also understand 1-d data like a picture. If the character in the story picks up a picture and talks about the picture, the robot knows that the picture is a 1-d object. It doesn't move and the humans in the picture are not animated.

2. Reading comic books

It is slightly different when reading a novel compared to a comic book. In fact, it's a totally different experience. In a comic book, there are pictures and captions to read. Rules have to be followed in order to read a comic book (I discussed these rules in previous books). Captions come from characters in the book as well as the narrator, who is telling the story.

There is no need to fabricate a "full" movie of the story in the comic book because the pictures of events are given on the pages. In each page, panels are used to show snapshots of scenes. The panels are arranged from the top-left to the bottom-right. There are some exceptions, such as double page spreads or side pages. The pages in a comic book are static data – they don't move because they are still pictures.

Intelligent pathways from the robot will guide the reading of the comic book. These intelligent pathways will also determine what captions belong to what characters. Sometimes, logic is needed in order to understand who is speaking. For the most part, captions have arrows that point to the character speaking the words. All the robot has to do is look at the arrow and assign the caption to a character.

There are two types of pathways that are being generated simultaneously by the conscious of the robot. The current pathway is the experience of reading the comic book. The pages of a comic book are static data and the robot focuses on panels on a page – linking these panels in a sequential manner. One page of the comic book is stored in memory as one static data. The sequence of panels are referencing where each panel was encountered in the full page. By the time the robot see all the panels in page3, memory will have page3 stored as a static data. The robot's long-term memory will store the linear panels encountered in page3.

While this is happening, the robot identifies objects in the comic book pages. Characters, events, actions and narrators are recognized. Each object will be tagged with their learned groups. For example, if the robot was reading an X-men comic book, and he recognizes wolverine, then the learned word wolverine will be tagged to the character. If the robot recognizes an action, such as wolverine jumping, then the event will be tagged with the learned words: "wolverine jumping". Thus, the robot's conscious provides identification of objects, events and actions.

In addition to identifications of objects, events and actions, the conscious also provide facts and information on these objects. Logical facts about objects will also be stored in the current pathway.

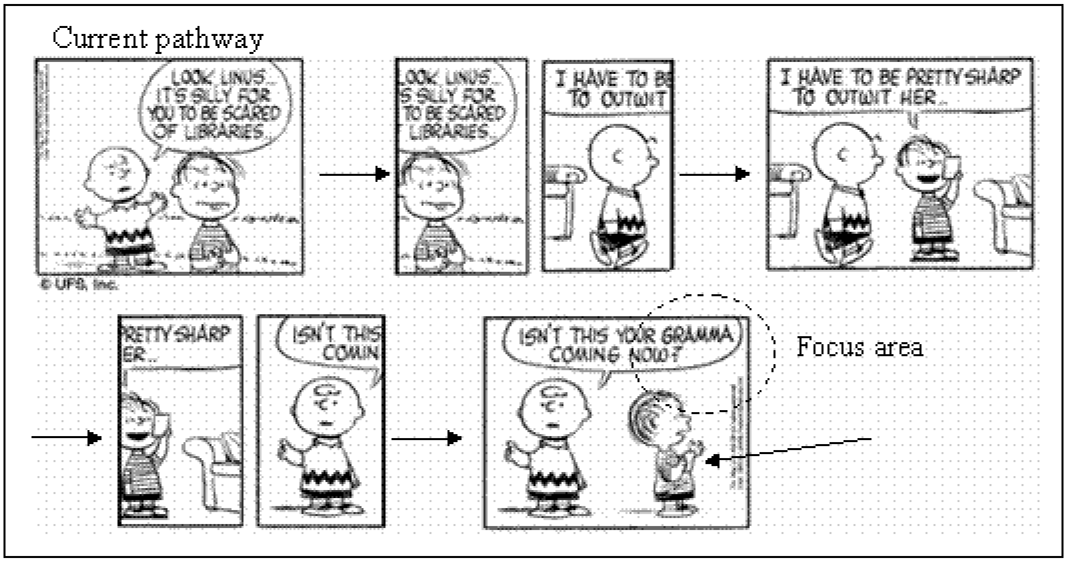
The current pathway and its conscious thoughts are one type of pathway. The next type of pathway is the fabricated movie. When a caption is read, the meaning of the words establishes relational links with the characters in the story. The words might give information about characters or provide a past event. Some of these words have nothing to do with the pictures on the comic book. For example, one caption is telling the reader what a character in the comic book is thinking of. A character5 might be sad in one panel and his thoughts are: he lost a good friend in the past. The fabricated movie created a movie that shows character5 watching a friend die or character5 looking at the grave of a friend. This fabricated movie provides visual meaning to the captions.

These fabricated movies will be stored right next to the caption that the robot encountered. This fabricated movie serves as separate data from the current pathway. However, this data will co-exist with the current pathway in the same memory area. FIG. 25 is a diagram showing a comic book page. For simplicity purposes, only three panels will be illustrated.

FIG. 25

© UFS, Inc.

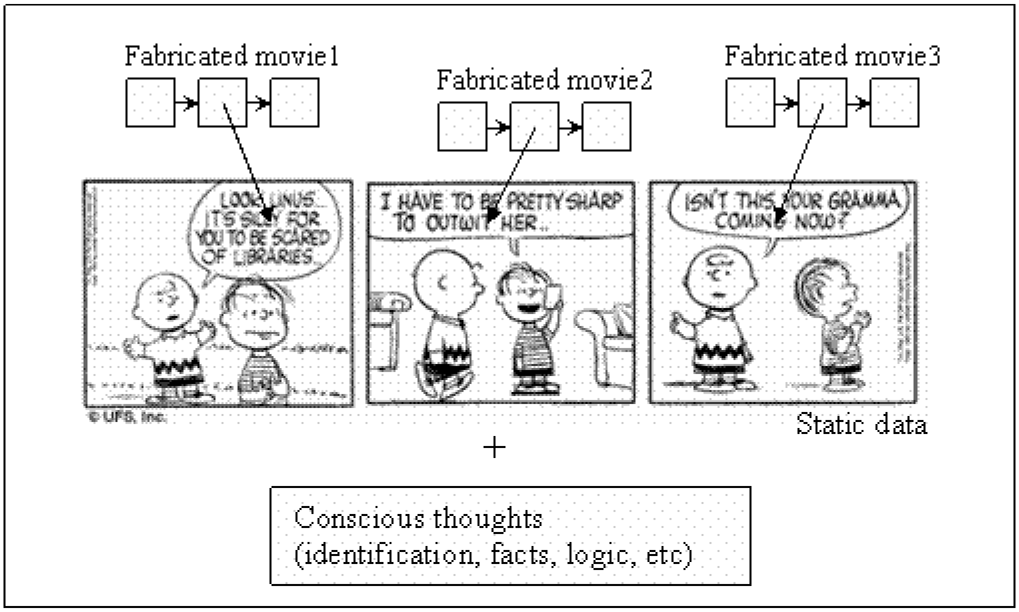
FIG. 26



In FIG. 26, the current pathway records the robot's reading of the comic book. In each frame, the robot has a focused area and a peripheral area (the arrow is pointing at the peripheral area). The robot will be looking at the images in the panels and reading the captions. The robot learned that it has to look at the first panel, finish looking at the images and reading the captions, then go to the second panel, finish looking at the images and reading the captions, finally go to the third panel, finish looking at the images and reading the captions.

Although in the current pathway, the robot is looking at different sections of the comic book, memory will only be storing a static data on the comic book. The static data will look like FIG. 27. As the robot reads the comic book linearly, the static data is formed in memory. The diagram also shows that a fabricated movie is stored right next to the captions. These fabricated movies will have relations to the robot's current pathway and its conscious thoughts. This means that objects in the comic book will have multiple relations to each other based on the current pathway, the conscious thoughts, and the fabricated movie.

FIG. 27



With all this information stored in memory (the current pathway, the conscious thoughts and the fabricated movies), the robot has a clear understanding of the comic book. If someone asks the robot a question about what happened in the comic book, the robot will be able to recall the comic book and give specific information.

Another interesting thing about the fabricated movies is that it can take a still picture from the comic book and create a short animation of that picture. For example, in the second panel, Charlie brown is walking. The conscious of the robot can fabricate a short animation of Charlie brown walking based on one still picture. This gives the robot a more detailed recollection of what is happening.

Imagine if there were two panels, in the first panel Charlie brown is standing on top of a table and the second panel shows Charlie brown on the floor crying in pain. The robot can fabricate a short animation of Charlie brown standing on the table, falling, and landing on the ground. The frames in this fabricated movie will be very close to the images in the panels. For example, Charlie brown will look like Charlie brown and not another character. The table will look like that table in the comic book and not a table in real life.

This is very important because the robot's conscious is trying to fill in any missing data from the comic book. Let's say there were two sequential panels. In the first panel there is a character1's face and he is talking. On the second panel is a character2's face and he is talking. The panels do not say that character1 is facing character2 and vice versa. Because of intelligence and because we fabricate that a character1 is facing character2, we are able to "know" that they are facing each other in a 3-d manner. The fabricated movie of these two panels will include a simple illustration of character1 and character2 in a 3-d environment that show them facing each other. This fabricated movie is just a basic illustration (comprising of lines and arrows and cartoon animation). Maybe, when the robot looks at the first panel, the robot will draw a line with an arrow outward towards something. The text in the captions will say who the arrow will point to (mainly character2). Maybe, the robot's conscious will show a 3-d grid with simple stick figures to represent character1 and character2. It really depends on the lessons learned in school.

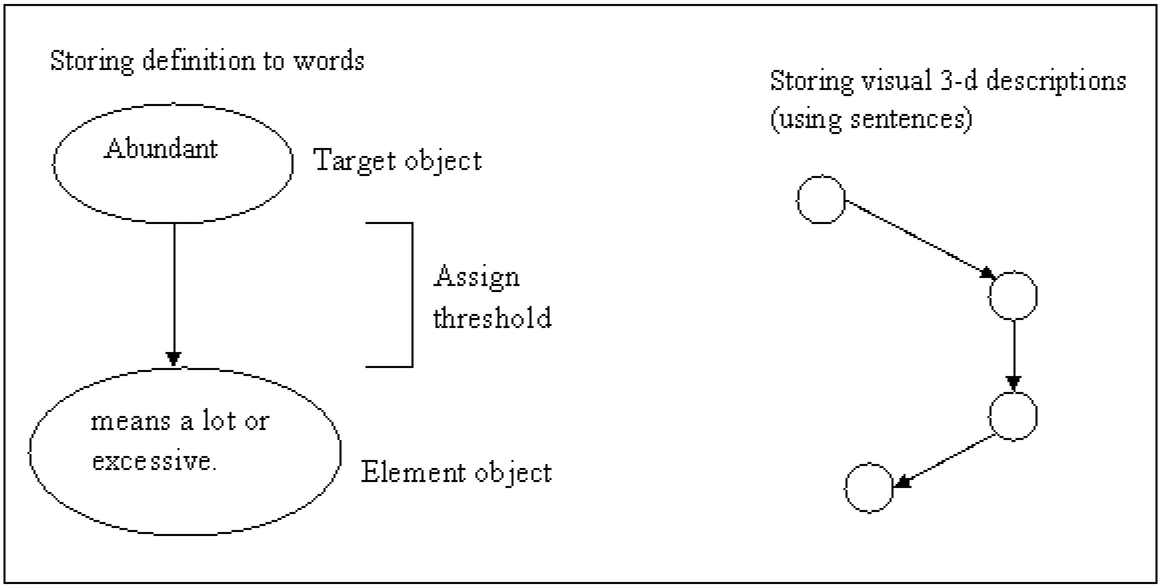
3. Reading history books

History books contain facts/knowledge about different subject matters. How does the robot's brain store this knowledge in memory? Sentences can encapsulate many form of information. Sentences can describe an object, explain a definition, answer a question, give facts about an object, predict the future, convey a message from an object, explain personal views, convey humor, and so forth.

In some respects different type of sentences should have different ways of storing data. For example, defining the definition of a word should be different from explaining a visual environment (in terms of storage of data).

FIG. 28 shows how a definition of a word is stored in memory compared to explaining a visual environment. When storing definition of a word, the word is the target object and the meaning is the element object. The robot's brain should store the word close to its meaning because they are equal. On the other hand, if sentences are used to describe a 3-d environment, the robot's brain will generate the visual aspects of the sentences and store them in a 3-d manner. These two examples show that different sentences store data differently. It is up to the robot to average out lessons learned in school and to find the patterns between storing data and the different types of sentence structures.

FIG. 28



If you look at medical books, they will show you visually the different parts of a human being. These parts are encapsulated in a human object. The head, or hand, or leg is a part of a human being. When facts are given about an object, these facts will be stored in the visual representation of that object. For example, if the object was a heart, and the robot was learning about facts related to the heart, that knowledge will be stored in the visual heart. The heart is stored in the human being. Knowledge about the eyes will be stored in the visual eye area, knowledge about the hand will be stored in the visual hand area and so forth. This is a very powerful way of organizing knowledge. Visually, the human body is very complex, but the brain is able to learn where certain body parts are located in the human body and store related information in their respective visual area. The more knowledge learned about a body part, the more organized the knowledge for that body part is.

There are some facts that store the interaction of two or more body parts. These types of facts can be stored in any object involved (usually the strongest object). But the area that the fact is stored in should have reference pointers to all objects involved. This is how a minor object involved in the fact can activate the minor object.

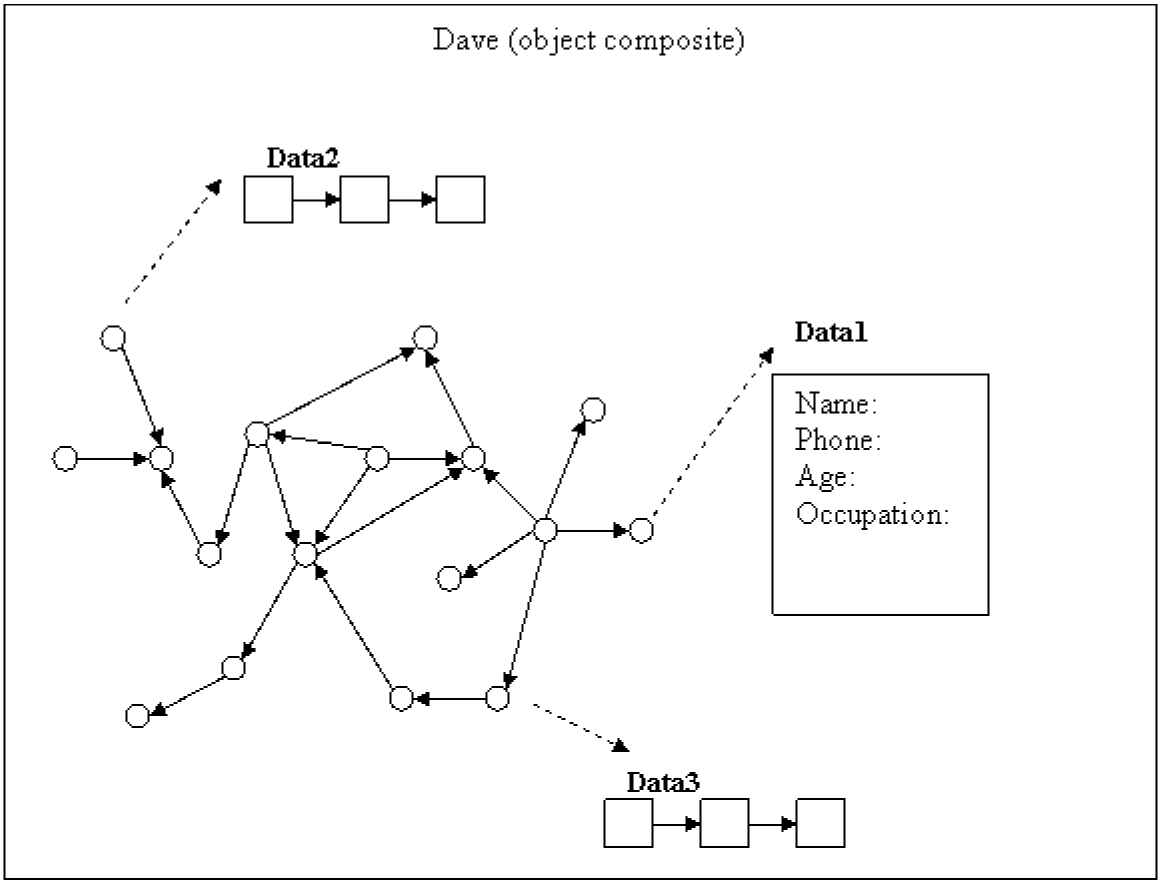
Facts that relate to sequential steps and processes are also stored in arbitrary areas. The self-organization of data can migrate knowledge from one area to another. Animate objects in memory have no fixed positions (like human beings). Sometimes, there are multiple copies of a human being in the robot's brain. In my patent applications I show how sequential steps self-organize, such as the ABC block problem. Also, I show how a human being is stored in memory. These non-fixed objects are stored in different areas in memory and there are multiple copies. Self-organization will minimize the storage of repeated copies.

Storing knowledge by learning how to store knowledge

The robot learns knowledge from school lessons and these knowledge are already stored in an organize manner. If you look at a form, for example, they list the slots of a person in an organized manner. The name is always the first thing you see, followed by other secondary slots, such as age, phone number, occupation, gender, race and so forth. The robot takes all the forms that it encounters during its lifetime and creates a template form in memory that will store slots and their respective values.

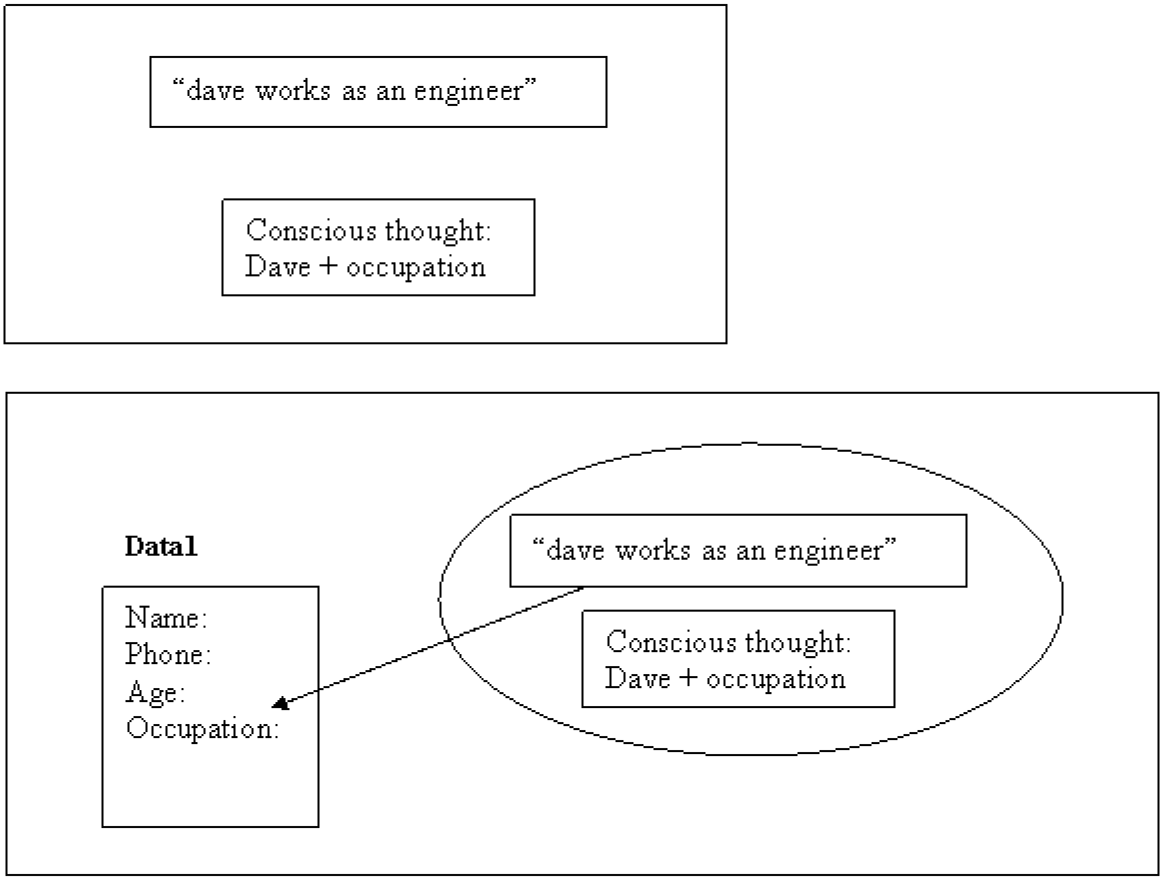
Each person that the robot encounters already has a template form stored somewhere in memory. When information about the person is given, such as their name, the robot takes the name and puts it in the slot: "name". In FIG. 29, data1 is known as the template form. This template form has certain slots and the robot will fill in these slots based on knowledge that it learns from the environment about the object Dave. If the robot was introduced to dave by someone and he said: "this is dave palski", the robot will take dave palski and store it in the template form under the slot: name. Facts learned such as "dave works as an engineer" will be tagged to the template form and it should be located near occupation.

FIG. 29



The process by which the fact about dave is stored near the occupation area is based on FIG. 30. The diagram shows that the robot activated occupation as the identification of the fact. The robot's conscious id that the fact is stating dave's occupation. The fact will be stored near occupation because of association. Both the fact belongs to dave and the fact is about occupation tags it to the occupation slot in the template form.

FIG. 30



Other lessons in school can also be learned to organize data in memory. The transitive statement: A = B, B = C therefore A = C, can be used to summarize events. The robot learns discrete math functions and uses them to represent events in life. The robot actually analyzes a situation first, using math functions to represent the situation and store the logical data in memory.

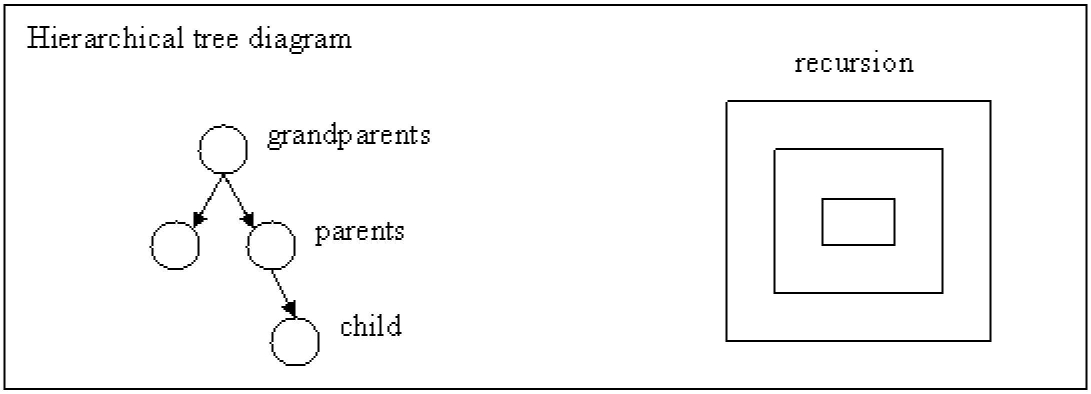
The if-then statement is very powerful in terms of recognizing something and taking action. This is very powerful in terms of doing tasks like driving a car or playing videogames. If the street light is green move forward, if the street light is red stop the car, and so forth. Sequences of if-then statements can be used or simultaneous if-then statements can be used.

These type of learned lessons (such as discrete math) help to organize data from the environment that can be easily remembered. These lessons also help to limit the different types of data organization that are allowed in memory. Easier to understand data organizations or limited data organizations will guide the search functions to find information quicker.

Diagrams of different data structures can help in organizing data. A hierarchical tree can explain the relationship between parent and child. The first node represents the parent and the child node represents children. Information regarding parent-child relationships can use a hierarchical tree to store knowledge. All the knowledge of the parent goes in the first node and all the knowledge of the child goes in the child nodes. The knowledge in the tree will have relational links to objects in the parent node or child node. Processes of relationships can also be explained.

One of the reasons that teachers use a hierarchical tree diagram to explain relationships between a family tree is because the facts are not static data, but are linear data. Steps of how family members are related need to be explained in sequence order. A diagram is drawn and a teacher has to explain what each family member is and how they relate to other family members. Facts will be stored near a family member. For example, the parent always has a mother and father (there are some exceptions). The mother is always female and the father is always male. Next, the teacher has to explain that the parents created the child and the parents have a responsibility to take care of the child. What are the functions of a parent, what does the parents have to do for the child at certain ages, and so forth are stored near the hierarchical tree (FIG. 31).

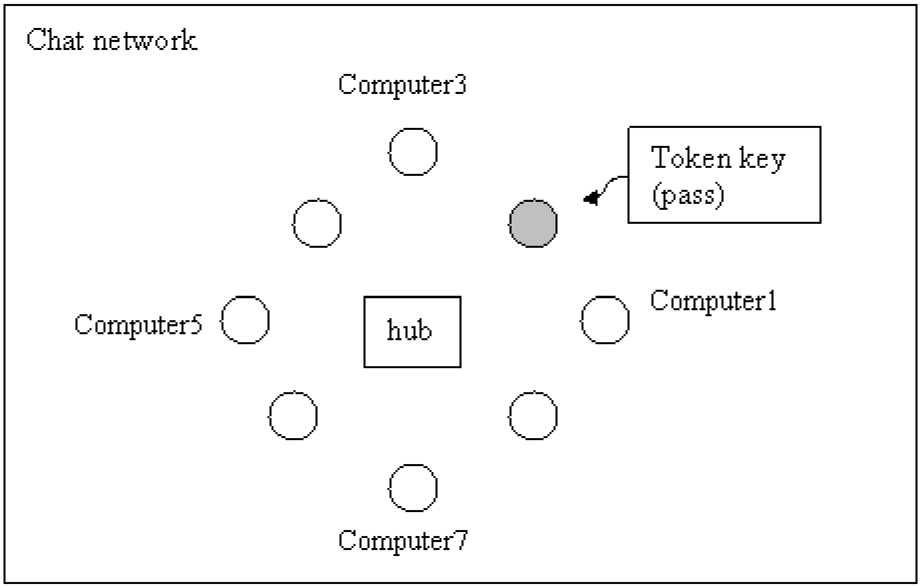
FIG. 31



If the robot wants to know information about family members, it will access the hierarchical tree diagram and search for facts based on areas in the diagram. For example, if the robot wanted to search for parent information, then it can search in the first node because that is where the majority of knowledge related to parents are located.

If the robot wanted to learn how the internet works, diagrams on interconnected computers and a hub can be used (FIG. 32). Explaining of the diagram will be the knowledge to describe how the internet works (this subject matter will be revisited in the next chapter).

FIG. 32



Complex things like knowing how the internet works, knowing how data base system works, or knowing how object-oriented programming works, will depend on visual diagrams in memory. These diagrams are accompanied by many explanations, in terms of a fabricated movie, of how complex things work. In some respects, using diagrams to represent complex things will organize data more efficiently. For example, a family tree diagram is very similar to an object oriented programming diagram. Although, the two diagrams share a hierarchical tree as their commonality, their knowledge is slightly different. The robot can also use family tree knowledge to try to understand how object oriented programming works and vice versa.

At the end of the day, the family tree diagram and the object oriented programming diagram share a common trait and because of the common trait they are brought closer together in memory. This method help organize data in memory.

Configuration of data in memory

An object composite is just a network of data related to one object. The above example about a template form, depicts one example of how data is configured in an organized manner related to one person. When people give important information about dave (FIG. 29) like his full name, address, phone number and occupation, these data are identified and stored in slots in a template form.

This is how the robot “learns” to configure data from the environment and to store them in memory in a meaningful and compatible format. Other configurations of data might include diagrams learned to store facts about a person. A hierarchical tree can store data on family relationships and the behavior of family members.

In discrete math, there are diagrams that people draw to show relationships and to store facts about objects. People learned from teachers how to draw these diagrams on paper. These diagrams can be a form configuration of data from the environment. It will create an organize object composite, whereby information about that object is stored and organized in memory according to learned diagrams.

For example, in some predicate calculus problems, the robot has to translate sentences into a visual diagram. A circle might represent an object and an arrow might represent a function. The writing down of the action on the arrow indicates what function. The robot can learn this behavior and generate its own type of data configuration, not just based on sentences, but visual images or 5 sense data as well.

Another theory is that maybe a book can represent information about a person. On the first page might be a form of the person. In the later pages there might be pictures and facts about that person. This template book is like a physical book, whereby the robot has to flip through pages to get information. The physical book has a front cover and a back cover so that it defines the boundary of this object composite. Information in this template book might have relational links to other template books.

To sum up this chapter, the brain learns diagrams from textbooks and teachers and uses these visual diagrams to configure and store data from our environment. No computer scientists are needed to write any of the functions or classes. Things like database systems, image processors, an operating system to manage multiple tasks, and logic gathering are all done through lessons taught in school.

The overall AI program illustrates an image processor with 6 dissection functions (FIG. 1). To tell you the truth, the image processor is not necessary. The robot learns how to delineate image layers by examples in real life. If you observe teachers teaching students, the teacher will point to an object and this pointing gesture is to guide the students' eyes to focus on the object. The human eye focuses on objects and the object itself is clear, while all other surrounding objects are blurry. This eye focus function automatically cut out a perfect image layer from the environment.

The eye focus only works for 3-d objects in real life, what about 2-d still pictures, how does the robot delineate these objects? The robot encounters data in the real world, so it know what objects look like in terms of 3-dimension. The robot learned what a dog looks like from seeing a dog. When the robot see a dog in a 2-d still picture, the 3-d data from a dog stored in memory will activate and it will give the 2-d still pictures a 3-d shape. This 3-d shape delineates the image layer from the 2-d still picture.

8 Years ago I was watching Sesame Street to observe how children learn information. I notice that one method that the show delineate objects and to focus the eyes of the child is to use a digital outliner. The scene is a chaotic jungle and there are lots of trees and animals in the scene. The teacher would ask: “where is the monkey?”, and a few seconds later, the digital outliner will outline the boundaries of the monkey.

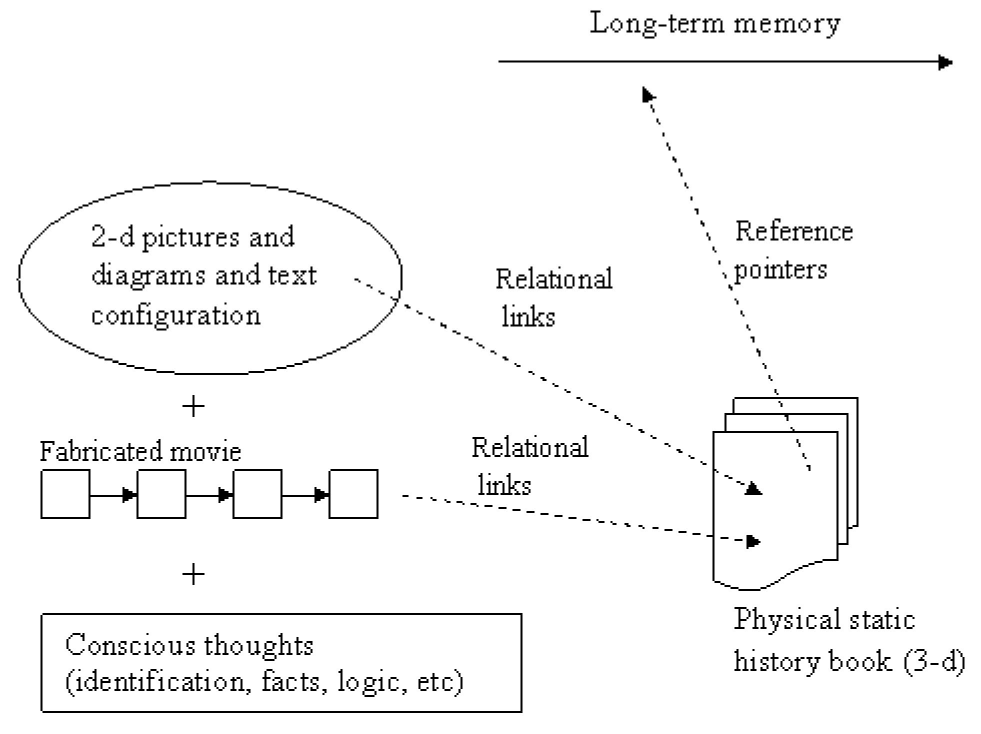
This behavior is learned by the robot. He is able to form a digital outliner to delineate the boundaries of any object in the environment. However, the robot id image layers in an approximate manner and not an exact manner. Identifying events and situations might be difficult, but the robot will assume the approximate area or objects involved in the situation.

If this behavior is learned by the robot, he will be able to see in his mind an invisible digital outline of objects in the environment. He is able to id what elemental parts belong to what objects and determine their boundaries.

Learning data in history books

FIG. 33 is a diagram illustrating how the robot learns knowledge in history books. Assuming that the robot reads the book linearly, the pages are stored in memory like a physical book – page by page. The page storage might be approximate and not entirely sequential. The history book will have reference pointers to the robot's long-term memory, when each page has been read.

FIG. 33



When the robot reads text from the pages, a fabricated movie will activate and this fabricated movie serves the same purpose as witnessing movie sequences from real life. However, the fabricated movie is fuzzy and is constructed by cutting, pasting and merging 5 sense data together based on the text read. The conscious thoughts will activate to identify objects in the pages and to give facts or provide logical analysis. In addition to the fabricated movie and the conscious thoughts are the still pictures, text configurations and diagrams from the history book. Some sentences are describing the pictures on the page and the robot's brain has to establish relational links between the text and picture.

The conscious thoughts are vital to the remembering and the storing of data. The robot will use logic to id paragraphs, to delineate chapter boundaries, to summarize paragraphs, to interpret alternative meaning to complex sentences, identify important facts and so forth. These thoughts really help to organize the data and prepare it to be stored in memory.

All the data from the physical 3-d history book and the fabricated movie are stored in fragmented places in memory. There are two types of areas that the data will be stored in: the optimal pathway and masternodes of objects in the current pathway. The robot's brain jumps from one area in memory to the next in order to store the current pathway. Thus, the physical 3-d history book is stored in fragmented areas in memory. It will attempt to store the current pathway in a linear fashion. The only thing that binds the history book together is the long-term memory. Next, other data like fabricated movie, conscious thoughts and still pictures are also stored in their masternodes (the area each data is strongest in). For example, if there exist a dog picture in the history book, the dog picture will be stored in the optimal pathway and it will be retrained in the strongest area where a similar looking dog image is stored. Retraining data is based on storing a copy of the original with less data. Or, if the copy already pre-exist in the strongest area (the masternode) the pre-existing data will be strengthened.

Over time, the robot's brain forget information. Let's say that 10 years after reading the history book the robot forget ever reading the history book. That doesn't mean the information from the history book is completely gone from memory. You still have data stored in the masternodes. This means that the majority of content from the history book still exist even though most of the physical 3-d book stored in memory is forgotten.

The physical 3-d history book is very important in some cases. If the robot was asked this question: "what chapter describes world war II", the robot has to activate a memory of the physical history book, locate world war II data, and logically output the probable chapter. In order to do this, the robot's brain has to use an intelligent pathway that analyzes the history book. This intelligent pathway will locate the world war II chapter, flip through chapters before it and flip through chapters after it and determine where the chapter is located. In the robot's mind the physical book in terms of its 3-d shape and size are stored in memory. If the robot determines that the World War II chapter is located in the middle of the book and the book has 20 chapters, then it can assume that the world war II chapter is chapter 10. If the robot determines that the world war II chapter is located in the beginning of the book, he will say chapter 3 or 4.

The conscious thoughts activate what chapter while you're reading the chapter. This will make it easier to organize the knowledge.

Other questions asked might be: write down a summary of all chapters in the book or explain what caused world war II? The first question uses both the physical 3-d history book and content in the masternodes. On the other hand, the second question uses only the masternode information. The robot doesn't need to know what chapter World War II is located, he just needs to know information about world war II.

Data in memory have multiple copies scattered throughout the robot's brain. The masternode is the strongest copy of that data and it has reference pointers to other weaker copies in memory. The primary storage area for a data is stored in the optimal pathway, which may or may not be the masternode. This type of learning is required because all data in memory are global and interconnected. Regardless of where the current pathway is stored in memory, each object in the current pathway must locate its respective masternode to be retrained.

Chapter 8

4. Reading science books

In this chapter, I will increase the complexity by discussing how the robot learns knowledge from science books. In order to learn science knowledge, all three previous learning methods have to be used. In science books, there are facts/knowledge about objects; and there are diagrams that explain how objects work. In the previous learning methods, we discuss how knowledge is learned and stored in memory. These knowledge are based on sentences and paragraphs. What makes science books particularly difficult to learn is that it contains diagrams and flow charts to explain a process.

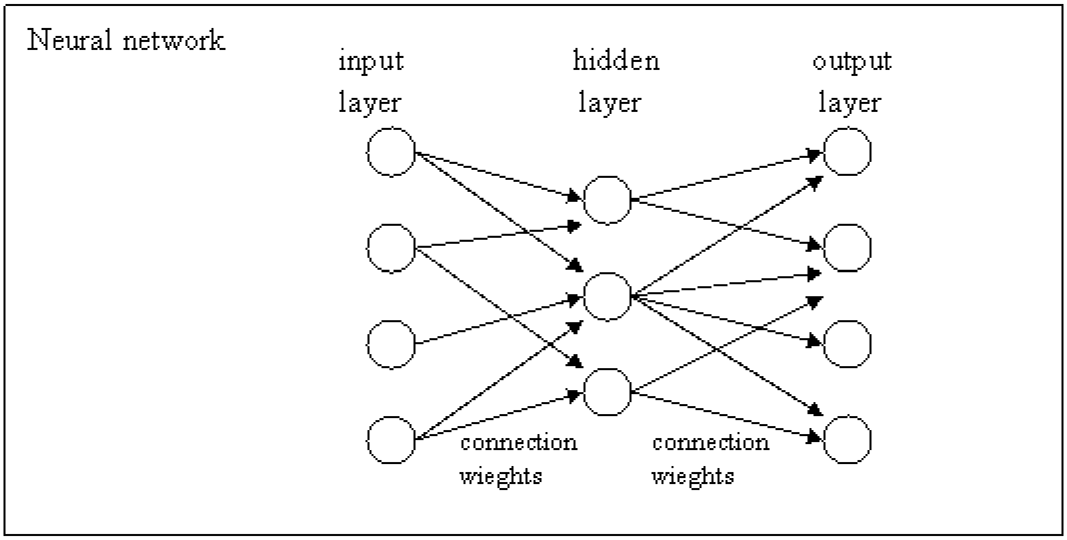
In addition to reading sentences and paragraphs, the robot has to also learn to understand diagrams and how they work. The sentences in science books explain how diagrams work. They give facts about various areas of a diagram and also to give the reader a fabricated animation of the sequential steps of a process in the diagram.

The pathways that the robot stores in memory contain both static data and linear data. Diagrams are considered static data because they are still pictures in the science book. Some diagrams are flow charts, so they are actually an animation that represents a process.

A neural network example

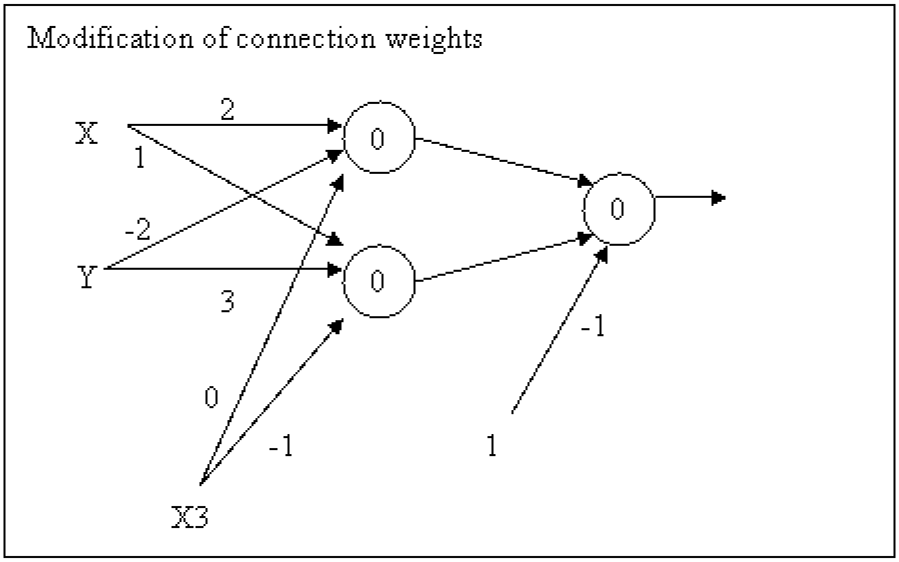
If you read an AI book about neural networks, they will give you a diagram similar to FIG. 34. The author will explain using sentences how the neural network works. First, the book will explain the parts of a neural network. There are three layers in a neural network: the input layer, the hidden layer, and the output layer. The explanation of the parts related to a neural network gives facts about what each part does. By understanding the individual parts, the author can then explain the linear functions of the neural network. The steps to a neural network includes: inserting data into the input layer, then the output layer will delineate the desired output, and finally the neural network modifies all of its connection weights in all three layers. The learning of the neural network happens during the modifications of the connection weights.

FIG. 34



The above information only outlines a general idea of the neural network. The science book will explain further the details of how the neural network works. In order for the robot to organize data, it has to use one type of diagram for one type of explanation. For example, the diagram in FIG. 34 is used for a general explanation. In FIG. 35, the diagram is used to explain how the connection weights are modified. The robot will see the steps that the neural network will go through to change the weights. Equations are given in the science book and the robot has to use these equations and apply them to the connection weights of the neural network.

FIG. 35



In some cases, the teacher will ask the robot to write down the flow chart of all the connection weights and how they are modified. The book will tell you how to apply the equations to the connection weights. The robot has to follow the steps and write down the numbers on each connection weight step-by-step. If the robot did the task correctly, the connection weights will have the correct numbers.

For every detail diagram of the neural network, the information will be stored in their respective area. The connection weight diagram in FIG. 35 will be stored in the general neural network diagram in FIG. 34, in the connection weight area. These diagrams will also self-organize and their optimal area of storage will be outlined.

If there was a diagram to explain each layer of the neural network, then each diagram will be stored in the general diagram in their respective places. The detailed diagram for the input layer will be stored in the input layer, the detailed diagram for the hidden layer will be stored in the hidden layer, and the detailed diagram for the output layer will be stored in the output layer.

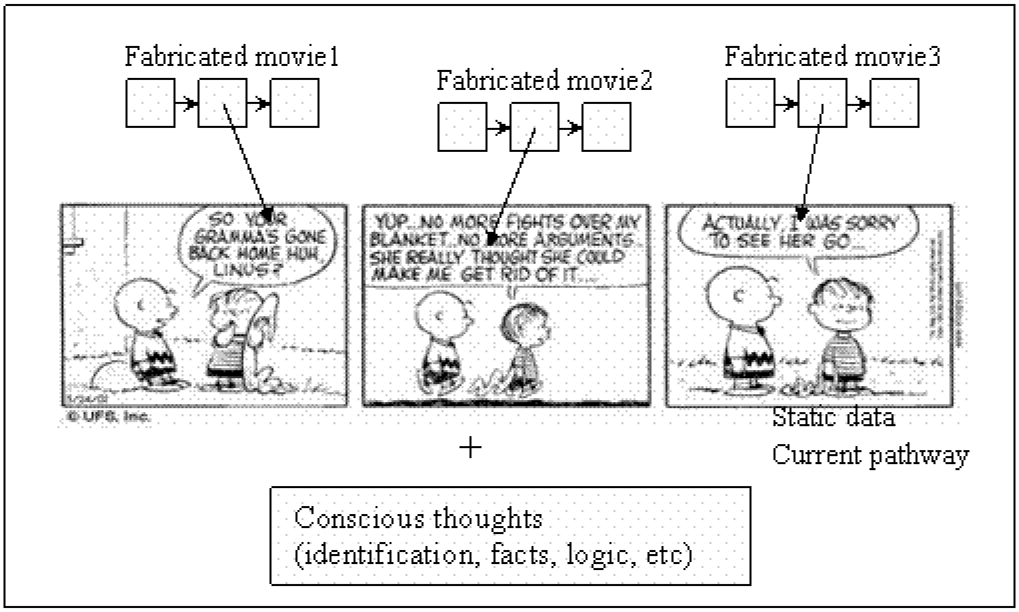
Another type of diagram is used to show the different variations of neural networks. Instead of 3 layers, some neural networks have 4 or 5 layers. The author can explain what kind of variables can be used in each layer. The author can also explain what kind of techniques to use to determine the variables in the desired output.

Sentences store different timelines of events

In history books, the authors give information about certain events in history. The author will tell you, through the books, when these events occurred. The constitution convention occurred in 1787. The author will then give you information about what happened during that time period. The fabricated movie created by the robot's conscious, while reading the section on the constitution convention, should record the event occurred in 1787.

History books aren't a very good example to use. Instead, I will use comic books. Referring to FIG. 36, the fabricated movie can actually be other timelines besides what is currently happening. It could be a scene that happened in the past or the future. It can also be a scene that is currently happening now, but isn't located in the current pathway. The current scene (the fabricated movie of a current non-viewable scene) can be a scene that is happening 1 mile or 1,000 miles from the current pathway.

FIG. 36



The primary story is based on the current pathway (the sequential panel recognitions of the comic book). The fabricated movies of other events happening at different times are secondary events. The characters in the comic book are explaining past events or future events. Maybe Charlie brown is planning to buy a lawnmower tomorrow or Charlie brown bought a new bike at the department store yesterday. These types of information doesn't exist now, but it either happened in the past or will happen in the future. Sometimes, Charlie brown will describe a 2 minute event that happened yesterday. The reader has to visualize what that 2 minute event is like (a fabricated movie).

Fabricated movies should generate a 5 sense data representation of any grammar sentence. A sentence might include an object belonging to Charlie brown or Charlie brown taking an action. Thus, the fabricated movie uses 5 sense data to generate a meaningful representation of the sentences read. These fabricated movies should represent all objects, events, and actions in the current pathway and establish relational links between them in terms of physical properties, hidden thoughts, object interactions, time, and so forth.

The fabricated movie can also reference events that happened in the current pathway in the past or future. Reference pointers will be established between events in the fabricated movie and past/future events of the current pathway.

5. learning science lessons from teachers

A more powerful way to learn information is to have a teacher demonstrate examples and give facts about different subject matters. It's one thing to read a science book and learn the information, but to have a teacher teach the lesson is a totally different experience. We learned how to study when we were in grade school. We learned the procedural steps that are needed to pick up a science book and to learn the information. Reading a comic book is totally different from reading a science book. Reading a Chinese book is different from reading an English book. There are rules that have to be followed based on a given book. These rules decide how to read a book, how to look at diagrams, and how are the chapters in the book arranged. (I will be discussing how to form intelligent pathways in memory to read a given book in the next chapter).

The reason why a teacher is needed to teach the robot knowledge is because he/she serves as a guide that helps the robot focuses on certain objects and illustrate, using a chalkboard (or monitor), how things work in a sequential manner. If the teacher is trying to teach the robot an addition problem, the teacher has to write things on the board, erase things on the board, focus the robot's eyes on certain things on the board and so forth. While the teacher is illustrating a lesson, he can also give facts and information about that lesson.

The robot's brain has to store all these lessons in memory in an organized way. If you combine all 5 learning methods:

1. reading novels. 2. reading comic books. 3. reading history books. 4. reading science books. 5. learning science lessons from teachers,

the robot should be able to organize the knowledge it experiences from the environment in an intelligent way. Not all information is stored from the environment. The robot uses intelligent pathways to focus on limited data from the environment and it stores these data in memory in an organized manner. The teacher that is teaching the lesson is the guide that will help the robot to focus on what matters from the environment and what information should be stored in memory. For example, the robot looks at the teacher and sees the way he/she dresses and he sees the furniture in the classroom, including the chalkboard. However, despite all this information, the lesson on the chalkboard is the primary data that should be stored in memory. The robot will give top priority to the lesson on the chalkboard and low priority to anything else.

This type of teaching is much easier to learn compared to reading a book because books has fixed diagrams and text. If the robot doesn't understand something he can always ask the teacher. If you read a science book and you got questions, the book can't give you an answer, you have to search for the answer yourself.

Learning information from teachers should be applied to mathematics. If you try to learn math from a book, it would be very difficult. People learn information mainly through visual images. Diagrams in math books are very limited because they take up too much space. For example, flow diagrams take up a lot of space in books. Usually, the author will provide a detailed diagram and expects the reader to use common sense to fill in any missing data. This confuses the reader and will make it more difficult to understand the lesson.

Teachers, on the other hand, understand the lesson and he/she will teach the class so that the students can understand the lesson.

Believe it or not the conscious mind of the robot is formed by lessons from teachers. Even when a person is reading a science book, the conscious is like a teacher that tells you things about the content. In other words, you are able to read and understand the science book because of lessons from teachers about guidance.

Teachers have a finger and can point to certain areas on a chalkboard. The pointer is a guide to focus the robot to a certain area. The teacher can also write things down, erase things and draw action lines on the chalkboard. Most of the lessons on the chalkboard can be categorized as static data or linear data. A math equation demonstrated by a teacher will show existence of numbers (static data) and linear steps (linear data) in order to solve. Through self-organization of many similar examples, the robot will know which are static data and which are linear data.

Teachers can also use finger pointing and diagram drawing to show relationships between objects. These relationship patterns can also help in configuring data for storage purposes. For example, the teacher can draw 5 diagrams on the chalk board and he/she will point to one diagram and say that object1 is a parent to objects2-5. This lesson will create a diagram in the robot's mind and store object1 as a parent node and objects2-5 as child nodes. The teacher might draw a diagram of a line and he puts object2 on the left side and object5 on the right side and say that there is a range between object2-object5. Object2 is low and object5 is high.

These types of lessons coming from teachers help to organize the material being learned. In the future the robot can use these demonstration lessons and apply them to understanding materials in science books. For example, if you read the last paragraph again you will create the diagrams in your mind. I did not physically write on the chalk board the two diagrams. I simply written a paragraph describing a teacher drawing the diagram. Because you are intelligent at a human level you are able to fabricate a movie of a teacher drawing the two diagrams. The first diagram is a hierarchical tree of a parent node represented by object1 and 4 child node represented by objects2-5. The second diagram is a line with object2 on the left and object5 on the right. The range from left to right is low to high.

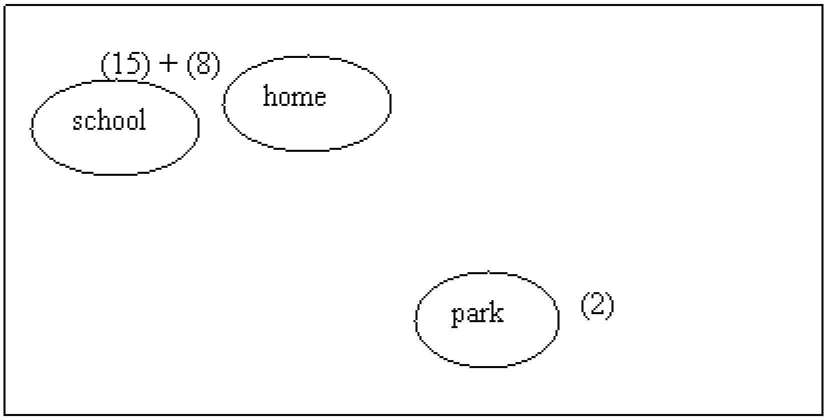
Self-organizing knowledge from multiple books in the brain

Books are read in the classroom or at home or at a park. Because robot's store data in a 3-d grid, the location of where the robot read the book is where the data will be stored.

The next factor is that the visual book also helps organize data. The book is a visual media and the robot is storing information from the book in a linear manner (page by page) of what it had read. Facts/knowledge, diagrams, pictures, flow charts are all arranged in a 3-d manner of where it was read in the book. All data will also have relational links to each other.

If the robot read a math book in school, at home and at a park at different times, then the information is stored in their respective locations (FIG. 37). During the self-organization phase, the data in the math book read at school and at home will self-organize (they combine the data and share them). The data from the park is too far away and isn't subject to self-organization. Self-organization prevents repeated data from being stored in memory. After self-organization, there are two areas where the data from the math book are stored: the location between school/home and the park. The masternode will represent the copy that has the strongest powerpoints, which in this case is 23.

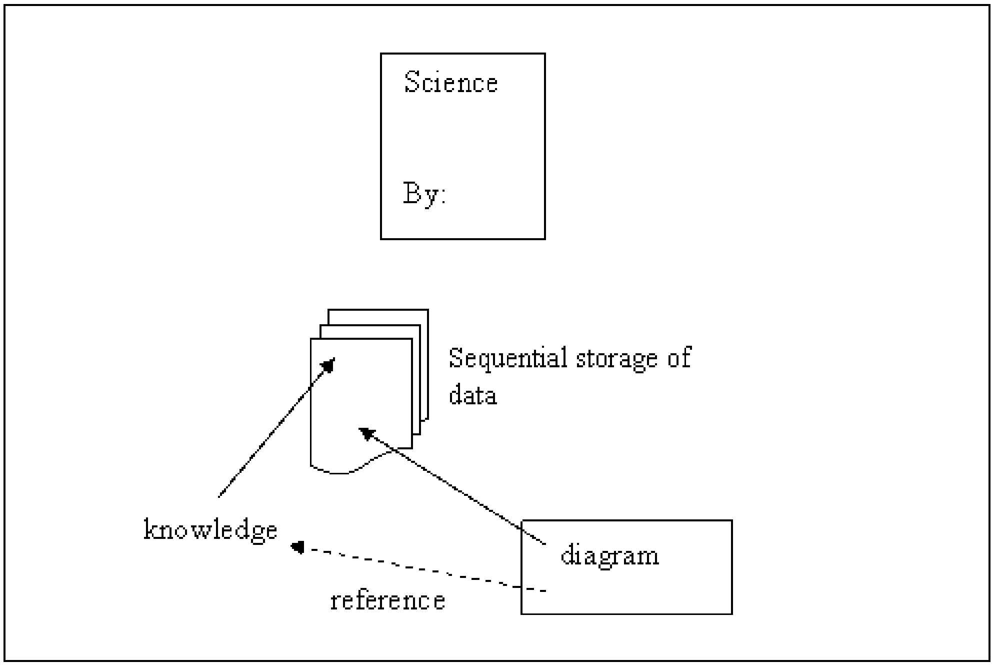
FIG. 37



Let's say that in the future, the robot reads the same math book (the same chapter) at a different location. The robot is in a library and is reading the book and the library is located very far from the school or park. The robot will initially store the math problem in the library location, however, after reading the data for a short while it will start to store the data in the school area (the masternode). The reason for this is because the data in memory regarding the book is so strong in the school area that when the robot reads the book and is unaware of its environment, the robot's brain found the school area to be an optimal area to store the data from the math book. This method demonstrates that storage of data doesn't totally depend on the current visual environment.

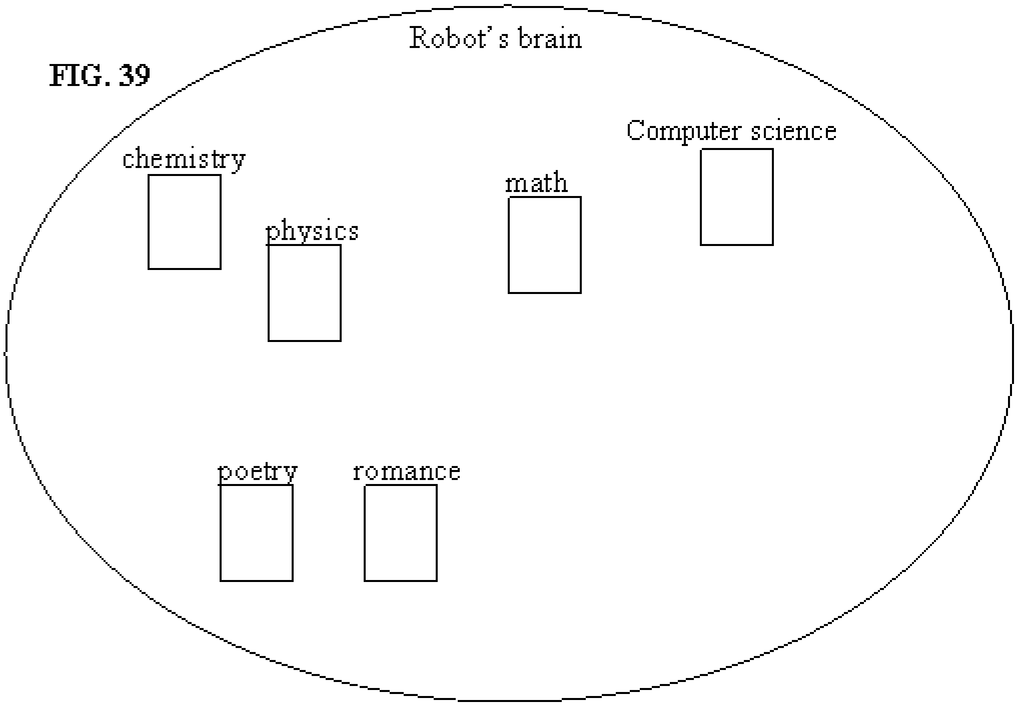
Referring to FIG. 38, as stated before, the contents learned in books are stored as linear data, organized page by page. Information are stored in an estimation. In real world examples, the brain forgets information and it confuses sequential pages. The diagram shows facts/knowledge being stored, along with diagrams and pictures. Most of the time, the knowledge is referring to the diagram and reference pointers are included.

FIG. 38



The color of the book, the title of the book, the author, the subject matter and so forth are information used to organize the content read in the book. Despite all the content inside a book, the robot will self-organize many books in his lifetime. The self-organization will depend on the subject matter. Physics books will be closer to chemistry books, love poetry will be closer to romantic books, math books will be closer to computer science books and so forth. These books will self-organize all their content. As the robot learns more information, they will be structured in a hierarchical manner (FIG. 39).

FIG. 39



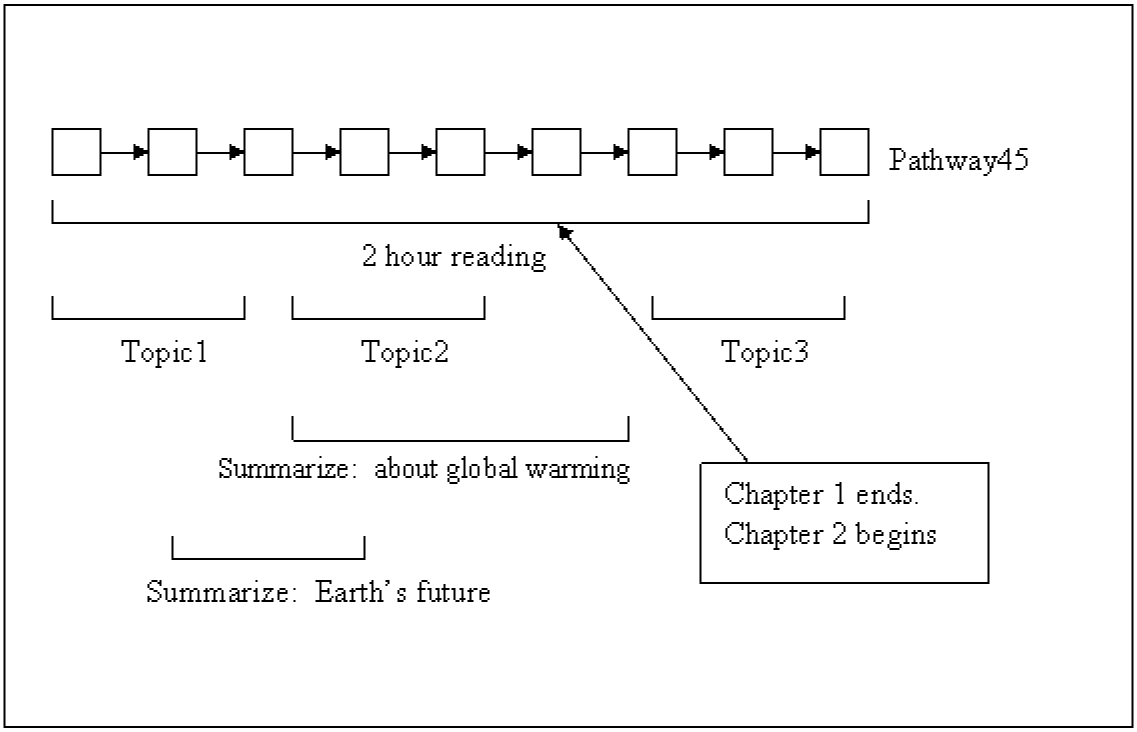
Organizing data through the robot's conscious thoughts

The above methods show that the robot's brain can store books in the place that they were read, and also, by the visual book itself. The content of the book will be stored in an organized way through the robot's conscious thoughts. Referring to FIG. 40, when the robot recognizes something the conscious activate sentences or words to id objects. The robot will id the topics contained in paragraphs or id important facts from several pages. The robot's conscious might also summarize several pages of reading. All this information that comes from the robot's conscious is used to organize data.

All facts from topic1 will be stored in topic1, all facts from topic2 will be stored in topic2 and all facts from topic3 will be stored in topic3. Hierarchical data storage is also possible. Topic1, topic2, topic3 and summarizing facts are all stored in the 2 hour reading.

This hierarchical data storage is an estimation and data in memory forget information. The beginning and the ending of topic2 is just an approximate estimation. Pathways are broken up into a plurality of fragmented pathways when they are forgotten. Some data might even migrate to distant areas.

FIG. 40



Intelligent lessons learned from teachers

The lessons learned in school form intelligent pathways in the robot's memory. The robot can do math, science, draw, speak and so forth. Some intelligent pathways require the robot to write down things such as an addition problem. Other times, the robot can do simple tasks in its mind, like planning routes to drive a car from the current location to a destination location. When planning routes, the robot will id its current location, activate a fabricated map in its mind, draw lines to connect where it is and where it wants to go, and remember the linear streets it has to travel.

The reason that the robot is able to plan routes in its mind is because of doing the same task in the real world. In school, teachers give the robot worksheets to do. A simple worksheet is a mouse in a maze and the robot has to draw lines so that it can get the mouse from the start location to the end location. The robot has to analyze all the routes and plan the routes by drawing lines. If the robot is stuck with a path, it can erase the line and take other paths.

By doing worksheets of plotting routes, the robot is able to use this lesson on any type of similar problem. It can take a toy car and push it around to a destination location or take a pencil and draw routes to get a car to a destination location.

In the case of planning routes in the robot's mind, the robot has to activate a map of the city. Next, it will use the lesson from worksheets related to planning routes, to determine what streets to drive to. Finally, the robot remembers the linear streets to go to and apply them to the real world. Planning routes on the road is a simple example, and the robot can do this all in its head. More complex route planning will require the robot to write down on paper the routes needed to get to a destination location. Usually, human beings plan at the moment. If the robot had to drive to a far distance area, he has to plan 3-4 times to get there.

I can't emphasize how important it is for the robot to have the ability to do all worksheets. The worksheets establish vital intelligent pathways in memory to do things. The robot needs to learn how to cut and paste images together in order to fabricate make believe movies in its head. When reading a book, a fabricated movie is created. The fabricated movie is based on intelligent pathways that take lessons from cutting and pasting images in the real world. If you have a worksheet that wants you to cut objects out and paste them together in a meaningful way, then the robot has to observe the objects and see which objects belong to each other. A dress should be pasted onto a girl and high heals should be pasted onto a women. A tuxedo should be pasted onto a man and a toy should be pasted onto a baby. The robot also has to know where they should paste these objects. The head of a girl should be pasted onto the top-middle of a dress.

Lessons about coloring books and cutting out boundaries of an object can be used for the robot's internal image processor. This image processor can delineate an image from a picture or movie sequence without any programmed computer codes. The robot learned how to cut out images from lessons in real life.

Looking at forms in the real world will generate template forms to store background knowledge on objects (like human beings). The observation of the form and all of its slots teach the robot how background data on a person should be stored and structured.

The search functions in the pathways of the robot are done by patterns found between similar examples. No programmers are needed to write the computer codes to the search functions. The robot does all the hard work by comparing and searching for these patterns between similar pathway examples.

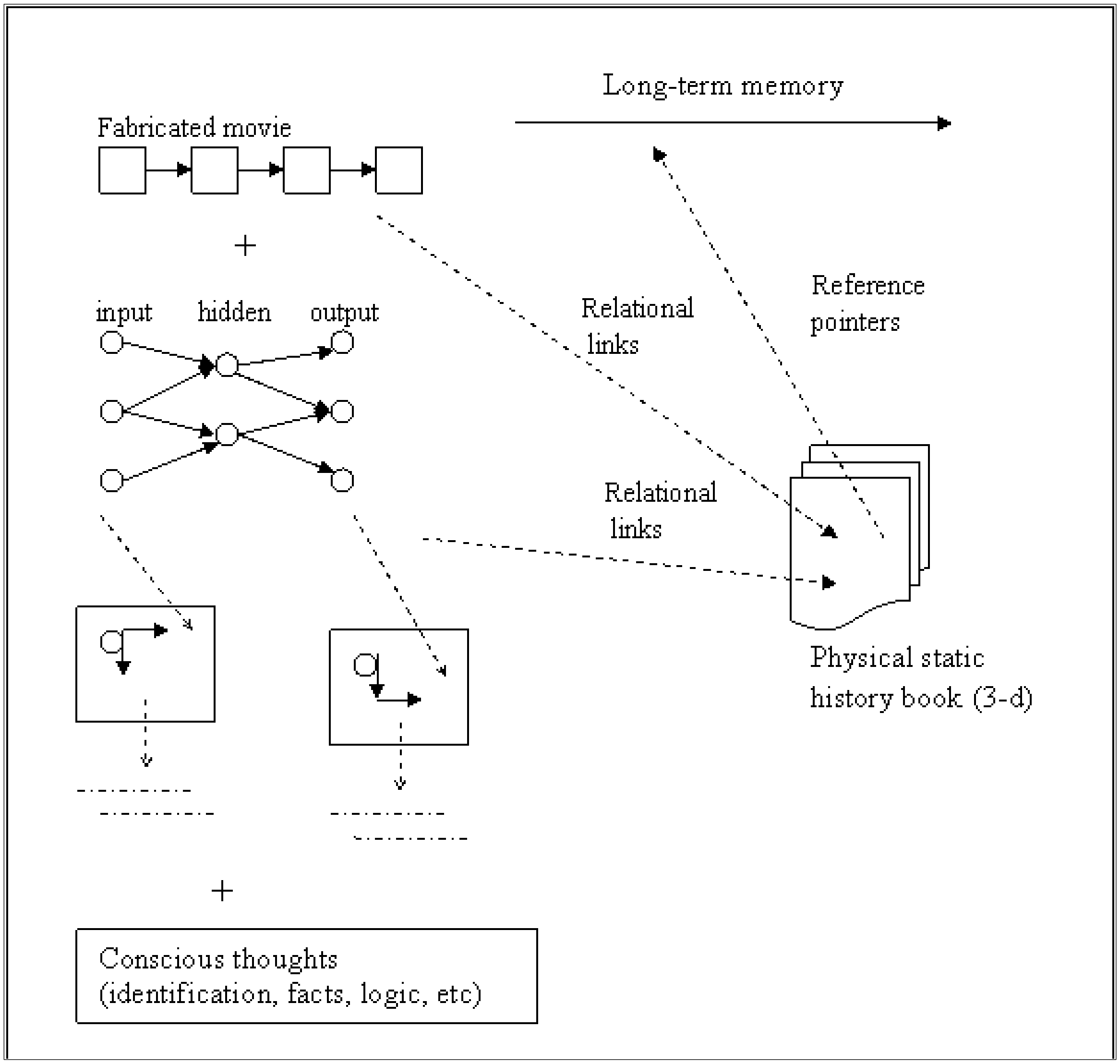
More on reading science books

The robot learns a science book very similar how it learns a history book. There are slight differences. For one thing, in the history books, the reader is more interesting in creating a fabricated movie of scenes from the text. It's almost like watching a movie of the history book. On the other hand, the science book requires looking at diagrams and understanding flow diagrams.

The understanding of a neural network require, first, understanding the element parts of the neural network: input layer, hidden layer and output layer. Next, the overall idea of the neural network is learned. The programmer has to define the output layer based on trained input data. Next, the programmer has to train the neural network with many training examples. During the training phase, the connection weights adjust between the input layer and output layer so that it can "learn". When the neural network is trained properly, it can be used by people.

In FIG. 41, the diagram of the neural network is the bulk of the information from the science book. All the text are explaining how the diagram works. The various diagrams from the book will organize itself into hierarchical levels. For example, an overall diagram of the neural network will be at the top and a diagram of the input layer will be located in the input layer area of the overall diagram.

FIG. 41



In science books there are diagrams as well as flow diagrams. The flow diagrams show you step by step how the neural network works. The robot's brain can use previously taught lessons from teachers (an intelligent pathway) to convert the flow diagram into an animated flow diagram. The intelligent pathways can fabricate a movie of what the flow diagram should look like. For example, a fabricated movie of how a neural network work can be a diagram of a neural network, a user inputs data into the input layer and the connection weights change, and finally, the output layer outputs something.

The fabricated movie will require the flow diagram and the sentences that tell the reader how the steps work linearly. The diagrams along with the sentences will generate a more detailed animation of the functions of a neural network.

These fabricated movies are also put into the overall diagram according to its associations. If the fabricated movie creates steps to how the input layer works, the fabricated movie will be stored right next to the input layer in the overall diagram.

Even if the book doesn't have a flow diagram, instead, have a still diagram, the robot can still generate a fabricated movie based on the sentences in the book. The still diagram might be the overall diagram of the neural network. The sentences around the diagram will explain the linear steps to how the neural network operates. This fabricated movie of the function of the neural network will be stored in the same area where the overall diagram is stored in memory.

The important thing to remember is that the information stored in memory about the neural network are based on 5 sense data, mainly diagrams and fabricated movies. The extraction of knowledge is based on the intelligent pathways in memory and not the data stored in memory. For example, let's say that a teacher asked the robot this question: "what color is a human heart", the robot will only extract a picture of a human heart from memory. An intelligent pathway is needed to analyze the human heart and determine that the color is red. It's the intelligent pathways that really do the hard work and not the data stored in memory.

How exactly does this analyzing intelligent pathway form in memory? Teachers will teach the robot in real life using examples. The teacher will take out a picture of a heart and ask the robot: "what color is the heart". The robot will focus on the color and say: "the color of the heart is red". In order to understand how to answer this question the teacher uses supervised-learning, whereby he/she has to ask the question and give the answer. The robot's responsibility is to find the search pattern between the question and the answer.

This intelligent pathway was trained to analyze a picture that is currently focused on by the robot. It can also be used to analyze a picture in the robot's mind. The teacher can trick the robot and ask this question: "I want you to remember the heart picture, what was the color of the heart". The robot will activate the heart picture presented earlier and it will use the intelligent pathway of analyzing the picture to output: "the color of the heart is red".

Intelligent lessons learned from teachers

The lessons learned in school form intelligent pathways in the robot's memory. The robot can do math, science, draw, speak and so forth. Some intelligent pathways require the robot to write down things such as an addition problem. Other times, the robot can do simple tasks in its mind, like planning routes to drive a car from the current location to a destination location. When planning routes, the robot will id its current location, activate a fabricated map in its mind, draw lines to connect where it is and where it wants to go, and remember the linear streets it has to travel.

The reason that the robot is able to plan routes in its mind is because of doing the same task in the real world. In school, teachers give the robot worksheets to do. A simple worksheet is a mouse in a maze and the robot has to draw lines so that it can get the mouse from the start location to the end location. The robot has to analyze all the routes and plan the routes by drawing lines. If the robot is stuck with a path, it can erase the line and take other paths.

By doing worksheets of plotting routes, the robot is able to use this lesson on any type of similar problem. It can take a toy car and push it around to a destination location or take a pencil and draw routes to get a car to a destination location.

In the case of planning routes in the robot's mind, the robot has to activate a map of the city. Next, it will use the lesson from worksheets related to planning routes, to determine what streets to drive to. Finally, the robot remembers the linear streets to go to and apply them to the real world. Planning routes on the road is a simple example, and the robot can do this all in its head. More complex route planning will require the robot to write down on paper the routes needed to get to a destination location. Usually, human beings plan at the moment. If the robot had to drive to a far distance area, he has to plan 3-4 times to get there.

I can't emphasize how important it is for the robot to have the ability to do all worksheets. The worksheets establish vital intelligent pathways in memory to do things. The robot needs to learn how to cut and paste images together in order to fabricate make believe movies in its head. When reading a book, a fabricated movie is created. The fabricated movie is based on intelligent pathways that take lessons from cutting and pasting images in the real world. If you have a worksheet that wants you to cut objects out and paste them together in a meaningful way, then the robot has to observe the objects and see which objects belong to each other. A dress should be pasted onto a girl and high heals should be pasted onto a women. A tuxedo should be pasted onto a man and a toy should be pasted onto a baby. The robot also has to know where they should paste these objects. The head of a girl should be pasted onto the top-middle of a dress.

Lessons about coloring books and cutting out boundaries of an object can be used for the robot's internal image processor. This image processor can delineate an image from a picture or movie sequence without any programmed computer codes. The robot learned how to cut out images from lessons in real life.

Looking at forms in the real world will generate template forms to store background knowledge on objects (like human beings). The observation of the form and all of its slots teach the robot how background data on a person should be stored and structured.

The search functions in the pathways of the robot are done by patterns found between similar examples. No programmers are needed to write the computer codes to the search functions. The robot does all the hard work by comparing and searching for these patterns between similar pathway examples.

Chapter 9

Decision Making and Planning Tasks

There are many different levels on decision making and each level influences the way the robot makes decisions. Below is an outline of the level of factors that will influence the robot (the AI program) in terms of decision making.

Levels of Decision Making:

1. innate reflexes based on pain--when a person is in great pain reflexes are most likely to trigger. These reflexes are wired into pain so that when the pain reaches a certain point it triggers the reflex. No conscious decision was needed to trigger this action. Some of these innate reflexes are: when a person is in great pain he/she yells out loud, when the knee cap is hit with a hammer the leg moves automatically, etc.

2. Learned decisions based on past knowledge--the conscious guides the robot to make decisions. These decisions are either based on future predictions or logical decision making.

3. Pain and pleasure built into the robot--attractiveness or ugliness, physical pain (degree of pain) and physical pleasure (degree of pleasure) are factors that the robot uses to make decisions. Is the robot going to eat lobster for dinner (the robot loves lobsters) or is the robot going to eat rice? (the robot eats rice only if he has to). These pain/pleasure factors that are built into the robot will make decisions.

4. Daily routine--Learned things that the robot was thought everyday by teachers are factors in decision making. Some of these daily routines are so natural that the robot doesn't need to make a decision to do them. Daily routines such as: waking up in the morning, brushing your teeth, eating 3 meals a day based on the time, going to sleep at night, using the bathroom when you need to go, and going to work or school on weekdays.

These are the levels of decision making. The higher levels overshadow the lower levels in terms of decision making. For example, innate pain overshadows learned decisions because innate pain is a reflex and is triggered by pain while learned decisions uses conscious thoughts to make decision. In other words, innate pain is triggered without conscious thought and overshadows learned decisions.

Another example is learned decisions can overshadow pain and pleasure. This form of pain and pleasure doesn't trigger reflexes, it's just a lower degree of pain/pleasure from innate reflexes--a degree where the robot can manage the pain. If a person has an itch on his butt and this person is walking on the street, the person can make a decision to scratch his butt or not. He can wait until he gets to a private area before scratching his butt. This is one demonstration of learned decision having higher priority than pain/pleasure. Even something like using the bathroom require learned decisions. If you have to go the pain is unbearable. However, you can't take a dump on the street or in a classroom. You have to make a decision to go to the bathroom and take a dump. Even though the pain is so great the learned decisions guided the robot to take the appropriate actions.

Pain and pleasure is another factor that is used for hidden objects. The AI finds these patterns and wire pathways with pain and pleasure. The strongest pathways have their powerpts strengthened because it's wired to pleasure and the weak pathways have their powerpts lowered because it's wired to pain. The learned decision encapsulates pain/pleasure to plan out tasks and make decisions for the robot. The main function of decision making is always to pursue pathways that lead to pleasure and stay away from pathways that lead to pain.

Daily routines such as brushing your teeth, sleeping at 9 pm, and waking up at 7 am are just things that we learn everyday and this type of learning is so normal that we do them without thinking. Learned decision can overshadow these things because we can control when we sleep by conscious thought. Instead of 9 pm we can sleep at 2 am. Instead of eating cereal for breakfast we can eat a hamburger. This daily routine is also another factor that can be encapsulated into learned decisions to plan out tasks and make decisions for the robot. The AI program finds patterns concerning daily routines and use this pattern in a hidden object. This hidden object will then be assigned to words/sentences as meaning of words/sentences.

In this chapter we will concentrate on learned decisions. Learned decisions are lessons that are taught to the robot by teachers. The teachers teach the robot how to make decisions. Many decision making lessons will be taught to the robot. When the robot is in the supermarket to buy food he has to decide what foods to buy. When the robot has free time, he has to decide what he wants to do. When the robot has extra money in the bank, he has to decide what to buy and for whom.

Thus, for each situation the decision making process is different. In order to create a universal pathway to make a decision for the current environment the robot must be taught decision making under many situations.

Computer programs inside the conscious

The robot will sense data from the environment through its 5 senses: sight, sound, taste, touch and smell. Target objects are objects the robot recognizes from the environment. The brain of the robot will extract element objects that have strong association with their respective target objects. All element objects from all target objects will compete with one another to be activated in the mind. These activated thoughts are known as the conscious of the robot.

There are two types of conscious thoughts: open activation and hidden activation. The open activations are element objects that are presented to the conscious and the robot is aware of the data. On the other hand, hidden activations are element objects that are not presented to the conscious and the robot is unaware of the data. Extremely complex tasks in memory, such as problem solving, will require both open and hidden activation.

Think of the conscious as a "memory bubble" that takes in and modify data; and the modified data instruct the robot to take action (FIG. 42A - 42B). Inside the conscious is a computer program (or a series of interconnected computer programs) that manipulate data in the conscious. The intelligent pathways extracted from memory and used by the robot generate this computer program.

FIG. 42A

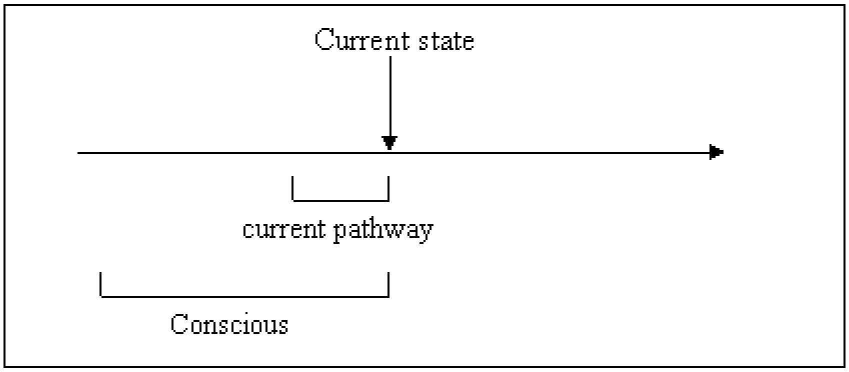
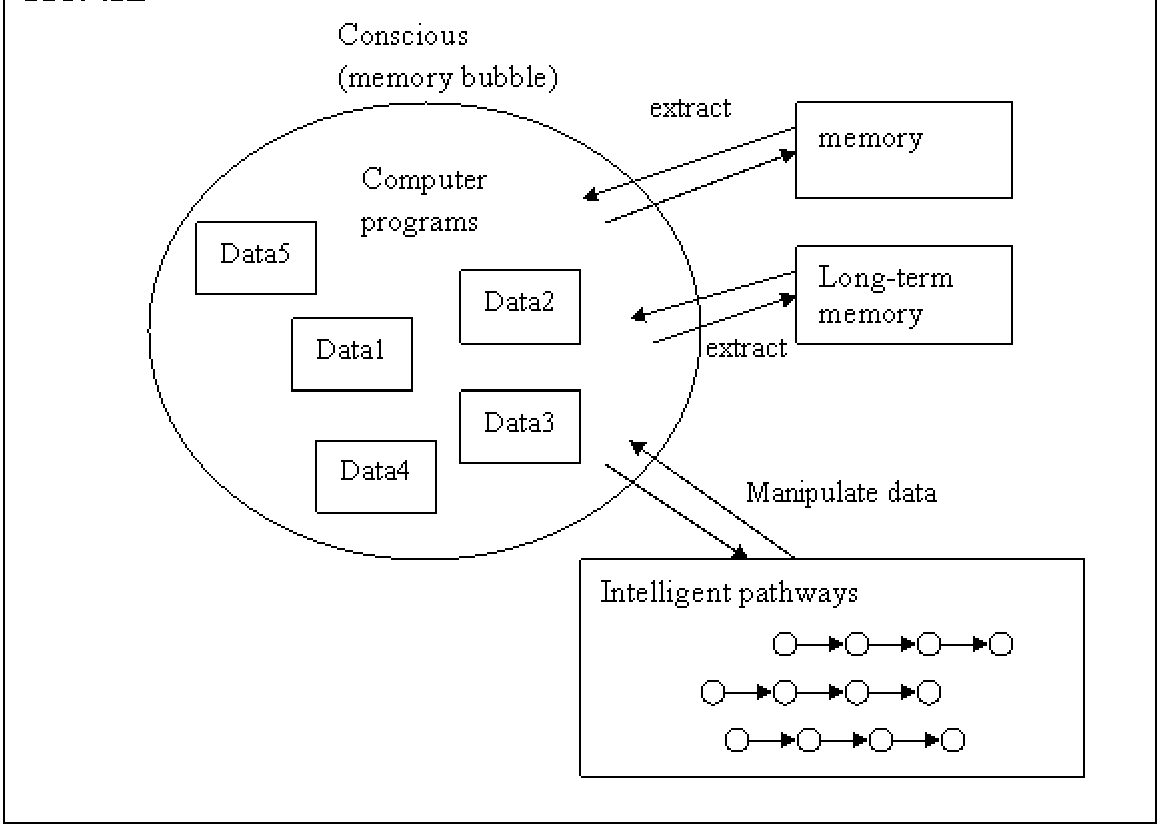


FIG. 42B



The intelligent pathways control the data in the conscious by adding, deleting and manipulating data. They also control the searching and extracting of data from memory. The instructions inside the intelligent pathways will have search functions that will extract relevant data, based on a situation, and put them into the conscious. Some data will be extracted from memory, while other data will be extracted from long-term memory.

There are also other things the intelligent pathways in memory can do. They can control the comparing of data; or analyze two or more data from the conscious. These intelligent pathways can even predict the future and generate a computer program to output future events of what might happen in the future based on the current environment.

The conscious actually does many other things for the robot. Some of these things include: giving knowledge about an object, providing meaning to language, solving problems, answering questions, following orders, planning tasks, solving interruptions of tasks, managing multiple tasks at that same time, following the rules of a game, coming up with imagination and so forth. The intelligent pathways in memory allow the conscious of the robot to do all these things (and more).

Important data in the conscious

The conscious does many different things for the robot. Four of the most important things the conscious does are: 1. manage tasks. 2. establish rules to follow, based on the tasks. 3. solving problems and planning steps to achieve tasks. 4. know identity. Referring to FIG. 42B, there are four containers the computer program in the conscious generated as a result of intelligent pathways: the task container, the rules container, the planning container and the identity container. All data from all four containers influence each other one way or another. For example, the rules will influence what tasks to follow/abort and the planning information will influence what rules to follow or what tasks to do.

These containers are just temporary caches inside the conscious that was generated by intelligent pathways in memory. Based on the current environment, the robot selects an optimal pathway from memory and that optimal pathway has instructions to create containers so that groups of data could be manipulated and logical thoughts and actions can be had by the robot. The intelligent pathways create any type of computer program or discrete mathematical functions to manipulate data in the conscious -- a database system, an operating system to manage multiple threads, a word processor, an image processor, a search engine, a state machine or any software program.

Managing multiple tasks

The robot's brain has a task container that manages tasks. When he wakes up in the morning, knowledge will pour into the robot's conscious in terms of 5 sense data. Most likely sentences will pour into the task container. Natural tasks that the robot does everyday will be the first things to populate the container. Natural tasks will include: brush your teeth in the morning, eat breakfast, eat 3 meals a day, take a shower at night, go to work at 8:30 am, leave work at 5:00 pm and so forth.

Previous tasks that the robot decided to do will also be included in the task container such as decisions made yesterday or last week or last year. For example, if the robot decided 2 weeks ago that he was going on vacation next year December, that task might be included in the task container. If the robot decided yesterday that he was going to eat hamburger for breakfast the very next day (which is today), then the task of eating hamburger for breakfast will be in the task container.

Tasks in the task container can be ongoing as well, if the robot was a student in school, then his primary task in the task container will be to go to school at 7:30 am every morning. If the robot was a business man, then his primary task in the task container will be to go to work at 8:30 am every morning. The tasks in the task container are not fixed and are changeable. For example, if the robot graduates from school and found a job at a local business, then every day that he wakes up the task of going to work at 8:30 am will be there.

Most of the task in the task container are "at the moment" and the robot does a task, then move on to other tasks and so forth. Future tasks are remembered in memory and it's the job of the robot's conscious to remind the robot what tasks to do at certain times. This robot forgets things so sometimes, he forget to do tasks. Sentence commands can also define a task to do at certain times or certain places. For example, if a friend says, "call me when you get home from work", the robot will remind himself when he gets home to activate the thought: "call the boss". The sentence has a pattern, which is: when you recognize your home, activate this thought: "call the boss".

The robot can be fickle minded as well. The robot might decide to go to Panda express for lunch, but when he gets there he decides to go to Burger King, next door instead.

Deciding on tasks, like I said before, is based on at the moment. If the robot is a student in a college studying computer science, and the professor gave him a software assignment, then the robot will plan when he will write the software program. The professor might give the students 2 weeks to finish the software program. The robot knows that he has lots of time so he decides to write the software program tomorrow. If the robot is writing the software program and he has to sleep, he will plan the task of continuing to write the software program tomorrow in his free-time. If the robot is writing the software program for 5 straight hours and he is tired, he will decide to take a 1-hour break, before continuing his work. Within the 1-hour break, the robot will use logic to find something to do to relax, such as deciding on watching a movie or playing videogames.

In other cases, the robot might be in trouble and he has until tomorrow to finish the software program. He will decide to stay up all night to finish the software program. This task will defeat any pre-existing task in the task container, such as sleeping at 9 pm or eating a sandwich before sleeping.

Rules in the rules container

When the robot gets up each day, knowledge in terms of rules will pour into the rules container. These rules dictate the boundaries of certain behavior. In the identity container, the robot has many rules that it follows. These rules include: likes, dislikes, goals, limits, how to do things, what rules that must be followed while doing tasks and so forth.

Since there are soo many rules in life, the robot has to follow rules at the moment. That's where the intelligent pathways are needed. When the robot predicts the future, there are actually "event pools" stored at certain times when events happen. In the event pool are rules ranked in order that activates in the robot's conscious. The computer program in the robot's brain will manage these rules and determine what rules to activate at certain times.

If the robot was playing baseball, and he is batting, the rules of conduct (regardless of how complex they may be) will be populating the rules container. The computer program will manage these rules and activate the most relevant for that given situation. Most of these rules are just sentences that have discrete math properties such as if-then statement, for-loops, while-loops, functions, encapsulated rules and so forth. The most famous is the if-then statements. If the robot recognizes this situation then activate these set of rules.

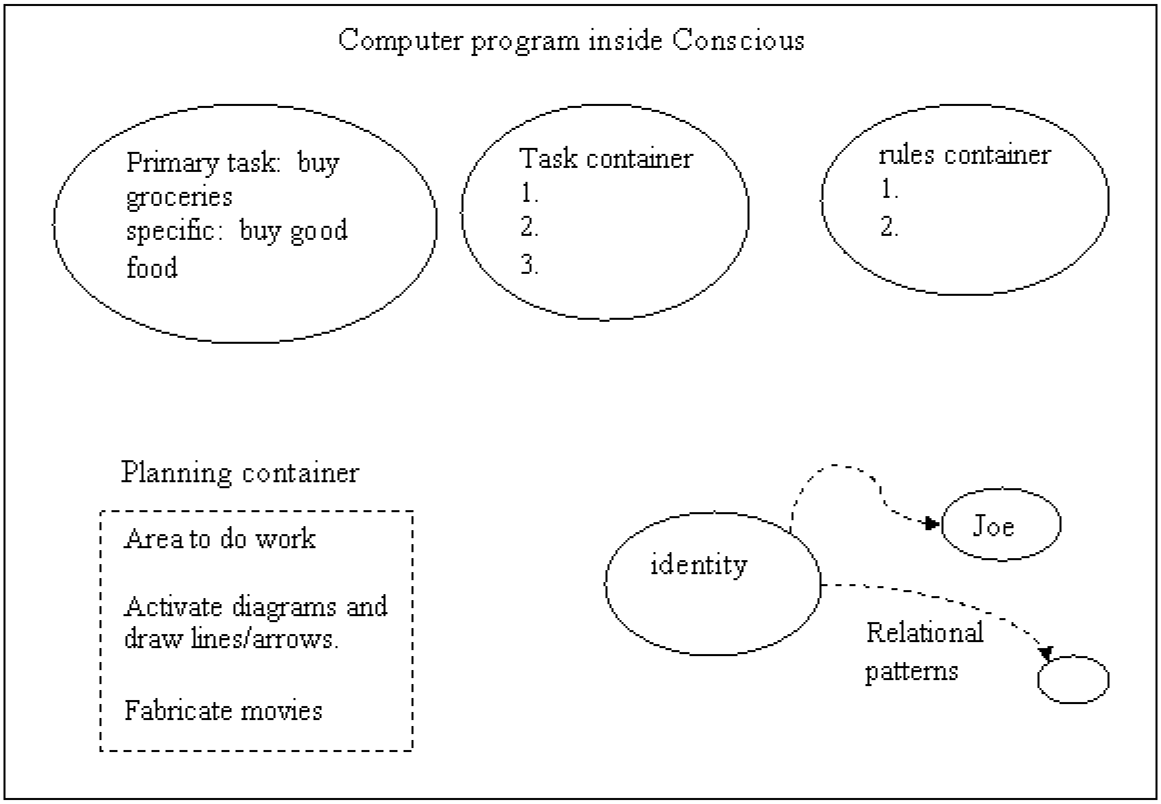
Finally, the task container and the rules container go hand in hand. The robot accomplishes tasks based on the rules. These are the rules that you must follow in order to accomplish this task. These set of rules are needed in order to solve a math equation. These set of rules are needed in order to play chess. The rules are boundaries set that the robot must follow in order to accomplish his objectives.

Deciding to do a task

The decision maker is ultimately the robot and he controls what decisions to make. The intelligent pathways in memory make decisions and extract relevant data from memory to make decisions. Because of this, the robot must have information about its identity (FIG. 43). In the identity container, there exists information that is important to the robot. Things like dislikes and likes are located there.

Depending on the situation, the data in the identity container varies. If the robot is shopping for groceries at the local supermarket, then likes and dislikes of food will be stored there. If the robot was in a clothing department, then likes and dislikes of clothes will be there. When the robot consciously decides to buy groceries from the supermarket, information about food and possible purchases pour into the robot's conscious like a facet valve. The more he thinks about buying groceries the more information is extracted from memory.

FIG. 43



The robot is alive because of all the containers in the conscious. The robot can control what tasks and how to carry out each task. The robot controls the rules that it must follow in order to fulfill the tasks. The identity container contains the likes and dislikes of the robot and what its values are when making decisions. The robot is self-aware because of the control of itself. External objects like people and the environment influences the robot to make decisions, but it's the robot that ultimately makes the decisions.

The identity container also establishes relational links with other currently encountered objects. Family members are established and friends/foes are established. The things that the robot possesses are known in the identity container. Things like the robot has 2 arms, 2 legs, a brain, two eyes, 1 mouth, 2 ears and so forth. He knows that the visual images coming into the eye belongs to him and the things that he hears from the environment come from his ears. The robot knows that he can control most of the physical aspects of its body, such as focusing the eyes on desired areas, focusing the ears to hear certain sound, thinking, moving its body and so forth.

If the robot has money in his pocket, he knows that the money belongs to him. Possession is a learned thing by teachers. Teachers will use sentences to associate how one object can possess another object. If the robot pays money for a domain name (which is a non-physical object), the robot knows that he possesses the domain name. The sentence best describe possession: “if the robot pays for the domain name, then the domain name belongs to the robot”. The word belong is a word that has hidden values that mean possession.

The robot is taught at a very young age common sense knowledge and what decisions benefit itself. Simple decision making are taught to the robot at its early stages. A lesson like selecting between two items based on price is taught. The teacher gives the robot two identical ice cream and asks the robot to select one. One cost 2 dollars and the other cost 8 dollars. Using common sense, the robot will determine that both ice creams are identical, but one is more expensive than the other. The teacher forces the robot to select the cheaper ice cream.

The teachers are there to supervise the learning by guiding the robot with sentences. These sentences tell the robot what is right and what is wrong. If the robot selects the 8 dollar ice cream, the teacher will ask the robot why. At that point, he/she will say: “that is wrong, you have to select the cheaper ice cream”. The teacher will also explain why. The teacher will repeat the lesson until the robot selects the correct ice cream and makes the right decision.

Learning will be gradual, whereby the teacher will teach more complex decision making examples. For example, the teacher might display 5 different chilli can and the prices for each can are different. The robot has to select one of the 5 chilli cans. The prices have to be compared, the presentation of the chilli cans have to be compared, the amount of money the robot currently has, and the brands of the chilli have to be compared.

In the identity container contains a very important fact: “the robot is allergic to onions”. If 1 of the chilli cans has onions in it, the robot should rule that one out. The robot doesn’t care if the chilli can is cheap or the name-brand is well known. The fact that he can’t eat the chilli tells the robot that he won’t be buying that chilli can. Thus, the teacher will teach the robot to rule out possible chilli cans. Next, the teacher will teach the robot to compare and narrow down the choices to 2 or 3. In order to do this, a sophisticated form of logic is required to compare prices and name-brands. Facts about the 4 remaining chilli cans are listed below.

1. $2.50	2. $10.95	3. $3.95	4. $1.99
well-known	best brand	well-known	unknown

The robot will know that he currently has 5 dollars in his pocket. Chilli can 2 will be ruled out because he can’t afford it. The next step is to compare the other 3 chilli cans in terms of price and name-brand. The robot likes the $3.95 choice, but only slight above the other two choices. The robot likes the $1.99 choice because its cheap, but dislikes the taste of the chilli. After an elaborate debate, the robot decided to buy the $2.50 because it is considered not too cheap and not too expensive. He also likes the taste of the chilli.

Decision making is a personal thing, but the teacher is there to help the robot make the best decision and to guide him in the decision making process. If I were to select from the 4 choices above, I would probable pick the $1.99 choice because I like to buy cheap things. Buying cheap is one of the rules that I follow when I buy anything.

In another example, the teacher can put 10 fruits on a desk and ask the robot to select one fruit. In the robot's memory all 10 fruits are stored there and each fruit has powerpoints to determine its desirability. The decision at this point is to search for each fruit in memory and compare their powerpoints. The robot might be taught to select 3 fruit out of the 10 quickly and then compare the 3 fruit in terms of different aspects to ultimately decide on 1 fruit. When the robot does select 3 fruits, he will have to compare aspects of the fruits such as size and taste. Let's say the robot selects a banana, an apple and an orange. The banana is quick to eat and taste good. The apple last longer, but it doesn't taste good. The orange last long and it taste good. Based on these three logical thoughts, the robot will determine which of the 3 will benefit itself. It's obvious from the 3 logical thoughts that the orange is a good choice because it taste good and it last long.

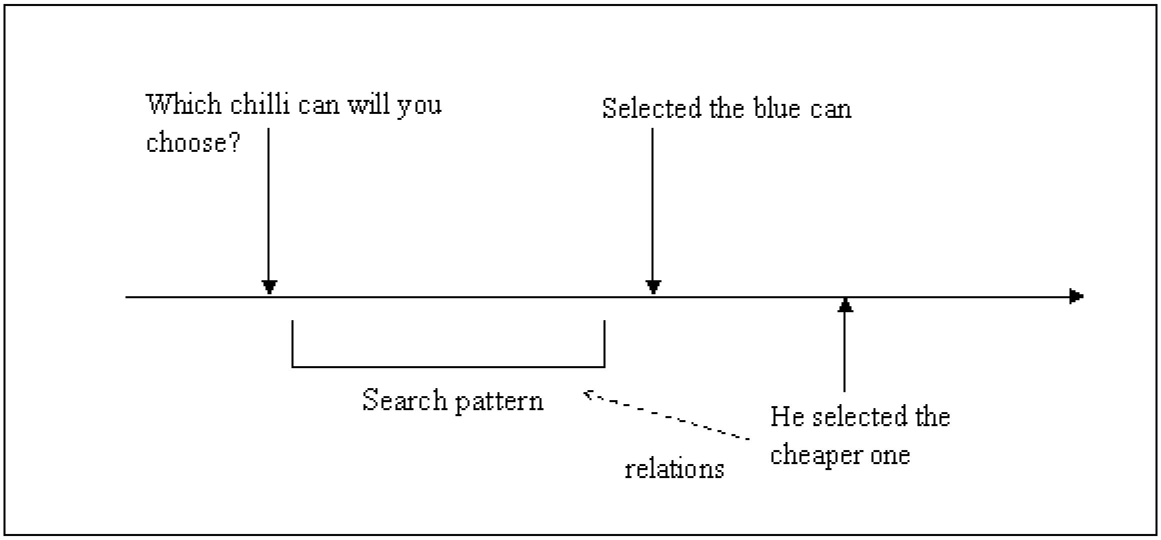
Thus, decision making is simply generating facts about each object and selecting among these facts. Comparing facts can be used as well, such as: "this chilli can is larger in size than that chilli can", "this chilli can's brand-name is popular and the price is cheap", "this chilli can is expensive, but the chilli is very good". Based on these logical facts generated by the items for selection, the robot has to determine which item is the best. In some cases, a decision is a debate in the robot's brain.

Another way that decision making is learned is by doing grade school worksheets. Decision making worksheets are given to the robot and he has to accomplish the assignment. The robot will be graded on how well it makes decisions. If the robot has a good grade, that means he made the right choices. On the other token, if the robot has a bad grade, that means he made terrible choices. The teacher will correct the errors in front of the robot so that he can see what mistakes he made.

Yet another way of learning decision making is by watching TV shows like the price is right. People are put on the stand to make decisions. Some decisions are based on logic while others are based on random selection. If the robot watches 10 scenes and in each scene the task was to select from 3 random doors, the robot will copy the average behavior of the contestants. For example, 7 out of 10 times the contestants selected the 2nd door, then the next time the robot has to select a random door, he will select the 2nd door.

FIG. 44 is a diagram showing input/output again. The robot has to find out the pattern between the input and the output. In this case, the input is the question and the output is the selection. When the robot watches people decide on things, he tends to mimic their behavior.

FIG. 44



Decision making can also be corrected by pain and pleasure. What if the robot buys a chilli can from the supermarket and found out that the chilli had tiny onions in it? The robot will remember the brand and never buy from this brand again. The pain of finding out the chilli he bought had onions in it caused the robot to put negative points on this type of chilli.

Tasks to do and rules to follow

When the robot makes a decision, that task will be inserted into the task container (FIG. 43). If the robot decides to eat lunch at Subway, it will insert instructions, via sentences, into the task container. The computer program inside the conscious keeps track of the tasks and remind the robot when to do these tasks. The computer program generates an operating system to manage tasks. Things like listing linear tasks, solving interruption of tasks, rearranging tasks, executing multiple simultaneous tasks, reminding when to do tasks, erasing tasks, modifying tasks, managing hierarchical tasks are all done by the computer program inside the robot's conscious.

Distant tasks can also be inserted into the task container. If the robot decides one day that he will take a vacation next week Monday, then he will store that task in the task container. Tasks in the task container are not fixed and can be changeable. For example, if Monday comes and the robot is too busy, he can procrastinate and say that he will take the vacation next month.

Tasks can come from both external commands or internal commands. A teacher telling the robot to do a task is considered external command and logical thoughts to do tasks is considered internal commands.

Playing chess example

Traditional chess AI programs rely on generating next step moves and ultimately selecting one move to play the game. A human being doesn't play chess like an AI chess program. In fact, the reason the best chess player in the world lost to a computer is because the AI chess program was cheating. The human being can only see 2-3 steps into the future. The AI chess program can see 300 steps into the future for all pieces.

I will explain how the robot plays chess using human level artificial intelligence. The reader can compare my methods with the traditional AI chess programs and note the dissimilarities.

FIG. 45

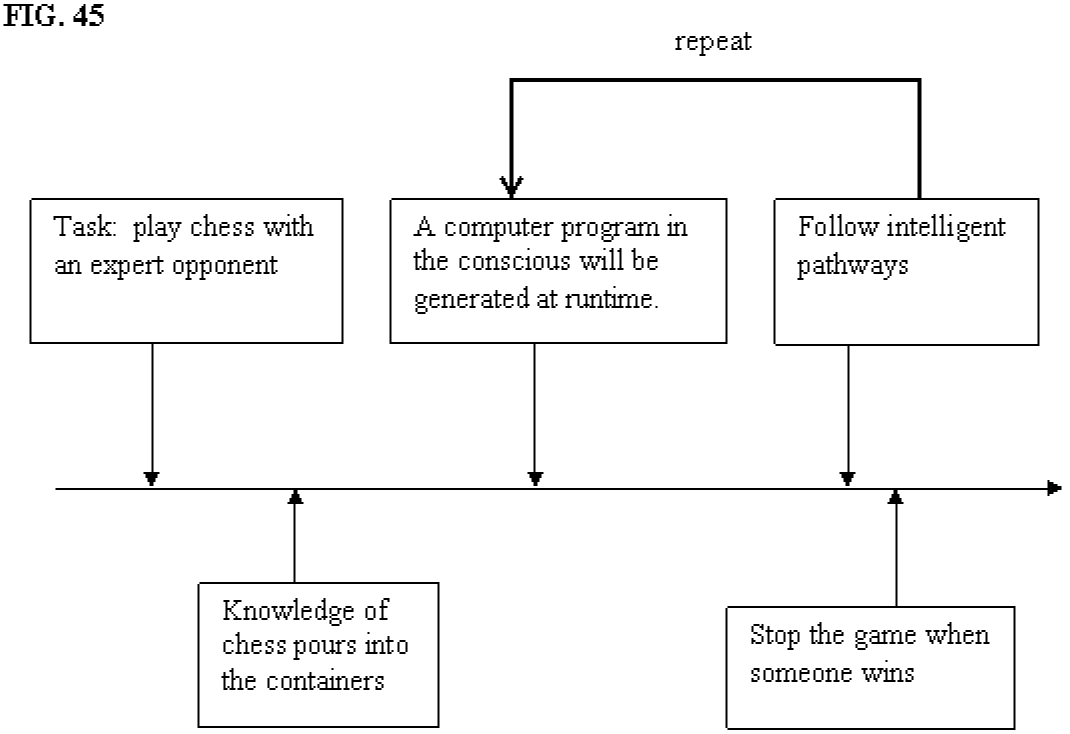


FIG. 45 is a diagram depicting the intelligent pathway that the robot will follow to play the game of chess. The first step is to consciously decide on playing chess. The robot determines that the opponent is an expert. This prompts the robot to play the game with great attention. Not only will the task of playing chess be in the task container, but the other task is to play the game with great attention.

The next thing that happens is that knowledge will pour in from memory into the conscious and populate the containers. The rules of the game are poured in, the objectives of the game are poured in and the strategies to use are poured in. The fact that a data is a task and another data is a task groups them together and stored in the task container. The fact that a data is a rule and another data is a rule groups them together and stored in the rules container. Thus, data coming into the conscious is organized based on association. The data should be structured in a hierarchical manner so that only important data enters the conscious.

As the robot play chess during runtime, the intelligent pathways selected will define the computer program in the conscious to play the game. Every action is based on at the moment. The only exceptions are the rules of chess. The rules container contains rules that might or might not happen. These rules are extracted because the robot's brain thinks that the data might be needed in a given situation. One of the rules to follow is that the player (the robot) and the opponent have to take turns when playing the game. Each can only make one move per turn. This rule is followed by the robot throughout the game.

The computer program will do things for each given moment in time. When the opponent moves, the robot will plan strategies and guess the opponents next move. When it's the player's turn, he has to plan several future steps and select only one to act. The player knows that one of the rules of playing chess is to make your move quickly. Thus, the robot has a time limit to make his move.

The intelligent pathways will generate an optimal computer program to manage tasks for the game of chess. The game will continue and the player and opponent will keep taking turns until one wins.

One of the strategies used to play chess is to predict the future by imagining a fabricated movie of 3 future chess moves (one future move per chess piece). The robot has to fabricate 3 future chess piece and decide which of the 3 moves to select. The method to do this is to imagine the pieces moving in this space or that space. While the imagining takes place the robot is also imagining possible opponent moves. The brain manages these fabricated future movies and selection of each future move is done by comparing with other fabricated future moves.

My guess is that the brain fabricates one future move and it fabricates another future move, compare the two fabricated moves to determine which is better, then selecting one of the two future moves. Next, the brain will fabricate another future move compare it with the previous selected future move and determine which is better. Previous unselected future moves can be reconsidered too. The robot's brain will do this until he has to act.

Each human player uses their own personal methods, but most chess players use the method described above. Some expert players can actually fabricate 3 step moves for each chess pieces and determine which is better.

While the robot's brain is managing the task of fabricating future moves, it will also follow the rules of chess. If the robot is trying to fabricate a future move, he has to first identify the chess piece and determine how it can and can't move. The horse piece can only move in an L movement. This rule will activate at the moment the robot wants to fabricate a future move using the horse piece. If the chess piece is the pawn, he can only move one step forward. While making decisions on which fabricated future move is better, the robot has to consider the worth of the chess piece. The king is a very important piece because it can do all the moves that any of the other pieces can do. The pawn is the weakest because it can only move forward. During the decision making process, the robot has to weigh all the fabricated future moves and only decide on one action. The computer program inside the conscious will also remind the robot he has a limited time period to decide.

Playing chess is a lot like shopping for groceries. There is an internal debate that determines which logical thought to choose. Lessons from teachers guide you to decide on things.

In terms of learning how to play chess, a teacher or a friend will teach you how to play the game. He/she might tell you this: "imagine several future moves and select only one". However, the learning of chess comes from a universal pathway that can play any game. The teachers have taught you to teach yourself how to play an unknown game. When you don't know how to play a game you have to first learn the rules and objectives. This can be done by reading instruction manuals or having a teacher teach you.

Learning the objectives and rules of a game is one thing, but going through trial and error to determine what strategies are good and bad is another. Trial and error makes the robot more skilled at the game. Trial and error will keep the good strategies and delete the bad strategies. The robot can also be taught to generate strategies and try them to see if they are good or bad strategies.

Also, teachers can teach the robot to identify bad things or good things in the game. Having lots of chess pieces on the board for the player is a good thing. Having less chess pieces on the board is a bad thing. Having strong chess pieces on the board is a good thing and having weak chess pieces on the board is a bad thing. By understanding the bads and goods of the game, the robot can learn to be better in the future.

Playing chess isn't a very good example to use in terms of identifying bad and good things in a game. If you play videogames, it is easy to identify the bads and the goods. When a player dies, that is considered bad. That bad move will give the player pain, which is necessary to improve in the future. Bad things in a videogame will include: dying, losing life energy, missing a powerup, getting hit by an enemy, jumping into a lake, losing the game, not having high scores and so forth. The good things in a videogame will include: having high energy level, gaining powerups, avoiding enemies, defeating enemies, having high scores, winning a level, beating the boss, hurting a boss and so forth.

If you play videogames, the identifying of good and bad things is a vital part of the game. If the player does good things, it will steer the game in the right direction. If you do bad things the game will steer the game into the wrong direction. The robot should also have the knowledge to identify the objectives of the game. In some unknown games, I have to play many times before I logically identify the objectives. For example, in Donkey Kong, I didn't know what the objective of the game was at first. I tried something and died, then I tried something else and I died again. However, by trying something and identifying what is good and bad in the game, I was able to use logic to identify the objectives.

The point I'm trying to make is that when the robot has to play an unknown videogame, the identifying of good and bad things in the game will help it generate better strategies to play the game. If the robot has good strategies to play the game, then he will eventually succeed in passing all the levels (refer to my previous books and patent applications regarding how the robot plays videogames such as contra and the legend of zelda).

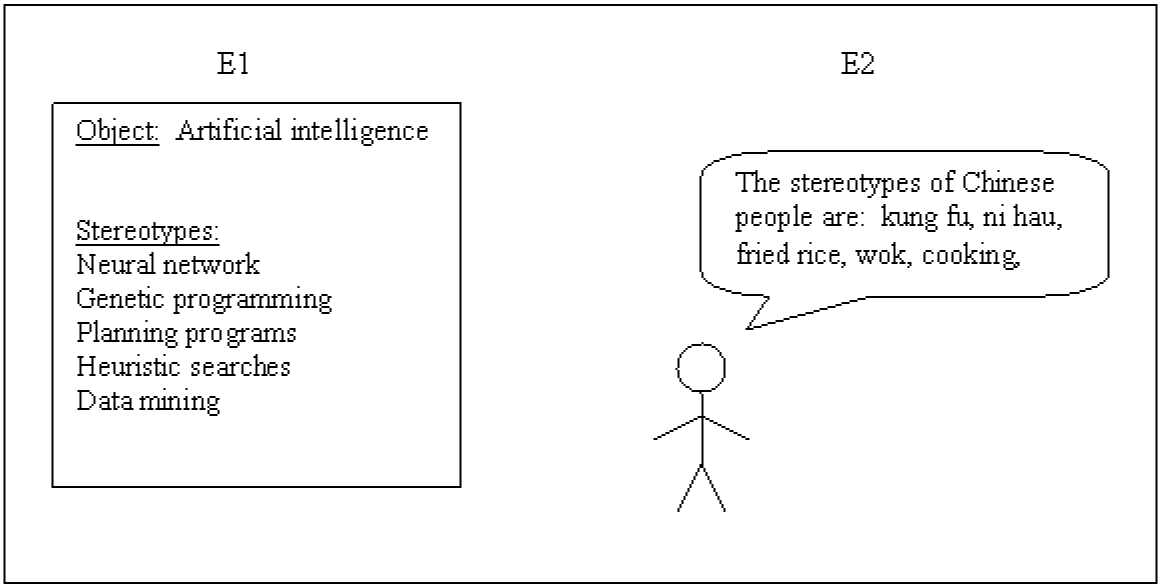
Chapter 10

Accomplishing various tasks in the real world

This chapter will emphasize on tasks that the robot can do in the real word, such as: writing essays, writing software programs, playing videogames, giving speeches, having a conversation, and so forth. But before I get into each task, I must explain the gradual forming of intelligent pathways in memory. Intelligent pathways in memory go through a bootstrapping process, whereby new data is built on top of old data. The robot uses trial and error and school lessons to keep pathways that lead to pleasure and forget pathways that lead to pain.

The search function to activate related element objects from a target object can actually be learned. If the target objects were: cat and dog. An element object activated might be: “cats don’t like dogs”. This fact is a stereotype of the two target objects: cat and dog. Below are diagrams showing target objects and their strongest element objects. Diagram E1 is a paper that has writings on it, showing what the strongest stereotypes are to the target objects. Diagram E2 is a spoken sentence that tells the robot what the strongest stereotypes are to the target object. FIG. 46 just shows the different medias that can be used to teach the robot what the strongest element objects are to a target object.

FIG. 46



In the future, when this pattern is formed and an intelligent pathway is created, the robot might be able to modify connection weights between the target object and its element objects. Intelligent pathways might be able to change data in memory according to the diagrams in FIG. 46. For example, in E1, the target object is artificial intelligence and the stereotypes are element objects that have strong association with the target object. What if the robot had only the first 3 element objects stored in memory? Based on the pattern, the robot's brain will add in the last two element objects and store that in memory. Maybe the order of how strong element objects are can be changed because of observing E1. The robot might have neural network as the least strong element object. After looking at E1, the robot's brain made element object, neural network, stronger.

Diagrams E1 and E2 can also tell the pathways that when the target object is identified by the robot, the stereotypes will activate in this linear order. In the future, when the robot recognizes Chinese people, stereotypes will activate such as: kung fu, fried rice, wok, and ni hau.

The above method is important for my next lesson.

1. Writing a sentence (A1)

Teachers in the past has taught the robot simple things like the ABCs, nouns, verbs, phrases, subjects, predicates, pronouns, numbers, and so forth. The identification of sentences are also learned, such as designating sentences into these categories: declarative, question, exclamation, comment, etc.

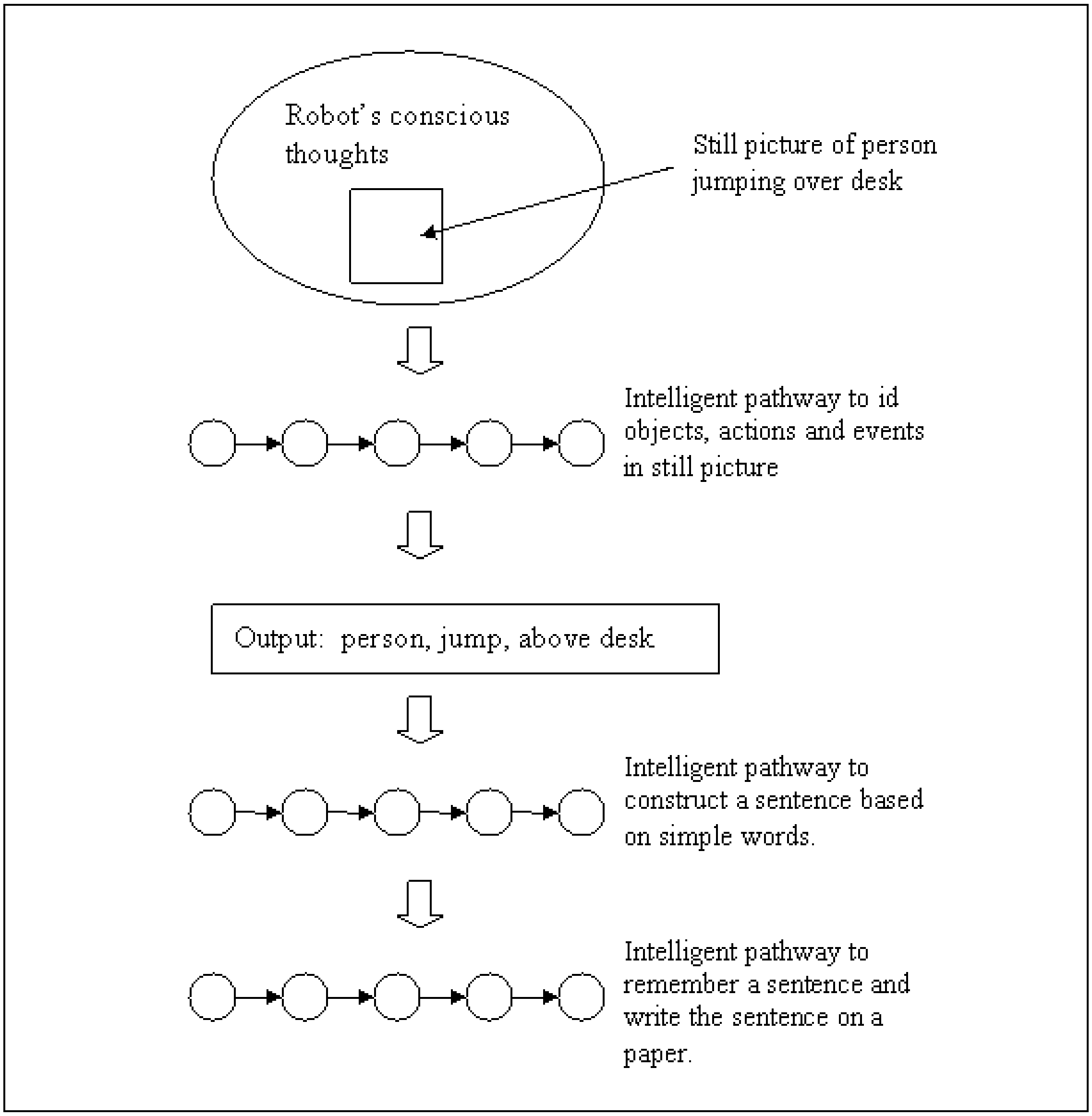
The robot should know what objects are nouns, verbs, and adjectives. Learning sentence structure is also vital. From this point, the robot has intelligent pathways to construct sentences based on its thoughts. The robot's conscious will activate images, sounds or 5 sense data and the robot has to translate these activated thoughts into sentences.

The translating of thought to sentences is actually learned previously. A teacher will show a picture and it will describe the picture using sentences. For example, a teacher will show a picture of a person jumping over a table; and the teacher will ask the students what does the picture mean? Either students or the teacher (or both) will give the correct answer. The teacher will say: this picture is depicting "a person jumping over a table".

How does the last lesson relate to writing a sentence? If the robot activated a picture of a person jumping over a desk, he has to translate this image into a sentence. In order to do that, the previous lesson on identifying objects and actions in an image will be used. The robot will identify the scene as a person jumping over a desk.

FIG. 47 is a diagram depicting the process of translating conscious thoughts into sentences. First the robot activates a thought. These thoughts could be any 5 sense data: sight, sound, taste, touch or smell. It could be a still picture or a movie sequence. An intelligent pathway in memory will be used to identify objects, events or actions in the still picture. A simple id would be to output words like: person, jump, above desk. Next, another intelligent pathway will be used to convert the words into meaningful grammar sentences. The robot knows that the subject comes first and then the predicate. The subject will start with a noun. Then, the predicate will include a verb that the noun belongs to and the other noun that is being affected. At the end, the sentence: "the person jumped over the desk" is constructed. The robot will use another intelligent pathway to remember the sentence and, using his hand, to write down the sentence on a paper.

FIG. 47



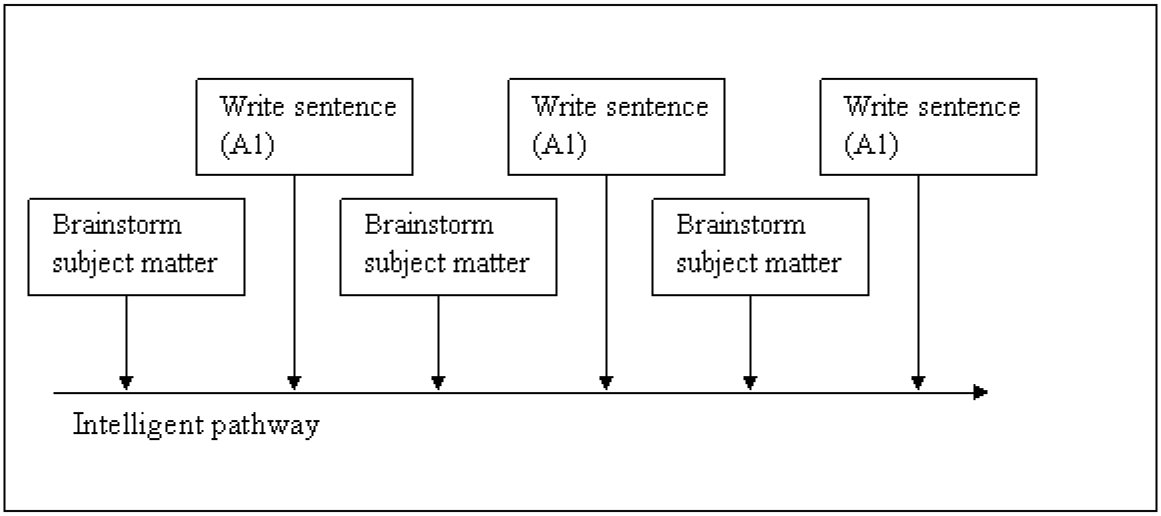
Images and movie sequences convey a thousand words. This is why the conscious activate images more than any of the other 4 senses. If a question is asked such as: "what color is a heart", the robot will activate an image of a heart and use an intelligent pathway to analyze the color. The output is the color of the heart. If the robot was asked another question such as: "what shape does a heart look like in a human body?", the robot will activate a heart from a human being, analyze the image and output its shape. Thus, the image of a heart contains many data. Using images to store data is better than using voice sequences. A voice sequence is a sentence spoken by someone. "the color of a heart is red" is one voice sequence. If we store all facts related to a heart using voice sequences, it would overwhelm the robot's memory. The image of a heart contains many facts related to a heart.

2. Writing paragraphs (A2)

Writing paragraphs will encapsulate writing sentences. A paragraph comprises a series of linear sentences. The paragraph gives facts about a subject matter (an object, event or action).

Teachers will teach the robot at the beginning of the paragraph to brain storm on a subject matter. Maybe the goal of the robot is to write facts about the relationship between brain injuries and football. During the brainstorming part, the robot is searching for specific data on football and brain injuries. The target objects are football and brain injuries and the robot has to search in memory for element objects related to the two subject matters. The outputs are simple facts pouring into the robot's conscious and an intelligent pathway will select which fact to use for the paragraph (FIG. 48).

FIG. 48



Sometimes, writing a sentence is done so quickly that, when an idea is selected, the sentence is automatically constructed. It's like speaking. When ideas pop up, the words just come out of a person's mouth. Minor things like grammar checking and checking sentence structures are bypassed.

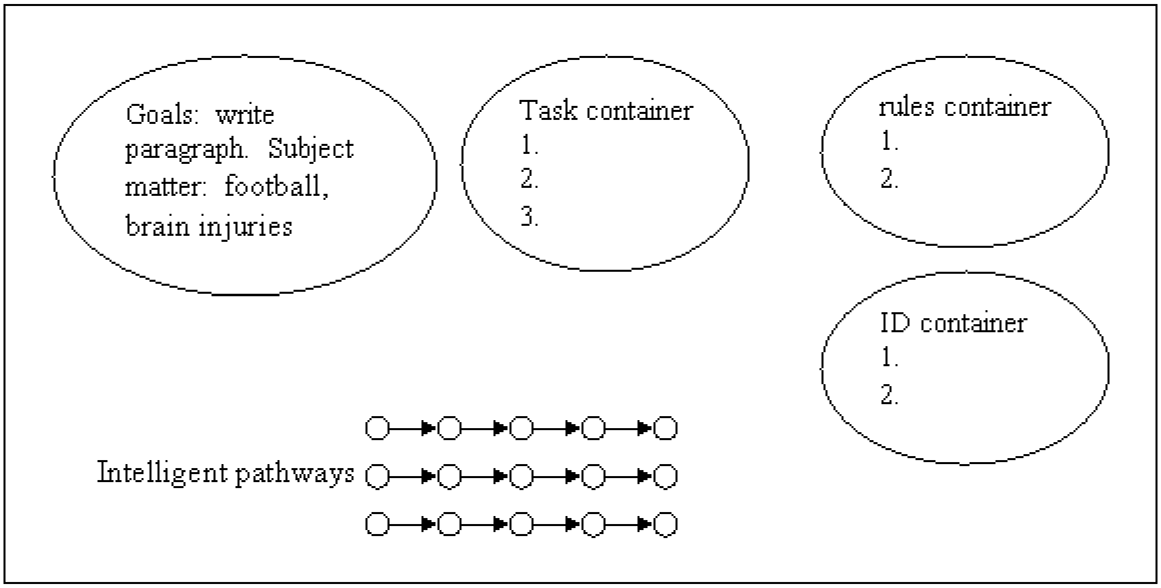
Once the first sentence is done it will continue to brainstorm more ideas and construct the second sentence. Other facts will activate such as: "the second sentence should continue from the first sentence" or "don't use the same sentence structure from the previous sentence". While the robot is writing the paragraph, the robot's conscious gives him rules and goals to do in order to complete the task.

In some cases, the robot might have 3 good ideas and he will use the first idea and remember to use the 2 other ideas in later sentences. The computer program inside the conscious will organize the ideas so that the robot knows which ideas to use first, second and last. For example, the robot selected 3 ideas in memory and constructs the first sentence. Next, the robot will remember the second idea and proceed to construct the second sentence (at the beginning of the second and third sentences, the robot doesn't have to brainstorm any ideas). Finally, the robot will remember the third idea and proceed to construct the third sentence. These ideas can be from fragmented movies that activate in the robot's conscious. The robot's brain simply selects data from these movie sequences.

The three ideas can be from one still picture or one movie sequence.

The intelligent pathway in FIG. 48 is only one pathway used to write a paragraph. Referring to FIG. 49, when the task of writing a paragraph is on the robot's mind, he will automatically generate a computer program to accomplish the task. General tasks and rules will pour into the robot's conscious. As the robot selects specific intelligent pathways to write a paragraph, the tasks and rules located in their containers become more detailed. The intelligent pathways provide more knowledge of what the robot should do in the future to accomplish writing a paragraph.

FIG. 49



These intelligent pathways actually define an optimal computer program to write a paragraph. When should the robot do this and do that? What rules does the robot follow for a given task? What are the limited choices in a given situation? The computer program inside the robot's conscious will manage the complexity of the task.

The robot does things sequentially in order to write a paragraph. Specific intelligent pathways are selected at certain times to do things. If you observe the complexity of a software that can write a book, you will notice that the software has thousands and thousands of individual functions. There is a function to id words, there is a function to check grammar, there is a function to string words together to form sentences, there is a function to search for ideas and so forth.

In a human brain, the complexity is managed by using specific intelligent pathways for given situations. For example, when the robot has to brainstorm ideas, only functions that search for ideas are being executed in the robot's conscious.

Writing a paragraph is very complex because there are multiple layers of tasks being done at that same time. The computer program inside the robot's conscious will manage all layered tasks. In one task, the robot wants to write a paragraph. Encapsulated in this task are other subtasks such as brainstorming ideas, translating ideas into sentences (A1), and checking for grammar errors. If grammar errors are found, the computer program will find a way to correct the problem. After finishing the paragraph, the robot might have to look at the overall sentences in the paragraph and to make sure that everything is done correctly, according to writing rules.

Doing tasks, following rules and solving interruptions of tasks are all managed by the robot's conscious. The robot's brain will select optimal intelligent pathways from memory at each iteration of the task: writing a paragraph. These intelligent pathways will form an efficient computer program inside the robot's conscious to accomplish the task in a linear manner.

3. Writing a book (A3)

Writing a book is a much more complex task than the previous two tasks (A1 and A2). Intelligent pathways in memory build on itself and become more complex. In terms of writing a book, the teacher must teach the robot to do an outline at the beginning of the task. The outline will be a summary of what the robot has to do in order to write a book. The outline can be a hierarchical order, whereby the data goes from general to specific. Some areas of the outline can be blank so that the robot can fill in the missing data during the writing of the book.

The rules and goals of writing a book are much more complex than writing a single paragraph. The computer program inside the robot's conscious has to provide the correct rules of writing a book. These rules include: "how long chapters should be, what the overall content of the book will convey, how much revisions should be made before completing the book, when to check the outline, how to modify the outline, how to solve conflicts in the outline and so forth". The rules should be about the overall book and not just about a specific chapter or section.

FIG. 50 is a diagram depicting an intelligent pathway to write a book. These are general steps that are required to write a book. There can be hundreds of other intelligent pathways learned to write a book. However, the diagram illustrates the most likely steps to write a book. First, the robot has to brainstorm ideas for the book. Next, an outline is created that summarizes the chapters in the book. The construction of the outline is taught by English teachers. Next, the robot has to do "work" – he has to follow the outline and write the chapters. After completing the task of writing the chapters, the robot has to check for grammar errors in all pages of the book. He must visualize how the reader will respond to the book and what the robot should do to make the book better. The checking of the outline, checking grammar errors and checking the overall book can be done in any sequence order. The final step is to do 3 revisions of the book before it can be published.

FIG. 50

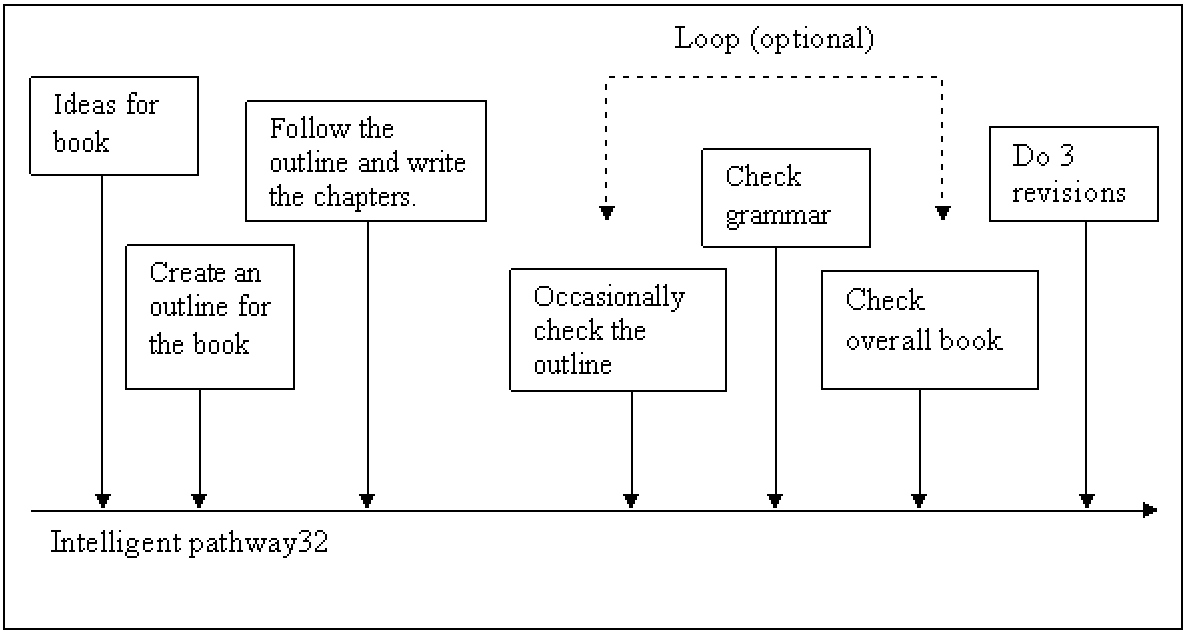


FIG. 51

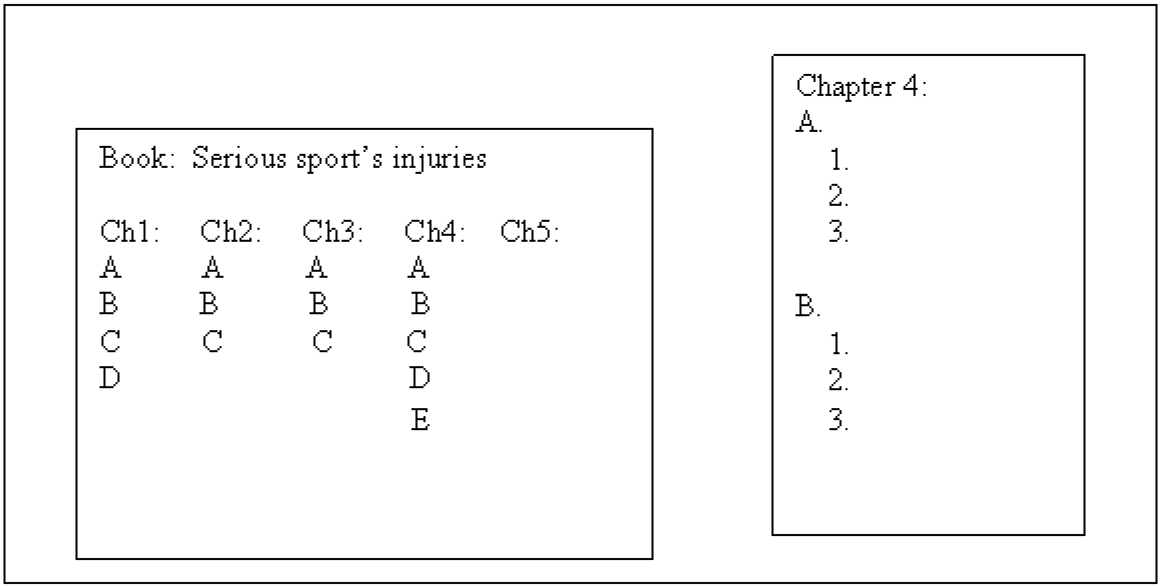


FIG. 51 is a diagram depicting the outline of the book. The robot will generate the outline and he will write the chapters according to its summary.

In English books, the steps to writing a book can be given and the robot has to learn the steps and generate its own linear intelligent pathway in memory. For example, the robot is reading an English book and the chapter is explaining the linear steps to writing a book. The robot will create intelligent pathway32 in memory based on what he is reading. This pathway will be further strengthened when the robot is given assignments to write a book. Over the years, after writing many books, the intelligent pathway32 will change and new steps are added or old steps are deleted or modified. Trial and error by the robot will determine how intelligent pathway 32 will evolve.

Gradual learning

Writing a book is extremely complex and in order to write a book (A3) the robot must first learn how to write a paragraph (A2) and learn how to write a sentence (A1). In the writing of a book, A2 and A1 will be used many times in addition to many other intelligent pathways. It took a human being a total of 20 years to fully learn the English language and to write a full book.

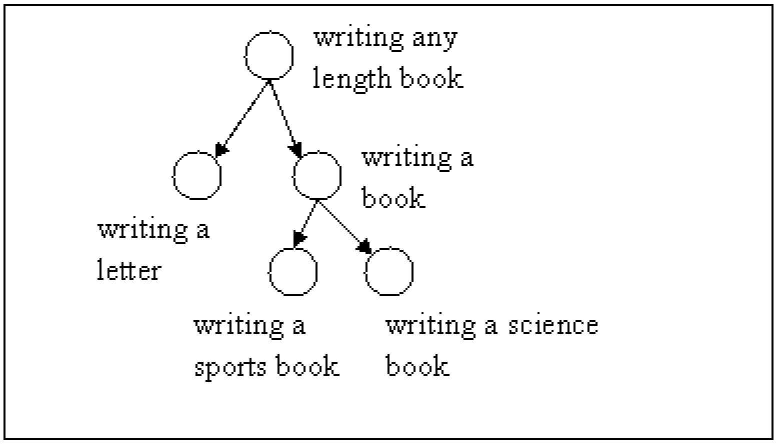
Many years of assignments and tests and lectures are needed to form the intelligent pathways to write a book.

A more simple type of task to A3 is to write short essays or to write letters. The task of writing a book comes from simpler tasks like writing a short essay. In fact, all tasks of writing anything are structured in a hierarchical tree.

Writing a book à writing a short story à writing an essay à writing a letter à writing several paragraphs à writing one paragraph à writing a sentence.

Writing a short story is encapsulated in writing a book and writing an essay is encapsulated in writing a short story. The intelligent pathways are structured in a hierarchical manner and there might be a general computer program to write any length book (short or long). This general intelligent pathway is located at the top of the hierarchical tree. At the bottom of the tree are specific intelligent pathways to write specific types of books (FIG. 52).

FIG. 52


writing any length book
writing a letter
writing a book
writing a sports book
writing a science book

Chapter 11

4. Giving speeches and having conversations

Giving a speech inherits most of the intelligent pathways for writing a book. Intelligent pathways in memory are structured hierarchically. Writing sentences on paper is similar to speaking sentences to a public. They both share commonalities.

FIG. 53 is a diagram depicting the hierarchical tree that shows the relationships between writing a book and giving a speech. The common intelligent pathway used is translating thought to sentences (FIG. 54). The intelligent pathway to write a book is used to write a book and the intelligent pathway to give a speech is used to give a speech. Both intelligent pathways inherit the root node, which is translating thoughts into sentences. For each task, there are slight differences for T4 and the robot's conscious have different rules that it must follow.

FIG. 54

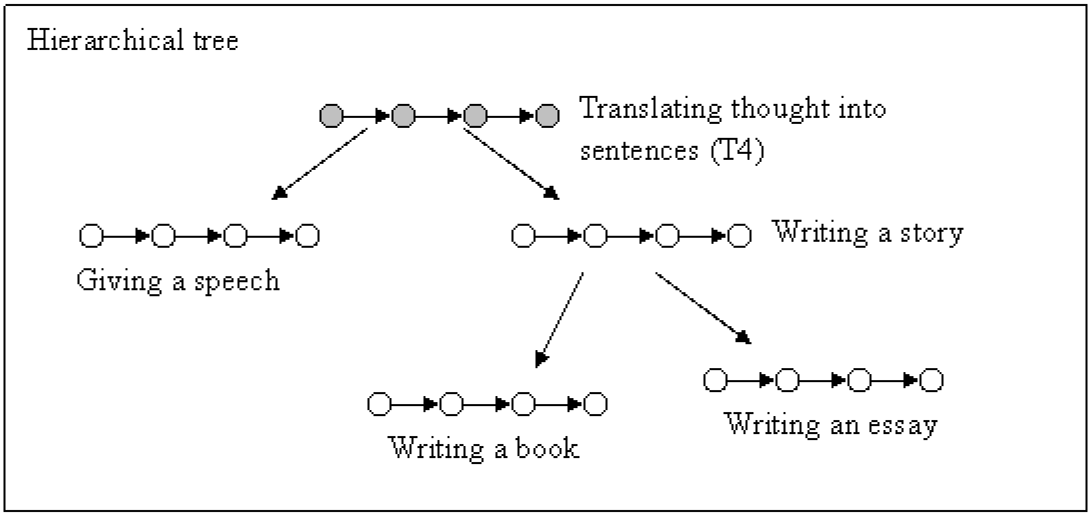
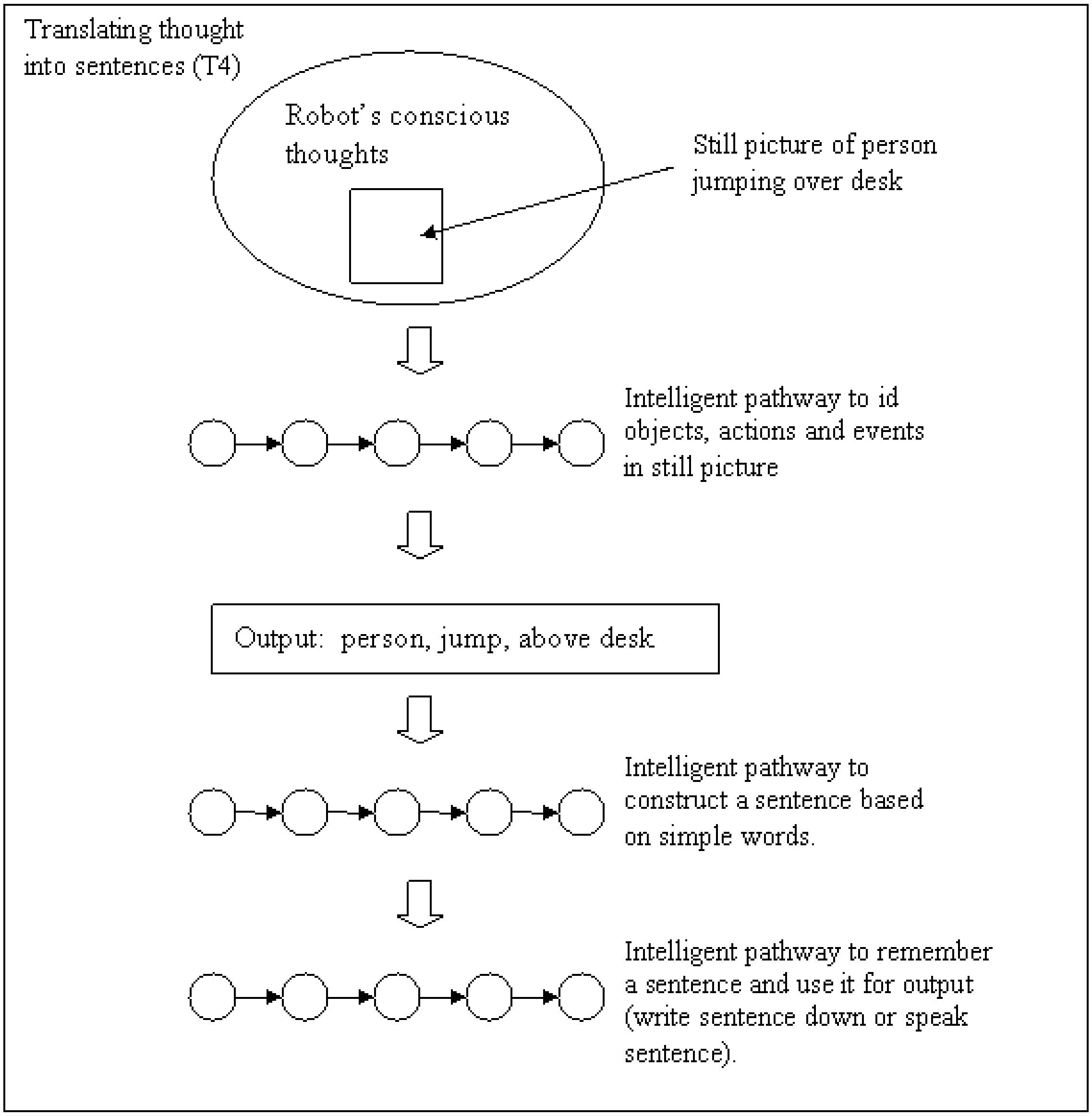


FIG. 55



In intelligent pathway T4, the last instruction will point to a different pathway to do things according to the task. If the robot was giving a speech, the pathway will be to open its mouth to speak the remembered sentence. If the robot was writing a book, the pathway will be to move his hand and write the remembered sentence on a paper.

Tasks and rules are very important to the robot when giving a speech. The robot has to know the people in the audience. A speech for an audience comprising children will be different from a speech for an audience comprising politicians. With the children audience, the robot will have rules that it must follow, such as: "use elementary words", "don't use bad language", "speak slowly", "make the audience laugh" and so forth. The job of the intelligent pathways changes as the robot understands the current situation. The intelligent pathways modify the tasks and rules and knowledge of the computer program inside the robot's conscious.

This is why the robot has to learn speeches in a variety of environments. He should learn to give a speech in front of politicians, school teachers, the public, in a classroom, in a family gathering, in a concert and so forth. At least, learn the knowledge of how to give a speech under various audiences. College classes on speeches are a good place to learn how to give a speech. Thus, the robot will be able to give speeches under any situation. He can give a speech in a news conference, or in a family party or in a debate class.

In the hierarchical tree, under give speech, the robot will be able to select the appropriate intelligent pathways for that given situation. All speeches will comprise very similar general instructions. The specific intelligent pathways will have instructions to do specific types of speeches.

5. Writing computer software programs

The writing of computer software is very similar to writing a book. Although, the rules of grammar and the rules of software engineering are different, they both have rules that you have to follow according to specific tasks.

The robot has to have knowledge about object-oriented programming, which includes discrete math functions like if-then statements, for-loops, and statements, or statements and so forth. Learning classes, methods and recursions are also a must to write meaningful software.

Once the knowledge and the rules of software engineering are stored in the robot's brain, he has to know the steps to write software. These steps can be read from basic programming books or learned through trial and error. When the robot takes computer science classes, the professor will give assignments to the robot, ranging from basic to complex. A simple assignment like writing a java applet that will output the text: "hello word", might be the first assignment. As the robot learns more about computer science and actively write software, he will get better and better. By the time he graduates from college, the robot will have the knowledge to write complex software.

By the time, the robot gets into college, his brain already has the capacity to learn anything. Learning how to write software or learning how to build bridges or learning how to draw the blueprints to a city is possible.

It might of taken the robot 20 years to develop the skills to write a book. However, when it gets to college, it will take the robot only several months to learn how to write a decent software program. The reason why is because the intelligent pathways are so well developed that it can teach itself how to learn. The way the robot store and retrieve information has been optimized.

FIG. 55

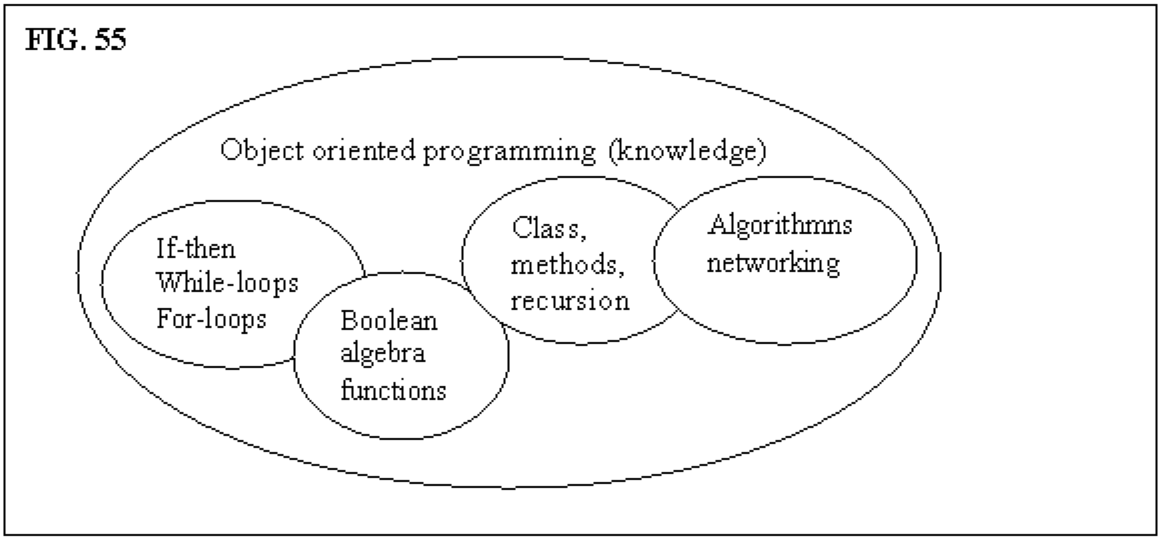
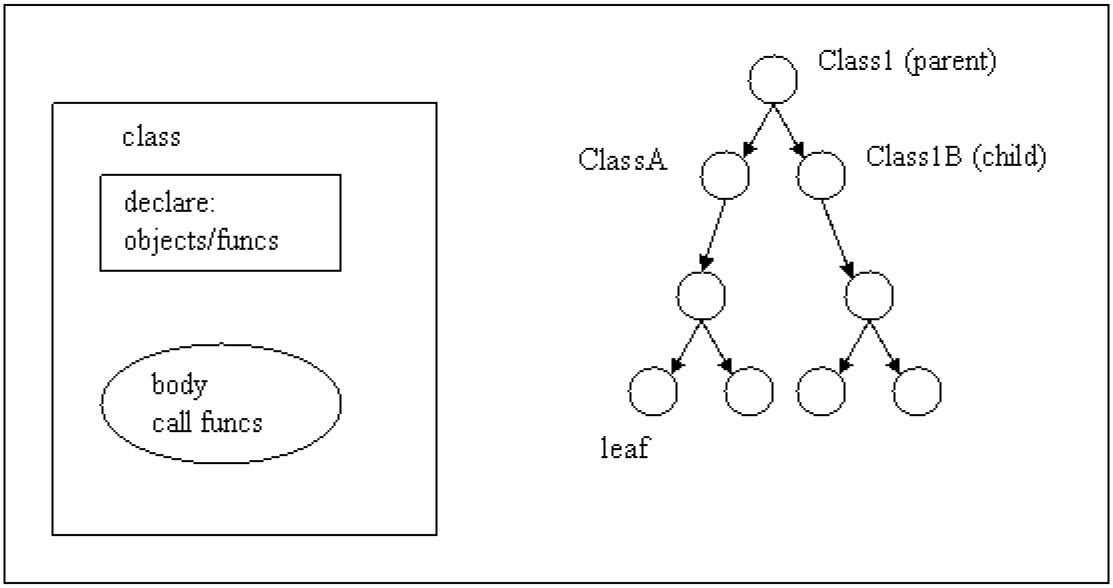


FIG. 55 is a diagram showing what knowledge the robot should know about computer science. These various knowledge will be needed to translate an outline of a problem into a software program.

The knowledge of object oriented programming will be configured in memory according to how knowledge in science books are configured (in previous chapters). Teacher lectures of chalkboard demonstrations will help to organize the data in memory. Also, visual diagrams in the computer text books will help to organize the data in memory. When the teacher draws diagrams on the chalkboard and finger points to numbers and shapes, he is providing a very strong method for the robot's brain to configure data in an organized way.

FIG. 56



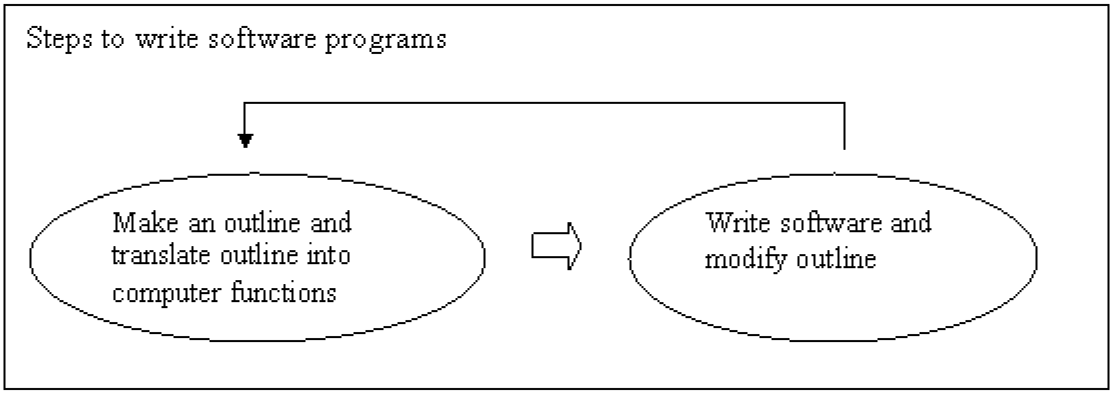
Referring to FIG. 56, classes and functions can be represented by a square diagram with inner shapes representing certain elemental parts. For example, the square at the top is the declaration, where the objects and functions are defined. The circle at the bottom is the main body and that is where objects are used and functions are called.

In the second diagram is a visual diagram of recursion. The rootnode is the parent, which is made up of a class and the child node is an inner copy of the parent. Other data and facts about binary trees can be included in this diagram. During the learning phase, the teacher is drawing this diagram and he is pointing here and there and writing down things on the diagram. The robot will store static as well as linear data from the lecture and configure data from the lecture in an organized manner for memory storage.

As far as accessing data from the knowledge learned about object oriented programming, the data is stored in memory as 5 sense data (either as static data or linear data or both). The intelligent pathways to analyze is the key to finding out how things work. For example, before writing a software program, the robot will activate the class diagram and analyze the diagram. Information extracted from the intelligent pathway, via search patterns, will determine the fruit of the search. The robot might think: "oh, that is where the declaration goes or that is where the body of the problem should be or that is where the functions are called". Another example is the human hand. If the teacher asks this question: "what does a human hand look like", the robot will extract a visual image of the human hand and using intelligent pathways in memory to output what the visual hand looks like. There is no data in memory that explains the detailed aspects of a human hand. The intelligent pathways analyze the image to extract the information of what the human hand look like, during runtime.

FIG. 57 is a diagram depicting the steps the robot will take to write software. The robot will be given a problem by the professor. He will take the problem and write an outline for a software program that will solve the problem. Next, the robot will translate the outline into software functions. Finally, he will test out the software program so that it is working correctly; and solves the problem given by the professor.

FIG. 57



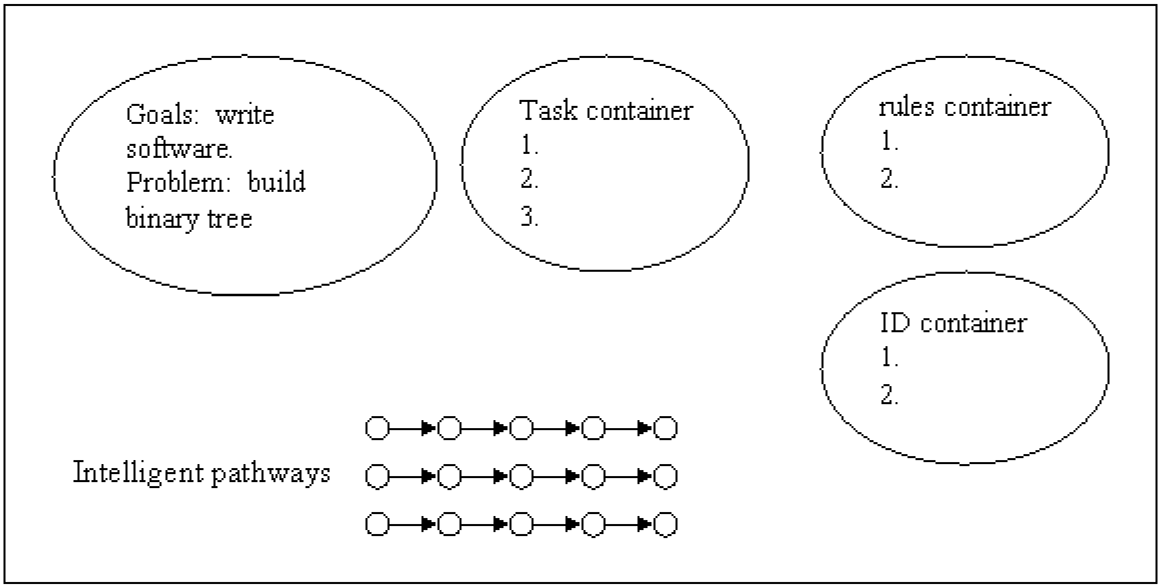
The outline of the software program is very important. At this stage, the robot needs to know how the software program will work and use computer science knowledge to translate the outline into computer functions. For example, if the robot is writing a computer program on a binary tree, he has to use recursive classes. The outline of the binary tree will probably be a diagram of a hierarchical tree. If the computer software is a chat network, the outline might be multiple interconnected nodes with a hub in the middle.

For me personally, I would draw out the outline of the software program I am planning to write. I do this by drawing circles, squares and connected lines. I also write down the functions of each node and what data is contained in each node.

Referring to FIG. 58, when the robot decides to write a software program, knowledge will start to pour into its conscious. At first, a general computer program will be used to manage tasks and rules. As the robot follows intelligent pathways in memory to write software, the computer program inside the robot's conscious will get more specific. It can write any software program regardless of how complex it might be.

The intelligent pathways contain the step-by-step instructions to manage multiple layered tasks and to be aware of possible rules that the robot needs in order to accomplish tasks in the task container. Task interruptions will be solved by the robot's conscious as well. There are limits to how certain computer statements work. The robot has to be aware of these limits. There might be number limits to a for-loop because the program might run out of disk space. Or certain variables can only use certain data types. The robot has to know these rules when writing the software.

FIG. 58



Learning how to write software and understanding the knowledge of writing software is one thing, actually doing the assignments is another thing. The robot has to write software starting from simple software programs to complex software programs. The robot's brain learns knowledge through trial and error and the pathways have to go through a bootstrapping process, whereby intelligent pathways build on itself.

Here is a list of simple to complex software programs:

1. write a program to output “hello world”.
2. write a program to convert Fahrenheit to Celsius.
3. write a database program to store customers using a binary tree.
4. work in a group to write software to an operating system.

When the robot learns to write a simple program like program1, he can use that knowledge to write program2. In program2 you have to know how to output string text to the monitor so that users can see the number conversions. Program2’s knowledge can be used in program3 because program2 has the ability to manage classes and calling functions from the main body. Finally, knowledge from program3 can be used to write codes in program4.

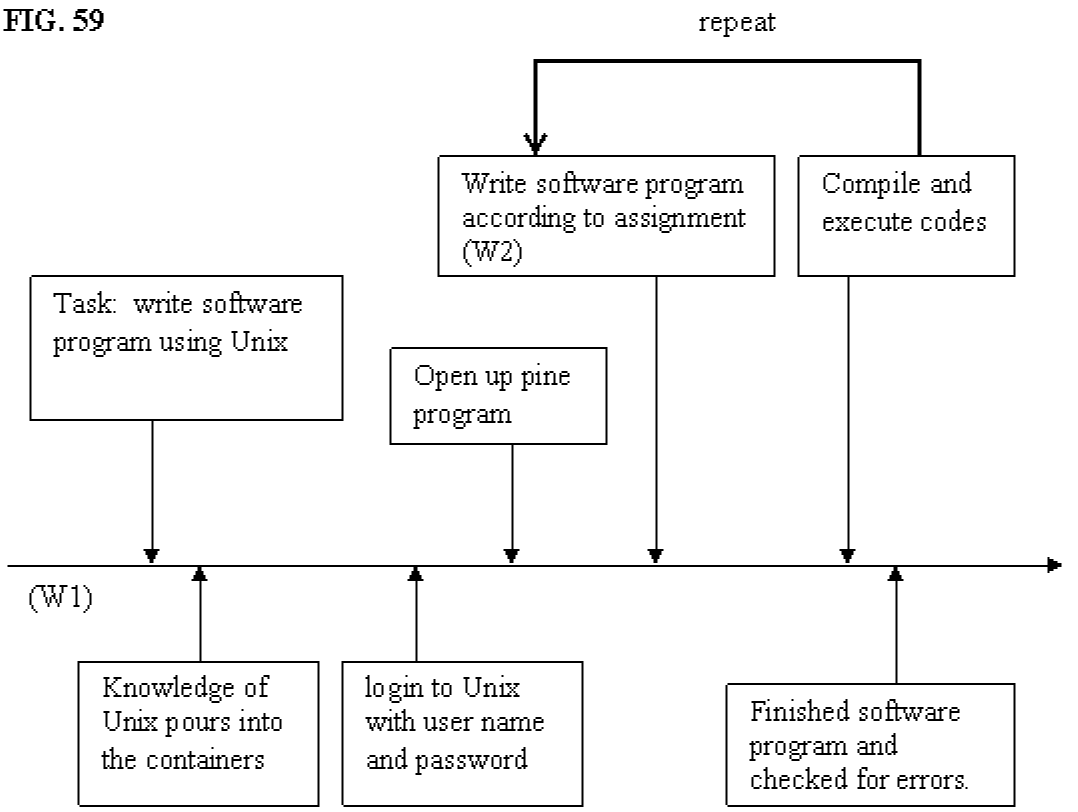
Thus, by learning and writing software programs from simple to complex, the robot can form smarter intelligent pathways to cater to complex tasks.

Hierarchical intelligent pathways are also considered when learning how to write software programs. Intelligent pathway (S1), which is a universal problem-solving pathway, can be used to write software. If the robot fails in completing the assignment, he will loop itself to try again. He will modify his goals, change his strategies and take another action. This loop will repeat itself, until the robot completes the assignment, which is to write a specific software program.

To make the task of writing software more complex, imagine that the professor wanted the robot to use Unix to write the codes. The robot’s brain has two tasks to juggle, one task is to operate the Unix system and the second task is to write the software program. Let’s say the robot knows how to write software programs, but doesn’t know how to use the Unix system. The professor will give a primer on the Unix system and the robot has to read the instructions and understand how to operate his account.

FIG. 59 is an intelligent pathway, called W1, to open and use the Unix system. W2 is another intelligent pathway to write a specific software program. The robot’s brain manages the two tasks (W1 and W2) and merges them together to physically allow the robot to write the software program. The robot will first login to the Unix system using his user name and password. Then he will open a writing program called pine. Next he will write the software program according to the assignment. He can test the software program by executing commands in the Unix system. The robot will apply the trial and error method to refine the software program. Finally, after the tests are over the task of writing a software program is done. However, in order to accomplish writing the software program (W2), the robot had to use the Unix system (W1). The robot's conscious managed performing two of the tasks in order to finish writing the software program.

FIG. 59



Learning to make decisions

Making decisions is based on a lifetime of learning from school. Teachers teach the robot how to make decisions. Assignments are given by teachers to instruct the robot to do tasks on decision making. These assignments include: worksheets, homework assignments, and real life examples. The robot will also learn to make decisions based on trial and error. The knowledge learned in school forms the foundation for intelligent pathways to make decisions. The robot will modify these decision making pathways through the process of trial and error.

When the robot is at its early stages of life, it will have to build its pathways from simple data then as it gets older and there are more data in memory it will organize the pathways into complex intelligence. Just like how we humans have to learn to walk, to talk, to move, to eat, these machines have to go through life the same way. Let's illustrate the gradual forming of simple data into intelligent data by outlining a series of stages.

1. innate reflexes
2. trained to do things
3. sequential events
4. sentence commands
5. give robot option commands
6. practice makes perfect
7. copy other peoples behavior

Decision making comes from many factors and the 7 stages of human intelligence form the intelligent pathways to make decisions.

Teachers will teach lessons on decision making, ranging from easy to complex. An easy decision making is selecting an item from a group of items. A complex decision making is answering a question for a physics exam.

Decision making is just one form of a task. The robot has the option of aborting the task or continuing a task or postponing a task. Once a decision has been made that decision can be modified. A human being can decide to go to Mcdonalds for lunch. However, when he gets to Mcdonalds, he might change his mind and go to Pizza hut instead.

Examples

Imagine a teacher wanting the robot to select one movie from 100 different movies. The teacher will help guide the robot by giving him criteria steps (they don't have to be in linear order) that would help narrow down the decision making process. The teacher might say things like:

1. "what kind of movie category do you like to watch?" -- "action and comedy movies"
2. "here are the choices. What actor/actress would you like to see?" – "Arnold".
3. "based on his movies, which one would you like to select?" – "terminator 2".

This is just one example of selecting a movie from a group of different movies. There might be a situation where the robot has to select from 5 movies or 2 movies. The robot can use the example above to select from an arbitrary number of movies. This example will self-organize with other similar examples and a universal intelligent pathway will be created in memory to make decisions related to movie selections.

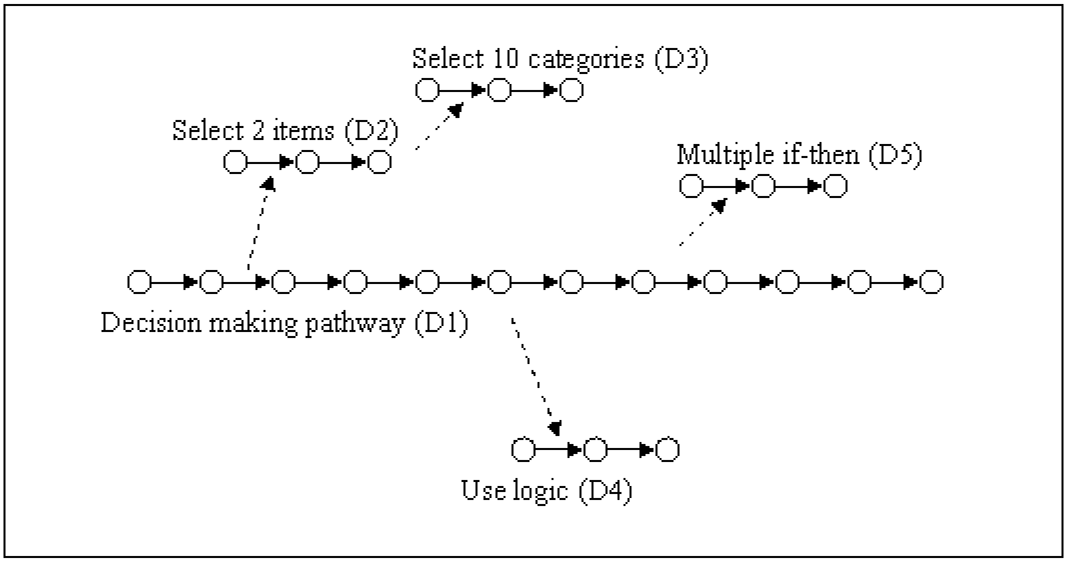
What about buying items in a supermarket? The decision making process will include similar steps to the example above. In other words, it inherits some instructions from the universal intelligent pathway to select movies. However, buying items is a different situation and requires different instructions. Teachers will teach the robot how to buy items in the supermarket in a rational way. The teacher will make up assignment worksheets that will ask the robot to select items in a supermarket. In the worksheet, there might be two items. One item is very expensive and the other item is very cheap and both items look and taste the same. The worksheet will ask which item should the robot select. The correct answer is the cheaper item. If both items are the same but their prices are different, then it's obvious that the robot should buy the cheaper item. Let's call this intelligent pathway (D2).

The last example is simple. To make the decision making process more complex, new facts should be included such as: the cheaper item is a generic brand and is known to have product complaints, but the expensive item is a name brand and has a good reputation.

Another factor is how much money does the robot currently have? If the robot decides to select the expensive item, but doesn't have the money, then by default, he has to select the cheaper item.

Complex decision making builds on itself through a bootstrapping process. In a complex decision making process, the robot has to use many encapsulated linear decision making pathways. FIG. 60 is an illustration. Intelligent pathway D2 is to select 1 out of 2 items. D2 is encapsulated in D1. Next, D2 might trigger a new decision pathway such as generating 10 categories based on D2's decision and selecting one out of the 10 (D3). Next, to narrow down the choices, a D4 is used to come up with new facts about D3's decision. Finally, with all the selections made from D2-D4, the robot will use multiple if-then statements to come up with one decision (D5). Intelligent pathways D2-D5 are all previously learned in the past. Through steps (via sentences and visual examples) by teachers, the robot is able to create intelligent pathway D1 to make a very complex decision.

FIG. 60



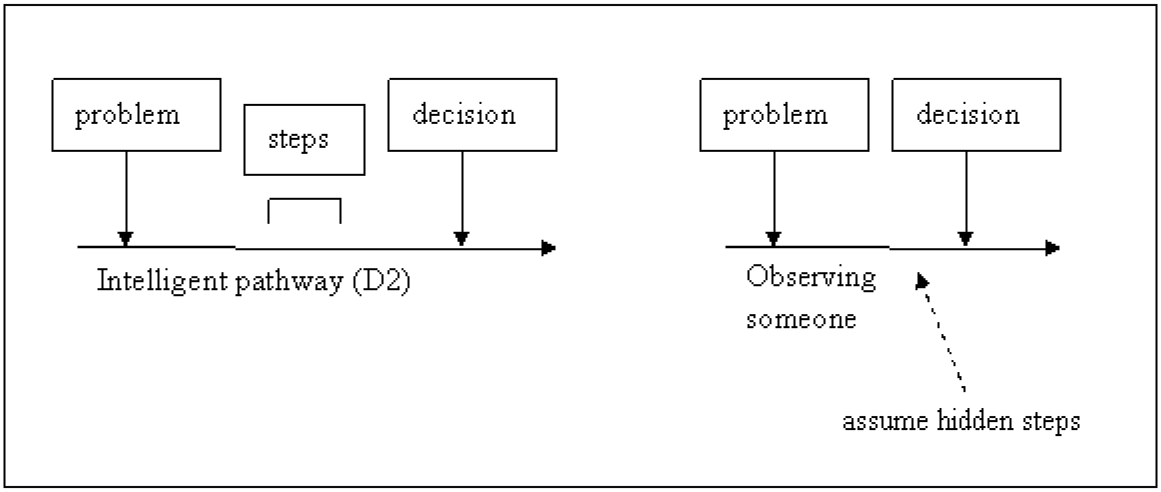
Decision making can also be learned and copied

The lessons from teachers should have relations with observation of decision making. This way, the robot can form intelligent pathways to make decisions based on observation and not by teachers teaching step-by-step examples.

FIG. 61 is a diagram depicting D2. The pathway shows the steps of the robot that will lead to a decision. To the right of pathway D2, is the robot observing someone making a decision similar to D2. Instead of seeing the hidden steps of making the decision, the robot only sees the problem and the decision. In D2, there is: the problem, the steps, and the decision. In the observation pathway the robot only see the problem and the decision.

Through relational links between D2 and the observation, the robot will assume that the person is using the missing steps to make the decision.

FIG. 61



With this said, the decision in D2 can also change if the decision making in the observation is different. Based on many examples, the intelligent pathway D2 will be modified based on the observation of decision making. The robot will try to make decisions based on what the average of society will do. The teachings establish the foundation for decision making. However, the observation of decision making and trial and error will decide how intelligent pathway D2 will evolve.

Many factors are needed for decision making. The intelligent pathways store the probability of selection. This probability could be a learned thing or it could be a hidden object that is not known to the robot. A learned probability is if the robot learned to use a calculator to determine the probability of something. The robot might be taught to calculate the probability of something in its mind. What are the chances an event will happen? When making a decision based on flipping a coin, the robot knows that there is a 50 percent chance, it could be heads or tails. This form of probability is learned. The math results of the probability will determine what the robot will select.

In some cases, the robot can do calculations in the mind and to use that info. to select items. In other cases, the robot has to work out the math problem on paper and to use the answer for the decision making process. Yet, in other cases, the robot is not aware of the probability and the probability is based on hidden data generated by the patterns stored in intelligent pathways.

Complex if-then statements

The intelligent pathways to make decisions are usually based on complex if-then statements (or common discrete math functions). Many similar pathways are learned in memory and these similar pathways self-organize to form patterns. These patterns establish computer programs in intelligent pathways to activate the linear steps to generate a decision. Intelligent pathway D1 is one example. D1 can have many variations and if the robot follows D1 in multiple same situations, the linear steps will be slightly different.

The reason why is because the intelligent pathways store a computer program that will select different steps, at given times, to accomplish a goal. The goal in this case is to make a decision or to select items. Following intelligent pathway D1 will not always generate the same steps to make a decision. However, the computer program in D1 will most likely come out with the correct decision at the end.

Generating complex if-then statements are done through patterns. The robot will use intelligent pathways to come up with meaningful logic to accomplish a task. The intelligent pathways are universal and it will generate complex if-then statements (or any combination of discrete math functions) to generate a decision. Universal means that the intelligent pathway was trained with many similar examples of solving a problem and it can solve the problem under various circumstances.

Types of decision making

There are many different types of decision making. Random selection is one type or logical decision making is another type. A situation where the robot has to select from a variety of food is one example of logical decision making. The robot has to decide on 3 food: hamburger, spaghetti or hot dog. He understands he can only choose one food to eat. The robot will look into each items' powerpoints in memory. The powerpoints is the desirability of that item. A function in selected pathways in memory will determine which item's powerpoints are highest and output that information to the robot's conscious. The robot will select the item with the most powerpoints. For example, if the hamburger has 10 powerpoints, the spaghetti has 18 powerpoints and the hot dog has 14 powerpoints, then the robot will select the spaghetti.

Things can be more complex. The robot loves to eat spaghetti and the powerpoints are high, but what if the serving of the item is small? The factor about size is confronted next. Although the robot selected the spaghetti, the size of the hot dog might over power the decision to pick the spaghetti.

What if the hot dog had onions? The robot is allergic to onions and can't eat any food that has onions. The robot will automatically turn down the hot dog and activate a new problem: select between the hamburger and the spaghetti. The hamburger is larger in size than the spaghetti, but the robot loves spaghetti more than the hamburger. Either the robot will randomly pick or it will come to a compromise (via friendly conversation). At the end, the robot picked the spaghetti.

You can see from the linear decision making problem that it takes the robot time to resolve the many conflicts in the decision making process. However, the more the robot learns how to make decisions, the better it will be at the task. Also, the process of trial and error will evolve the intelligent pathways learned by teachers. The steps to making a decision should be based on using logic to narrow down the possible choices -- breaking up the complexity into chunks and tackling each chunk separately.

Other decision making, like random selection, is based on hidden data in pathways that randomly pick an item. The robot learns to pick randomly by observing decision making situations. For example, if the robot watches a game show and he sees players select the 2nd door more than the other doors, then when the robot is confronted with the decision of selecting random doors, the robot will choose the 2nd door.

I was asked by a friend once this question: “pick a number between 1 and 10”. My answer was 7. My friend told me that 80 percent of all people answered 7. The reason why is because he told me to pick a number between 1 and 10. My obvious thought was 5. I remember he said it could be any number, so I didn’t pick 5, instead I picked a number between 5 and 10 which is 7.

This type of example shows how predictable human beings are. Even though they learned knowledge differently and live different life styles, society made them into the average joe.

Other types of decision making include selecting pictures and movie clips from webpages. When people go online they make decisions in terms of what they are searching for online. The search engines aren’t really good at finding exact things online. They give the user an approximate search, but all the hard work is done by the user. When I was writing my patent applications it took me hours just to find 30 images that I needed for my drawings. All 30 images online needed some form of modification, such as resizing or adjusting the contrast, in order to be used in my patent application. The Patent office has rules that the drawings must follow and I had to make sure the images follow these laws.

If the search engine was intelligent at a human level then all I needed was one search and my 30 drawings would be found. In fact, the search engine would help me modify the 30 drawings and download them into my computer.

That’s not the case with modern day search engines. More complex type of decision making is to compare images or compare movie clips and select from a group of items. There can be a mixing of different items. The robot might have to select from 3 websites, 2 movies, and 3 pictures. Only one item can be selected for purchase.

Comparing a movie to a website is difficult because each item is completely different in many aspects. It’s because of years and years of learning that the robot is able to compare different media types. Teachers have taught the robot many different examples and he has personally engaged in decision making situations in his life. This knowledge and experience of comparing different media types is self-generated by the robot actively making decisions. In order to actively make a decision, teachers have to teach the robot how to compare and choose.

One such method taught by teachers to compare different media types is to identify aspects of each item and compare these aspects. The comparison might reveal facts from each item that might benefit the robot. If the robot was comparing a website to a movie, maybe content of the two items can be compared instead of presentation of data. Maybe the financial worth of the website can be compared to the financial worth of a movie. Let’s say that a rich person had a choice to make between buying a website or owning the rights to a movie, he will have to compare the two items. He must compare different aspects of each item and ultimately reveal how each item can benefit the rich person. After debating with itself, the rich person will weigh the benefits and select one of the two items.

Chapter 12

6. Playing videogames

Observation of pain and pleasure

When the robot plays a videogame, the robot's conscious searches for the objectives and rules of the game. The videogame was made by a producer and the producer defines the objectives and rules of the game. One of the jobs of the producer is to guide the player to play the game in a certain way. He/she has to make sure that the game runs smoothly for the player and the player is having fun.

The robot has to use intelligent pathways in memory to identify what the objectives of a game are. In videogames like baseball and football, the objectives and rules are already known to the robot. Videogames the robot never played before will require the robot to search for the objectives and rules of the game using trial and error and logic.

For example, let's use the popular game joust. If the robot never played joust before and he is playing the game for the first time, how will he know what the objectives and rules of the game are? He will know by playing the game. The robot will press buttons on the controller to identify the controls of the game. In joust, there is only 1 button and a joystick. The robot presses the button and found out that the button makes the character in the game fly; and the joystick moves the character around in a 2-d environment.

Once the robot understands the controls, it will store this information in the rules container in terms of what actions are possible and how to control these actions.

Next, the robot will use identification, in terms of pain and pleasure, to understand the objectives and rules of the game. Pain and pleasure will be the factors to determine what the producers want the player to do and what the producers don't want the player to do. The robot will make the character in the game take action. The robot can make the character fly, collide with enemies, bounce into walls, or land into the lava pit. Intelligent pathways will choose these actions in a heuristic way.

The robot will find out that when the enemies touch the character (in the game) on the head, the character will die. This event will trigger pain because the character lost a life. When the character lands on a bird's head, the bird turns into an egg. The robot is also aware of linear sequences of events. After the bird turns into an egg, the robot has 5 seconds before the egg turns back into a bird. The robot also found out, through observation, that if the character touches an egg, the egg disappears.

Based on logic, the robot will find out that it is a good thing to hit a bird on the head and collect the egg, instead of hitting a bird on the head and not collecting the egg. By collecting the egg, the character is eliminating an enemy. Through further observation, the robot will find out that if all enemies are eliminated, then the character passes that level. The robot also found out that the more levels it passes the higher its points become.

These trial and error gameplays will determine the objectives and rules of the game joust. No prior knowledge was ever given to the robot about the game. Only through trial and error and a form of logic, did the robot find out what the objectives and rules are of the game.

Games like Donkey Kong require the robot to use logic to find out the objectives and rules. Each level has its own objectives and rules. When I first played the game, I didn't know what to do. All I knew is that the controls allow a character in the game to move in a 2-d environment and jump over things. During my gameplays, I found out that if the character touches a barrel, he will lose a life. I also learned that if I jumped over the barrels I will not lose a life. Losing a life is pain and jumping over a barrel means living, so I understood, one of the rules: "jump over barrels the donkey is throwing at you".

From close observation, I see a princess at the top of the building. Through logic, I concluded that the objective is to reach the princess at the top. The only way to do that was to go up the stairs and dodge barrels. When I got to the top of the building and rescued the princess, the videogame said that I past the level. This passing of the level gives me pleasure and it's telling me that I had reached one of the objectives of the game.

In the next level, I encountered a different type of environment. Donkey Kong was at the top of the building and there were ladders placed to allow the character to climb the building. At first, I thought that I had to climb up the ladders and reach Donkey Kong. To my surprise, I found out that I lost a life. This event caused me pain and I found out that climbing up the ladder to the top of the building isn't the objective of this level. Next, I observed that there were pins located at the corners of each level of the building. I was curious as to what these pins were. When I touched these pins they disappeared. I had to also watch out for enemies. At this point, I logically decided to collect all the pins and to avoid the enemies. When all pins were collected the building collapsed and Donkey Kong landed on the ground. The videogame said I past the level. This event caused pleasure and it also gave me the objective of the level, which was to collect all the pins and avoid touching any enemies.

Other games like Pac-man are very difficult to find the objectives and rules. Someone has to tell the player what to do. A simple sentence such as: "your objective is to collect all the dots on the screen", will be enough to understand the objectives. Some games for more modern videogame systems are much harder to find their objectives and rules.

It's not about the short-term observation of pain and pleasure that determines what the producers what the player to do. It's more about the long-term observation of pain and pleasure. In the case of the super Nintendo game, Gradius, when the character receives more powerful weapons, he is less likely to die in the game. However, if the character receives too much powerful weapons, the videogame will introduce an unbeatable enemy that will rob you of all your weapons. Thus, just because the character has more powerful weapons don't mean that it's beneficial to the character.

The robot playing the game has to look at the long-term benefits and not the short-term benefits. Should the character collect 7 out of 10 powerful weapons and keep all 7 weapons or should the character collect 10 out of 10 powerful weapons and lose all 10 weapons?

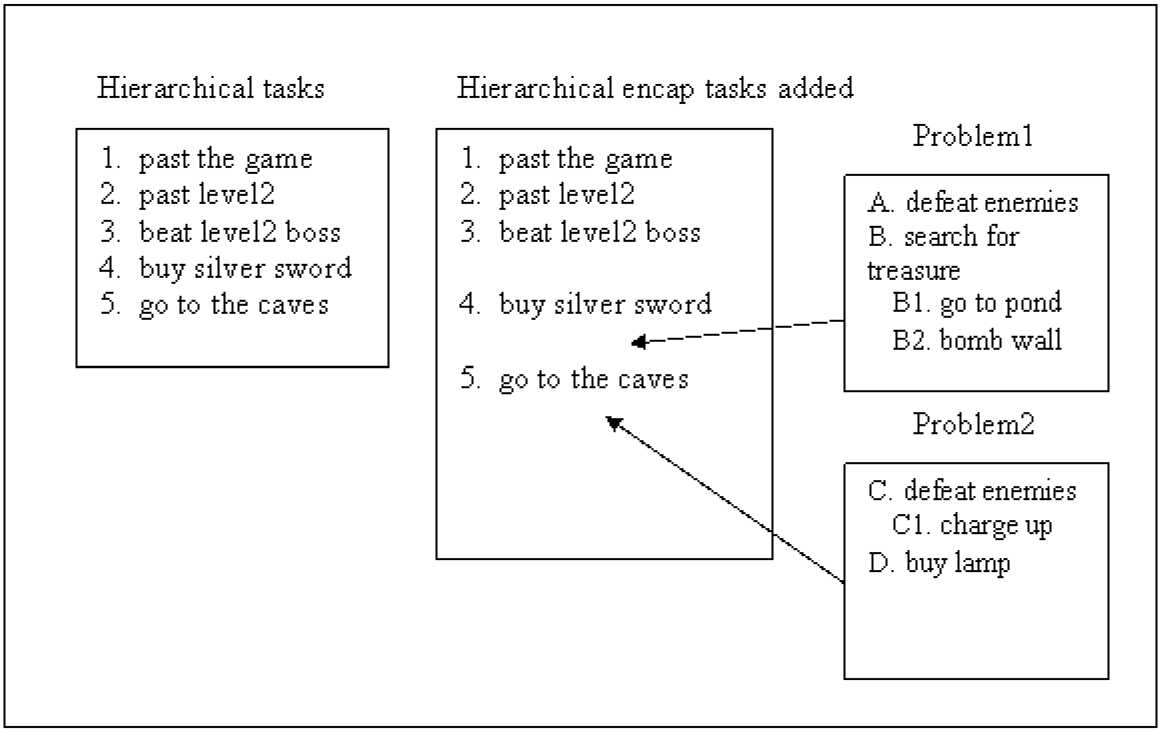
The producers of the videogame define how the player should play the game. If the producers wanted the character in the game to go up a mountain, the screen will automatically move upwards. The character has no choice but to move upwards. If the character doesn't move upwards and touches the bottom of the screen, he will lose a life. If the character moves upward, he will not lose a life. Pain and pleasure decides how the player should play the game.

In another case, the producers don't want the character in the game to go into the swamp. If the character jumps into the swamp his life energy will be lowered. In fact, the longer the character stays in the swamp, the more energy he will lose. The robot (the player) knows that the event of losing life energy is a bad thing. This bad thing will give the robot a rule to follow, which is: "don't go into the swamps".

Playing the legend of Zelda game

The robot has to manage hierarchical tasks when playing Zelda. A problem might arise and additional hierarchical tasks are added. FIG. 62 is a diagram showing how hierarchical tasks are modified. The original task has 5 encapsulated tasks being managed by the robot to play Zelda. The modified tasks add in two other encapsulated tasks. This diagram shows what the task container will look like after playing level2.

FIG. 62



When the robot plays the Zelda game, his goals are to past the game. In order to past the game he has to determine what he has to do right now, which is to past level2. In order to past level2, he has to beat level2's boss and in order to beat level2's boss, he has to buy the silver sword and go to the caves.

This is the original hierarchical tasks the robot must follow. During the playing of the game, the robot will run into problems in the game (problem1 and problem2). The robot must use logic to solve these problems and generate additional tasks and to modify pre-existing tasks. When the robot is trying to buy a silver sword, he will probably encounter enemies. He has generated a new task, which is to defeat the enemies that are blocking his path to the shop. After completing that task, he has encountered a villager who told him that a treasure is located in a pond close-by. At this point, the robot will generate a new task, which is to find the treasure. He will go to the pond and bomb the walls as instructed by the villager. After completing these tasks, the robot will go back to its original task, which is to buy the silver sword (problem1).

After purchasing the silver sword, the robot has to go into the caves. While the robot is traveling to the caves, he will encounter enemies and new tasks are generated: to defeat enemies. When the robot actually gets into the caves, he finds out that the caves are dark and he needs a lamp. He will not be able to see if there is no lamp. This prompts the robot to do another task, which is to buy a lamp from the village and go back into the caves (problem2).

Although the robot had 2 interruptions, he was still able to remember his original tasks. He knows that he must beat the level2 boss; and beat the entire Zelda game. He also remembered what he did in the past and what he has to currently do. The human brain is very primitive and it can only remember a limited amount of tasks. Human beings can forget previously committed tasks.

Tasks in the task container are modified during runtime. After the robot passes level2, he has to pass level3 and after passing level3 he has to pass level4. Logic determines the order of tasks. Role playing games aren't as straight forward as most videogames. The levels aren't given to the robot automatically. The robot has to use logic to figure out where the next level might be located. Sometimes, maps are given and they tell the robot where the next level is, but other times, the character in the game has to talk to villagers and wizards to find out where to go next and what his next mission should be.

When the robot is stuck in the game, he has to use logic to solve the problem. In fact, when the robot is stuck he has to first identify the problem. Once the problem is identified, then he can use logic to solve the problem. The steps to solving the problem aren't fixed. The robot might try an action and that action doesn't solve the problem, so the robot tries another action. This loop will repeat itself until the robot is successful in solving the problem.

Sometimes, when trying to solve a problem another problem pops up. Thus, one task can have infinite amount of recursive sub-tasks. It is a fact that the human brain can only remember and manage a limited amount of recursive tasks.

The key to managing recursive tasks is to remember what the robot was originally doing before. The robot is doing task1 and he runs into problems, so he adds in 2 new tasks. After doing the 2 tasks, he remembers that he was doing task1 and continues where he left task1. During the continuation of task1, the robot runs into another problem. This time he adds one addition task, called task2. While the robot is doing task2 he runs into yet another problem and the solving of the problem requires 2 more tasks. After the 2 tasks are done, the robot remembers that he was doing task2 and he also remembers that he was doing task1. After completing task2, the robot remembers he was doing task1 and continues where he left task1.

This is the linear thinking of how the robot does multiple layered tasks simultaneously. If the recursive tasks go too deep he will forget what his original tasks were or he might skip a previously committed task.

The more the robot plays a game the easier the game will be

When the robot first plays a scene in Zelda, the character might die a few times. However, the more he plays the same scene the easier the scene becomes; and the more skilled the robot will be in playing the overall game. The reason why is because, the robot's brain already stores trial and error pathways related to the scene; and he is aware of any obstacles that are in his path. He also knows the correct knowledge needed to get itself from one location to the next. The rules of certain situations are automatically activated. The tasks needed for a given situation are automatically stored in the task container.

In the first gameplay of a scene, the robot's conscious is using a general type of computer program to past the level. As the robot plays the scene 5-7 times, the robot's conscious is using a specific and optimal type of computer program to past the level. The knowledge gathered, the tasks to be done, the rules to follow are all managed by the robot's conscious. The repeated playing of the scene will optimize how the computer programs manage knowledge, rules and tasks. The trial and error helps the conscious to come up with rules to follow or new methods to use for that specific scene. New ideas on solving problems might activate because of trial and error. These ideas can be used to solve a problem in the scene.

Using logic to identify tasks

In the game of Zelda, the robot has to think and use logic to determine its future actions. A strategy is needed before you can go into a castle (or cave). The robot must know what its goals are before entering a castle. A castle is just a big place that has many rooms arranged in a maze like setting.

Based on my personal experiences, through trial and error, I found many strategies that are useful to past the Zelda game. I found out that the goals of entering a castle is to: 1. find the boss and defeat him. 2. search for the vital item. These two tasks are a must when it comes to passing Zelda. The producers of Zelda made the game so that in every castle or cave, there is a boss and there is a hidden vital item. If the robot doesn't find the vital item, he will not be able to beat the next level. So, even though the robot defeated the boss in that castle, he still needs to find the vital item in order to successfully pass the level.

I personally learned this strategy the hard way. I was playing Zelda in the past, and in level5 I didn't get the vital item. I found level6's castle, but I couldn't get in because there was a boulder in front of the castle. I was stuck and didn't know what to do. I found out, through trial and error, that the vital item in level5 is a glove. The glove is used to break boulders. From that moment on, I reminded myself that before I go into a castle my tasks are to defeat the boss and to find the vital item.

This is just one strategy that I found that would help me in beating the game. Throughout the gameplays, I found many strategies. These strategies are tagged to the objects involved. Object association will activate these strategies when the time comes. For example, when I enter a castle or cave, I will activate the strategy and the two tasks, beat the boss and find the vital item, are stored in the task container automatically.

Other methods can be used to play the game. The robot has to be aware of the maze and where the rooms are located. This temporary map of the castle will prevent the robot from doing repeated travels.

Also, the map in the robot's memory has to remember what each room looked like. There might be a locked door in one room. The robot will remember this room and will search for other rooms to find the key. When the robot finds the key, he can go back to the room with the locked door. The robot has to have a detailed map of the rooms and where the room with the locked door is and where the robot is currently located. This map is important because the robot has to plot out a route to get from its current location to the room with the locked door.

In some complex settings, there might be 4 locked doors and the robot has to find the keys and open all 4 locked doors arbitrarily. In order to do this, the robot has to remember where the locked doors are. He also must remember where the robot has traveled so that he doesn't visit repeated rooms or go around in circles.

In another game called super metroid, the maze is even more complex. Thankfully, the game provides a map of the environment as well as what areas the robot has visited. In the game of super metroid, it's all about getting through unlocked doors. One door might be locked and a missile is needed to open the door. In order to get the missile the robot has to defeat a boss. Other locked doors require a bomb. In order to get a bomb the robot has to defeat another boss. Other locked doors are locked and will not open, unless the robot completes a mission.

The point I'm trying to make is that the robot has to be aware of the entire map and how to get from one location to the next. He has to remember which areas have locked doors and which have opened doors. The complexity of the game is managed by the provided map in the game. However, the robot still has to identify where the hidden doors are in the complex map.

In some cases, in order to open one door, a series of encapsulated doors must be opened first. The strategy in super metroid is to remember where the locked doors are and how to reach a locked door. The ability to remember where the locked doors are and how to reach them is a vital skill in the game. For example, the red door requires a missile to open. All doors in the game with the color red will not be able to open without a missile.

The robot travels in the game. The blue doors can be opened, but the red doors require a missile and the yellow doors require a bomb. He opens the blue doors, but skips the red doors. The robot finds the missile and therefore he can open any red door. At this point, the robot has to remember where all the red doors are in the map. He might look at the map, use logic, and say there is one red door here and one red door here and a possible red door here. Then, he will locate the current position of the character and plot out a route from the current location to one of the locked red doors.

Unlocking all or most of the red doors is important because there might be items in these locked rooms that can be used to unlock other doors. For example, a bomb might be located in one of the red doors. The bomb is needed to open the yellow doors. Opening the yellow doors might lead to a boss. By defeating the boss, the unlocked doors with criterias might be unlocked as well.

Chapter 13

Other topics (part 1):

Learning knowledge from teachers in the real world

When the robot is doing a math problem like addition, his mind has an invisible guide that tells him how to solve the problem – where to focus his eyes on, what knowledge to extract from memory, and how to solve the problem step-by-step. It's like an invisible teacher is there pointing out things to the robot and tells him what to do next.

Solving any problem in life requires an intelligent guide. It doesn't matter if the robot is drawing a picture or writing an essay or giving a speech. The invisible guide is there giving the robot the knowledge to solve problems. This invisible guide is the robot's conscious.

Doing worksheets on paper requires this guide to tell the robot important information. The first thing he needs to do is identify the instructions. Next, he has to follow the instructions. If he has a question, he will ask the teacher. Once, the robot reads the worksheet instructions, the conscious is actually translating the instructions into specific tasks. Tasks are put into the task container and rules are put into the rules container. Knowledge of that specific worksheet will pour into the robot's conscious. The robot will do tasks in linear order until the entire worksheet is completed.

When the robot is doing an addition problem, the robot's mind shows an invisible finger pointing at numbers on the paper. This invisible finger is all in the robot's mind and isn't anything physical. Sentences will activate that tell him how to modify the numbers, how to add numbers, how to erase numbers, etc. The robot might be instructed to use scratch paper to do secondary equations and use the answer for his original equation.

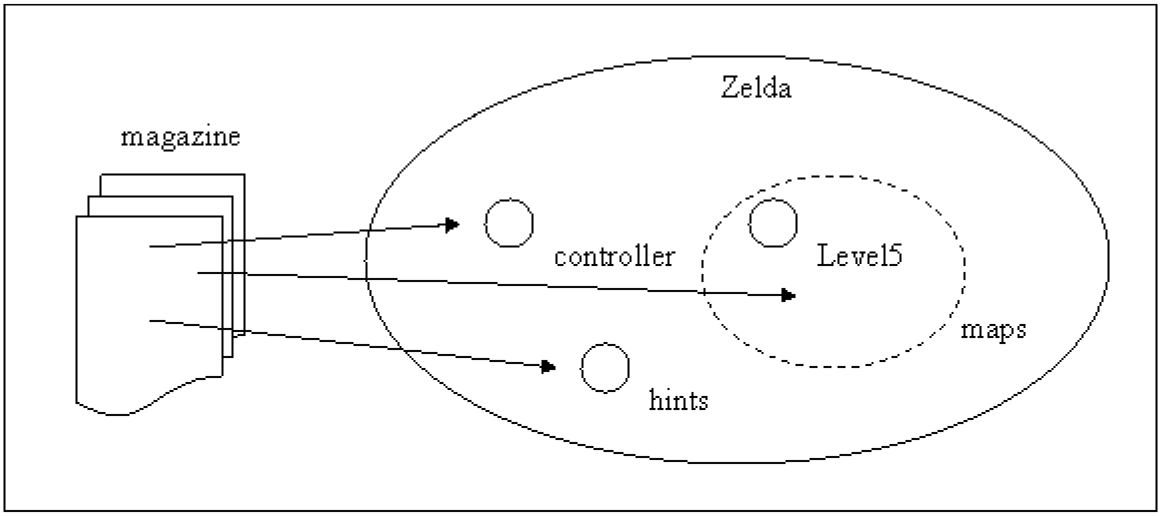
In a worksheet, the instructions might be to cut out images. The robot's conscious has an outliner that delineate where the boundaries of the image are. This outliner can be a finger that is pointing at the outer thick lines of an image or it can be a computer generated line that appears at the outside lines of an image (it depends on how the robot learned to delineate images). The robot might be watching TV and the teacher is using computer generated lines to show the students where the boundaries are. On the other hand, a teacher can point a finger, linearly, where the boundaries are.

The collective knowledge of doing worksheets is a vital ability to accomplish complex tasks. Doing any worksheet require an intelligent guide that help the robot to focus on specific objects, point to things he should be aware of, give facts about objects, manipulate data, and identify things in the worksheet.

Reading videogame magazines and applying the knowledge to games

Knowledge read in any book or magazine is stored in memory in an associational way. If I am reading a magazine about videogames, I ‘m storing all that knowledge in my videogame area in my brain. Specific games in the magazine will be stored in specific areas in the videogame area. Let’s say that I was reading a magazine, and in it, is a section on Zelda. I will read the knowledge and these knowledge will be stored in the Zelda part of my brain. There might be facts about the controller and my brain stores that fact in the controller area. There might be facts about level5 and my brain will store that fact in level5. Finally, there might be hints given about the game that will give a general strategy for the reader (me). My brain will store all the hints in the Zelda area. If you look at the interconnected structure of the network, you will see that data read on Zelda is stored in a general area (FIG. 63).

FIG. 63



Because the data is stored in memory, as well as, stored in the long-term memory, the data is very strong. The next time I play Zelda, the facts from the magazine will be strong data and it will most likely be activated in my mind. Association is a powerful tool used to recall data. If I was looking at my controls, the data read in the magazine about the controls will activate. If I was in level5, the data on level5 in the magazine will be activated in my mind. If I get stuck in the game, the general hints from the magazine will activate in my mind.

These facts learned from the magazine were activated because they had association with objects in Zelda. If I was playing a different game like contra, none of the facts from the magazine about Zelda will activate. The reason why is because my brain stored the data from the magazine in the Zelda area. The word Zelda stores all the facts. The content in the facts store them in specific areas.

One of the most powerful facts is the if-then statement. If the robot recognizes a condition, then it might take action. The if-then statement is powerful because the robot is always identifying things.

If the robot recognizes a dog, then activate its stereotypes.

The above sentence defines the rules program. When the robot recognizes the target object it will activate its strongest element objects. The robot can be reading a videogame magazine, and during his read, he might translate knowledge read into an if-then statement. “If you recognize the wizard in level5, give him 5 rupees and he will tell you where to find the lamp”. The next time the robot recognizes important objects in the environment such as wizard or level5 or lamp, the if-then statement will activate. This will give the robot a new task, which is to go to level5 and find the wizard and do the instructions in the if-then statement.

Association will also bring far away data in memory closer together. Migration of data from one part of memory can be done using association. The process is accomplished by activating conscious thoughts, storing these thoughts in the current pathway, and letting the self-organization do its part.

The robot talking to itself

When the robot learns how to have a conversation with people, the intelligent pathways can also form self-awared behavior. The robot can talk to someone and that someone is itself. Using intelligent pathways to have a conversation with someone is also used to talk to itself. In fact, all intelligent pathways can be used to talk to itself. When the robot is bored, he might “daydream” by planning what he will do 1 week from today or plan what kind of food he wants to eat for lunch.

He might even talk to itself any moment that he isn’t engaged in the real world. During analyzing and observation, the robot is talking to itself about what he is seeing. During a conversation, the robot might be telling itself how the conversation is going or tell itself the good experiences his having in the party.

I call this imaginary friend, the robot’s conscious. This friend is the robot, and he knows that secrets can’t be hidden, or any private knowledge is revealed. The robot will be totally honest with this imaginary friend. However, there are some exceptions. The imaginary friend might be in denial or the friend believes a false fact.

Teachers are partly blamed for creating this imaginary friend. The teachers give lessons to the robot such as: “when you are bored, think of something”, “critique about a person”, “close your eyes and come up with imaginative thoughts”. The voice in your head can also be learned through trial and error.

Watching TV might create this imaginary friend. When the robot watches TV, a character might be talking to itself. The robot will copy this behavior. If the robot was watching a comedy show and the character is critiquing about how a person dresses, this knowledge can be copied by the robot. He will go to a party and talk to itself about how a person dresses.

The fabricated movie defines the relationships between objects

Teachers will teach the robot the relationships between objects. The teacher will draw diagrams, in terms of primitive lines, to show relationships. For example, if the teacher wanted to show a car driving from destination to destination, he/she can use arrows to represent the driving. If the teacher wanted to show a parent child relationship, he/she can use a hierarchical tree, with nodes representing a parent or child and lines representing a relationship. The teacher can use flow animation to show the actions of an object. For example, the first frame can be a person1 kicking person2 and the second frame can be person2 on the ground and person1 standing. These two sequential frames indicate an action. The action is person1 kicking person2 to the ground.

These diagrams are very important because they represent the meaning to sentences. When this sentence is read: “Sam kicked Danny on the stomach”, the robot will activate the meaning to each word. The meaning might be the two sequential frames mentioned above. The two sequential frames might be universal and can be modified. For example, the learned lesson might be person1 kicking person2 on the arm. The universal 2 frames can be modified so that an image of Sam replaces person1 and an image of Danny replaces person2. Instead of Sam kicking Danny on the arm, the 2 frames will be replaced with Sam kicking Danny in the stomach. The robot fabricated a 2 frame sequence using other intelligent pathways in memory.

If this sentence is read: “Danny is driving to the store”, the diagram of a map with Danny and an arrow drawn to the store, will be activated in the robot’s mind. If this sentence is read: “the kid has stubborn parents”, the robot might activate a hierarchical tree diagram with facts related to the sentence. Facts like the word stubborn on the parent node and the kid node is represented by a picture of an average kid.

The meaning to sentences is based on what the robot learned from teachers. No language parsers are used or no semantic networks are used to represent language. The associated things the robot learns from the environment creates the meaning to language.

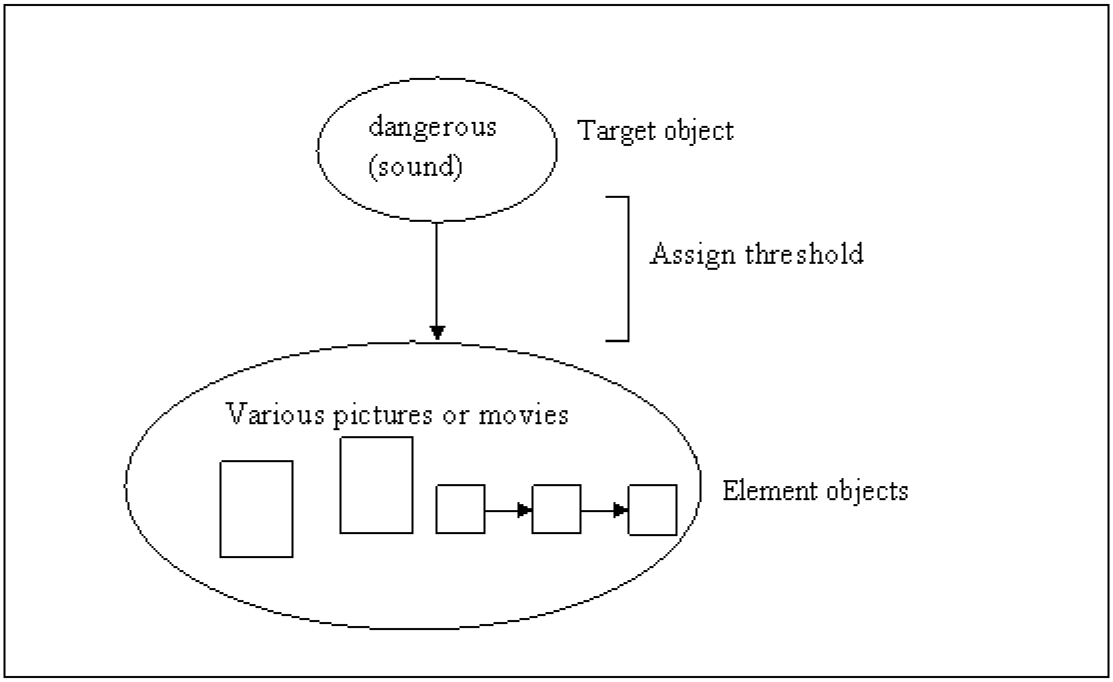
Referencing words to other words

FIG. 64 is a diagram depicting the meaning to the word dangerous. The word dangerous is represented by images of destruction. The images are various pictures or movie sequences depicting nuclear bombs, guns, highways, swords and viruses.

The word is a sound "dangerous" and is the target object. The meanings, represented by pictures or movie sequences, are the element objects.

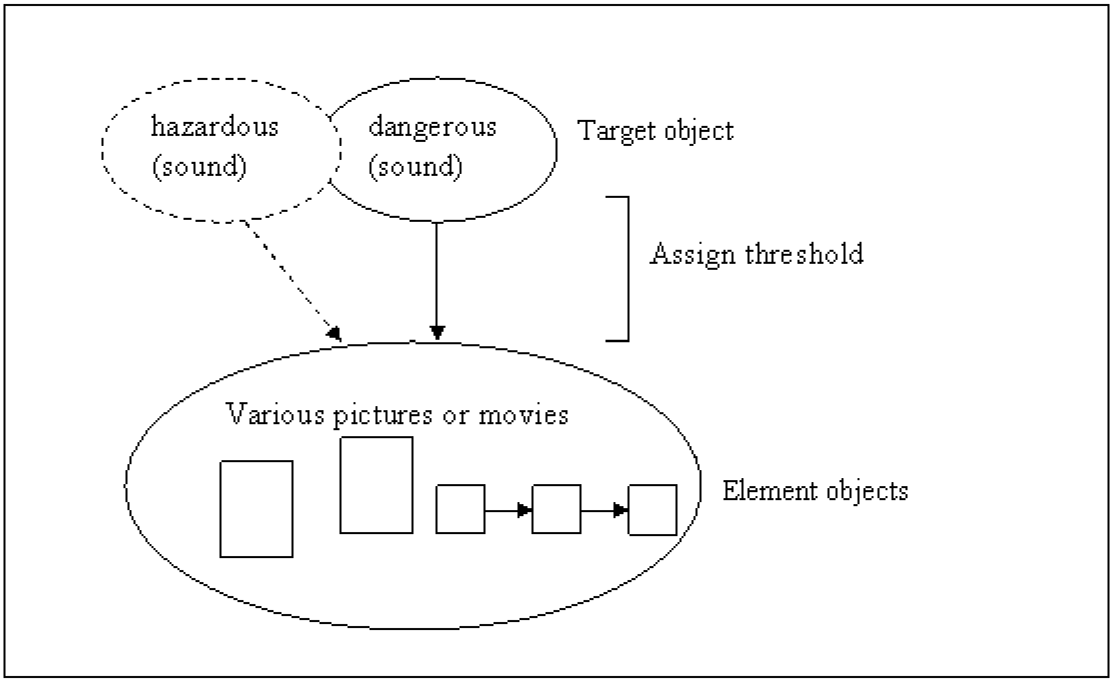
If a sentence is read to the robot such as: "the word hazardous is another word for dangerous", the robot will store this sentence in memory in a meaningful way. The sentence is commanding the robot to store data in an intelligent way.

FIG. 64



The pattern in the sentence is trying to instruct the robot to store the word: hazardous near the word: dangerous. The closeness of the word hazardous to the meaning is the end result (FIG. 65).

FIG. 65



Even more remarkable is if another word points to hazardous. For example, this sentence: "the word harmful means hazardous". Now, we have three words pointing to the same meaning. Of course, each word will have a specific type of meaning. However, all three words share common traits. The self-organization of data in memory will determine this.

Learning to activate correct stereotypes based on target objects

When a target object is recognized in memory all element objects will compete with one another to activate in the robot's mind. The robot can learn from teachers what to activate based on recognized target objects.

I described how this method works in previous chapters. I just wanted to note that the search patterns are based on the learned lessons in school and the configuration of data in the robot's brain. For example, the teacher said: the stereotype for H1 is H2, H3 and H4. With this sentence, the robot will find patterns between this sentence and the data configurations in memory. It will find out that H1 is the target object and H2-H4 are the element objects and that the sentence is a command to configure the data of the target object and the element objects.

This way, the data in memory are configured based on lessons in school. We can train the brain to activate correct stereotypes based on recognized target objects in the future. The stereotypes of a person are different from the stereotypes of an event. The facts from an event are different from the facts of a science book. The searching for data will create optimal search results for different target objects. This behavior is learned through sentences read in books or by lessons from teachers.

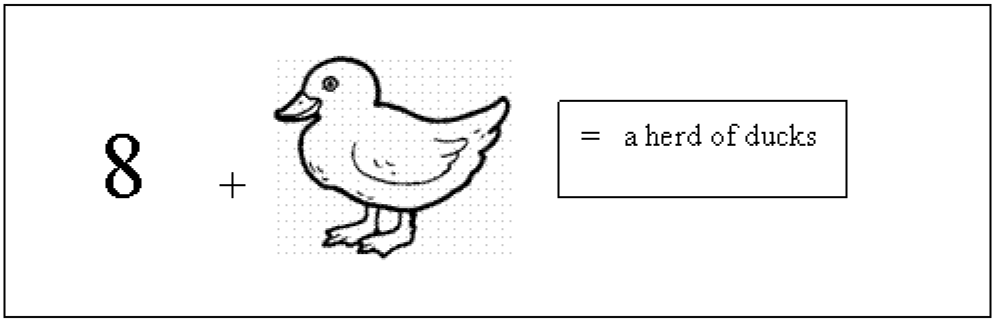
It depends on what the current situation is and the goals of the robot. If the robot was reading a book, he wouldn't want grammar rules to activate. When the robot is reading a book, he just wants to activate a fabricated movie on the story. On the other hand, if the robot was correcting papers, then grammar rules and lessons should activate. It depends on many factors, how the rules program will activate element objects based on recognized target objects. These factors include: the goals of the robot, the current situation, the mood of the robot and what happened in the past.

Using discrete math to represent objects, events and actions

Meaning of something can be done by using diagrams or animation. A picture can represent complex things like the universe or an entire city or an ambiguous word such as war.

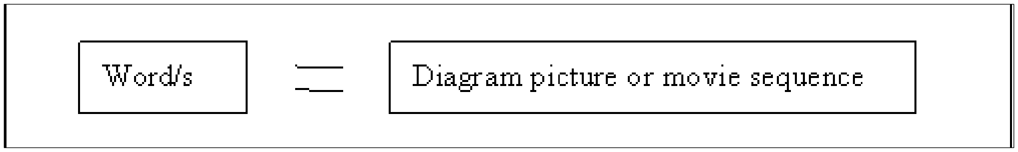
FIG. 66 is an illustration depicting how definitions of words can be represented by diagrams. In the illustration, there is one picture of a duck and the definition is a herd of ducks. Apparently, in order to be considered a herd, the ducks have to number 8.

FIG. 66



This just goes to show that word/s can be defined by pictures or movie sequences -- an animation of a diagram, whereby pictures and lines are put together to mean something. In other cases, a single picture can represent that word. For example, the word universe can be represented by a picture of stars in the night sky. Or the universe can be represented by an animation of the solar system (FIG. 67).

FIG. 67



If you look at a dictionary, words are listed next to their definitions. When the robot reads the definition of a word, a fabricated movie is activated. This fabricated movie is made up of a combination of 5 sense data (mostly visual images) that represent the meaning of the definition (sentence/s). This meaning is a diagram animation. Thus, innately, each word is represented by a fabricated movie. This fabricated movie can also contain static data like a form or picture.

The event pool

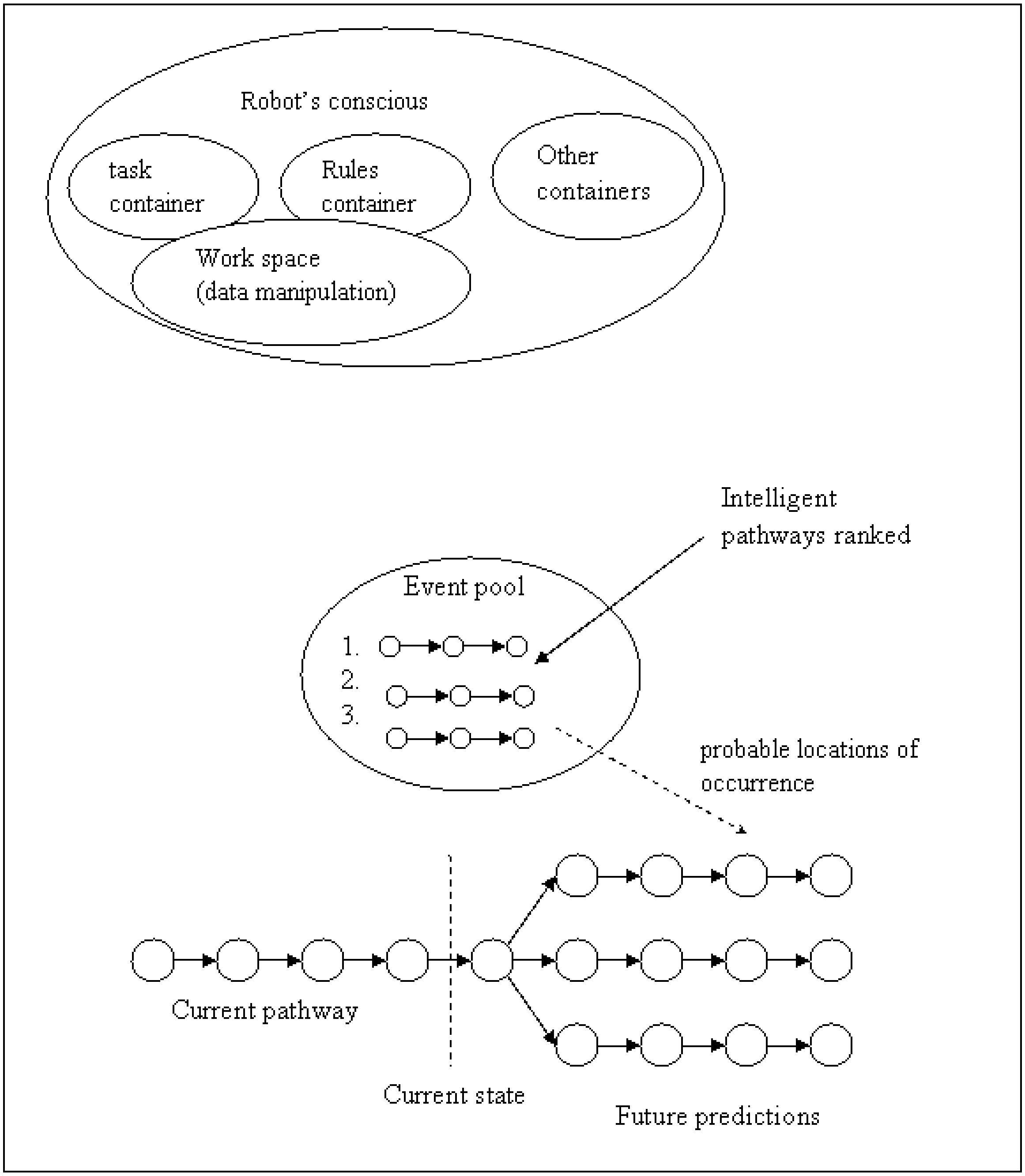
FIG. 68 is a diagram depicting how the event pool works. While the robot's brain predicts the future pathways, an event pool is used to rank what events will most likely happen in the future.

The event pool comprises intelligent pathways ranked in a hierarchical manner. All these intelligent pathways will point to future events in predicted future pathways. Intelligent pathways can be rules, tasks, complex discrete math functions, logical thoughts, searching for data, storing data, modifying data, facts about an object or group of objects, meaning to language and so forth.

The event pool stores the instructions of what will happen in the future and what actions the robot will do in the future. At each iteration of the for-loop, the robot will select an optimal pathway from memory. This pathway basically defines the robot's conscious. It defines the tasks to be done, the rules to follow, the logic to manipulate, the information to extract from memory and so forth.

The optimal pathway selected at each iteration will shape the computer program inside the robot's conscious and it will instruct the robot to: do tasks, solve interruption of tasks, follow rules, provide knowledge about an object or situation, provide meaning to language, fabricate a movie, solve problems, logically think, do complex decision making and so forth.

FIG. 68



Playing street fighter

Imagine that the robot has never played street fighter before and he has no prior knowledge of what the videogame's objectives and rules are. The robot will play and observe the game and identify what is good and bad in the game. Logical pathways that analyze a situation can be used to come up with the objectives of the game. The rules of the game will also be identified when the robot (the player) does something good or bad.

These good and bad behaviors in playing the game keep or delete tasks and rules. The good facts are kept and the bad facts are deleted. When the robot is confronted with the same situation, vital facts (such as rules or tasks) will activate and the robot will follow these facts.

Trial and error is one way of learning the objectives and rules of a game. Another way is from reading magazines and from listening to other people. The robot might be reading a magazine about street fighter. In the article, the author list four hints to the game. The robot will store these vital hints in memory and remember to use them when playing the game. When the robot is playing street fighter, these hints will activate, depending on what their contents are. The robot will try these hints. If hints are tried and they don't work, the robot will delete these hints from memory. If hints are tried and they work, the robot will remember these hints in memory. Thus, the objectives and rules of the game are learned, but only through trial and error are these facts modified. The robot will keep information that are important and forget any information that is not important or wrong.

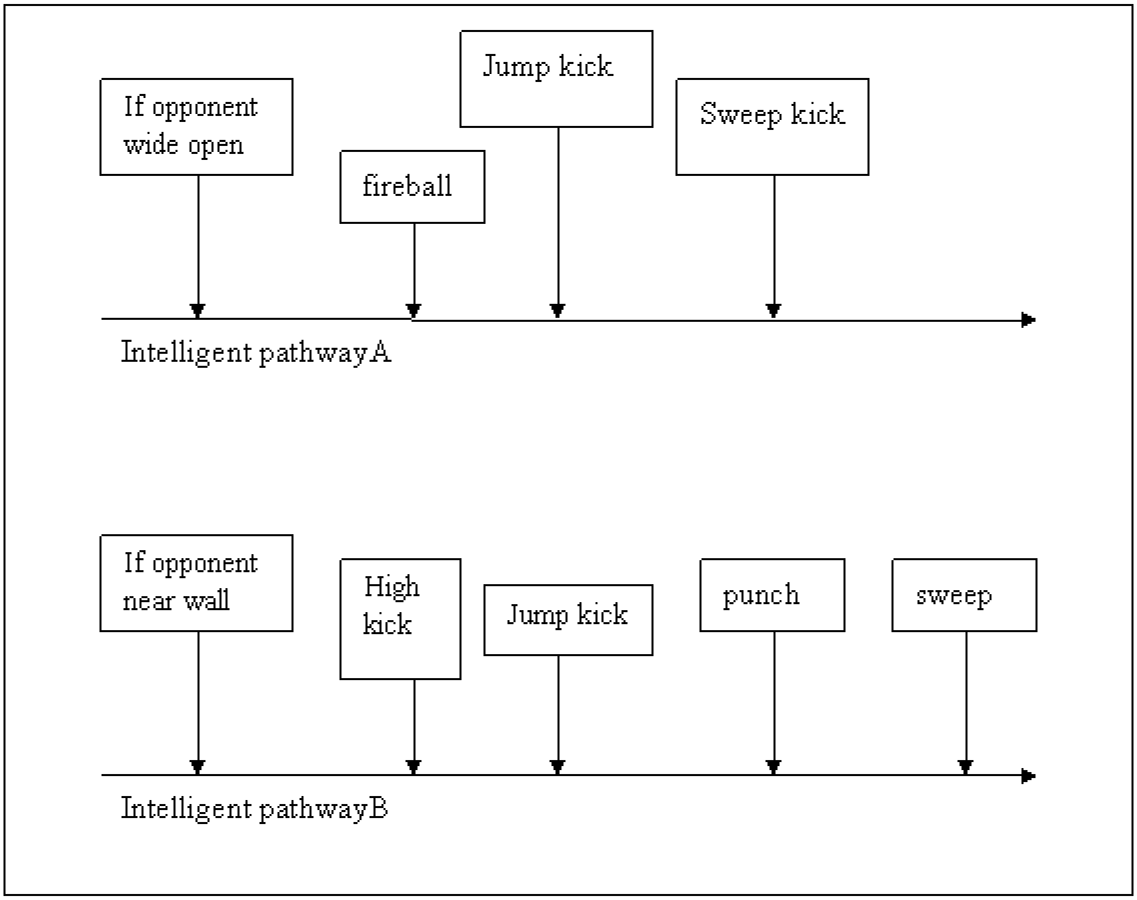
Let's say that in the magazine, the robot read an article about street fighter and there were 4 hints that the robot identified as being important. The 4 hints were extracted by reading the magazine, summarizing paragraphs, identifying important facts and so forth. The robot stores these hints in their respective storage areas in memory. When the robot plays street fighter, these hints will activate based on situations, whereby the situation has associational links with the hints. Let's say that the robot tried hint1 and hint3 and they work in the game. The robot will remember these hints in future similar situations. On the other hand, the robot tried hint2 and hint4 and they don't work in the game. The robot will not remember these hints in future similar situations. These bad hints will not be deleted from memory, they will only have low powerpoints. Data in memory can only be forgotten and not deleted. Besides, if the hint was correct, but the robot misinterpreted the hint, then the hint can be recalled in the future and the robot can reason that the hint was misinterpreted. This opens up the opportunity for mistaken or bad data to be recalled again.

These examples just show that the objectives and rules can be learned by reading books and being taught by teachers. However, the knowledge learned should be modified by the process of trial and error. This limits the amount of knowledge learned and greatly prioritizes knowledge.

In the game of street fighter, the robot must learn a lot of rules (mostly comprising if-then statements). If the opponent does this, then take this action. If the opponent does this, then take that action. Very complex combinations of moves must be used to defeat enemies. The robot might throw a fireball, jump kick, and do a sweep. Or the robot might do a high kick, low kick, helicopter kick and a fireball. The fun of playing street fighter is to chain linear moves together to maximize the damage of your opponent. This objective of the game may not be apparent to players, but everyone who plays street fighter have to accomplish this objective in order to beat the opponents.

FIG. 69 is a diagram depicting various intelligent pathways to play the game. A condition must be met in order to execute the linear tasks. In the first pathway, the condition is that the opponent is wide open. Being wide open means the opponent is open to an attack. When Ryu does a flying uppercut; and when he lands on the ground he is wide open. If guile does a razer kick and lands on the ground, he is wide open. When an opponent has spinning stars on his head he is wide open. The intelligent pathway can be used when any opponent is wide open. In the second intelligent pathway, the condition is that the opponent is close to a wall. When an opponent is near a wall the player can do aerial moves on the opponent and chain moves together.

FIG. 69



The intelligent pathways can have similar type of conditions or actions. For example, in intelligent pathwayB, the condition part can be any solid object behind the opponent. The condition part can be a sequence of criterias or it can be a range.

These intelligent pathways (rules) to beat the game came from another intelligent pathway to identify “workable” actions. The robot will set up goals like: chain moves together and see which combination of moves can give the maximum amount of damage. The result of trial and error will store these move combinations in memory.

Playing an unknown game require pre-existing intelligent pathways that guide the player to benefit in the game. The strategies that work will be kept and the strategies that don't work will be forgotten.

When all these rules and objectives are stored in memory, the robot can select which ones to use at certain times. The optimal pathways selected each millisecond will generate a computer program inside the robot's conscious to tell him the rules of a situation and what linear moves to use in the game.

The computer program inside the robot's conscious might have functions that can give facts about what the robot should do and the robot can select from a number of possible actions. For example, if the opponent is guile, the conscious will tell the robot to stop jumping over guile. The reason for this is because when a player jumps over guile, he will unleash his razor kick. While the task of not jumping over guile is followed, there are other functions that tell the robot to use various strategies to beat him. These strategies are learned through trial and error.

If the robot was playing against Dalsim, the conscious will give facts that the robot must follow, such as stay close to Dalsim and do not attack from far away. While these rules are being followed, the conscious will give specific strategies to the robot to beat Dalsim. A strategy would be to charge him and use any move in combinations.

If the robot was playing against Blanka, the conscious will give facts that the robot must follow, such as don't use the fireball. Blanka has a very fast jump time and when the robot does a fireball, it opens up the robot to attack. Other strategies will activate such as using specific sequence of tasks to beat Blanka.

For different opponents, different strategies are used and different rules are followed. In fact, for specific types of situations, a unique type of action is taken. All these strategies and rules are learned through trial and error or by learning information in videogame magazines.

Another factor is the mood and objectives of the player (the robot). The identity container in the robot's conscious will provide important facts of the robot in terms of decision making and rule boundaries. In this case, the personal rules and personal decision making of the robot will pour into the conscious. The robot might play the game for fun or to beat the entire game. Depending on the factors of playing a game, the robot will do things differently. If there is a scene where a combination of moves might not work, but the robot is playing for fun, he might actually use this combination. He has nothing to lose because he is playing for fun. On the other hand, if the robot was playing in a tournament and he was confronted with that decision, he will not select that combination. The reason why is because he knows there is a high probability that the combination will not work.

The robot might be playing against a kid and his goals are to play the game like a beginner. He wants to make the kid (opponent) feel good, so he is playing poorly.

One more note is that the robot can select any character to play in the game. He can play as Blanka, Ryu, ken, Chunli, Dalsim and so forth. Also, the videogame can select any character as the opponent. The robot has to be aware of who the player is and who the opponent is.

The identification of pain/pleasure events provides the basic strategies of the game. Through trial and error, being hit by an opponent will cause the robot's life energy to be lowered. This event causes pain. The robot (the player) also learned from playing other videogames that when the player's life energy is lowered it is a bad thing. When the robot blocks an attack from an opponent, he doesn't lose life energy. When the robot attacks the opponent, the opponent's life energy is lowered.

The identification of pain/pleasure events generates basic strategies for the game. The robot will know that: "he has to attack the opponent until his life energy is gone", "he has to block attacks from the opponent", "he has to face the opponent and be in close reach in order to attach". These rules are the basic foundations of playing street fighter and the robot learned these rules through trial and error.

Next, logic is needed to generate goals to benefit the robot in the game. One logical goal might be to: "hit the opponent with chain moves to maximize the lost of life energy". This goal is generated by logic. The robot is able to analyze the game and come up with beneficial facts to play the game. After the logic is generated, the task of the robot is to try the goal and see if it really does work. The trial and error generate more strategies for the game. These strategies are more complex and it benefits the robot more than basic strategies. Id of goals can also be read in magazines or spoken by a stranger.

Most strategies generated by the robot's conscious are from trial and error. When this condition happens, then take this action. When that condition happens, then take that action. These if-then statements (or combo of basic discrete math functions) will be stored in specific intelligent pathways.

Teaching the robot how to imagine in 3-d space

When the robot lives in a home for many years, his brain stores the 3-d sequences of walking through the home. In memory, he has a static 3-d representation of the house he is living in. This is important because the 3-d sequence of the home from memory will be accessed at some point to do reasoning. For example, if the robot was in the living room and someone from the bathroom flushes the toilet, the robot will know that someone is in the bathroom. The robot currently sees the living room, but doesn't see the bathroom.

If the robot was in the living room talking to friends and someone asked: "where is the bathroom", the robot will activate a simple 3-d surrounding of its current area and use this information to answer the question.

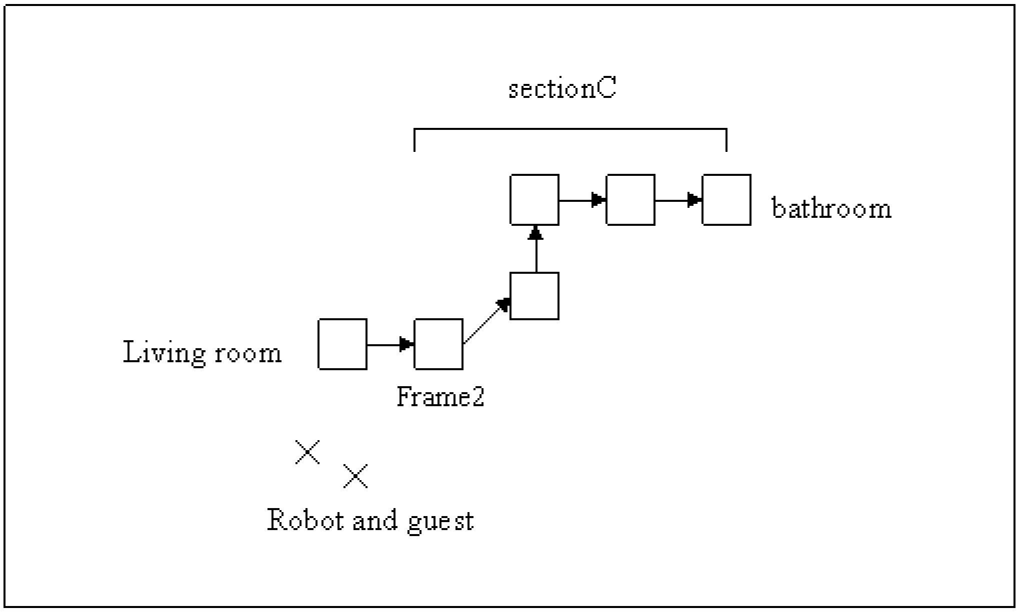
In order to answer the question the robot has to be taught how to answer the question. In elementary school, teachers set up a make believe environment. The robot is taught how to identify where he is currently at. The teacher will put the robot in a room and she will ask the robot where he is currently located. The robot will look around and answer the question.

Next, the teacher will ask the robot: "where is the bathroom?" The make believe environment is set up so that the rooms are identified easily. The robot will analyze the different rooms and point to a room with a toilet and a sink and say: "there".

In real world examples, a home is large and rooms can't be seen all at once. If someone asked the robot where the bathroom is, he has to activate a 3-d sequence of the home and locate the living room (the robot and the guest is currently located), and follow the 3-d sequence of getting from the living room to the bathroom. By analyzing certain areas of the living room, the robot knows which path leads to the bathroom.

FIG. 70 is a diagram showing the 3-d sequence to go from the living room to the bathroom. After activating the 3-d sequence, the robot's brain analyzed data in the frames and determines that frame2 is the area he should point to. In sectionC, the robot has to translate the direction someone has to travel to get to the bathroom. The two tasks, pointing to the direction and translating a path to the bathroom, are done by analytical pathways in memory. These intelligent pathways are learned previously.

FIG. 70



The teachers can use sentences to control how the robot imagines things. Sentences like: "imagine your house and see where you are currently at and determine travel paths to get to the bathroom?" or "imagine your house and give directional path from the current location to the bathroom?". These sentences are guides that will hopefully form intelligent pathways to teach the robot how to answer 3-d locations. Although the robot can't see the entire house all at once, he can extract the 3-d sequences of the home in memory and use that information for reasoning problems.

A more difficult task is to visualize entire cities and states. Maps are used or spaced-out still pictures of far away places are used. Sometimes driving in a car will help speed up the travel from one location to the next. The robot can use car driving 3-d sequences to map out entire cities. If the robot walks around the city, the map would be large. However, if the robot drives around the city, the map will be small.

Other learned 2-d still maps of the city can activate and the robot can use those data to answer location questions. 2-d maps of the state are used for planning routes. This map can also be used for showing distance between two areas. The map statistics should be the measurement for distance. For example, in the map, the legend says that 1 inch is equivalent to 1 mile. If the robot is asked a distance question such as: "how far is it from town1 to town5"? The robot will activate the 2-d map and analyze the approximate distance. He determines that the distance between town1 and town5 is equivalent to 3 inches. 3 times 1 mile equals 3 miles. The robot will remember the data "3 miles" and tell the person: "it takes 3 miles to drive from town1 to town5".

Other times, the robot will take still pictures of a location from memory and use analytical skills to interpret distance. The still picture is the current location of town1 and in the far distance, the still picture has a view of town5. The robot can use logic about distances in pictures and determine how far it is from the current location (town1) to the destination location (town5).

Distinguish knowledge in science diagrams

When a science diagram is stored in memory, there are many different types of data. The robot's brain will distinguish different data types by tagging most likely sequences together. A visual human body is one type of data and the connections weights will be unique. Facts about a particular body part, like the heart, is another type of data and their connection weights will be unique. There should be a clear distinction between searching information on visual human body and facts about the heart.

Using logic to assign objects/events/actions to words

Very complex words are represented by logical identifications. The sentence below needs the robot to identify all if conditions. If all conditions are true, then the word will be assigned to an object, event or action.

Sentence: If he looks left and his explanation is unbelievable and he is doing hand gestures, then he is lying.

In the sentence above, there are three conditions that must be identified in order for the person to be lying. The conditions are: he looks left, his explanation is unbelievable and he is doing hand gestures. If all three conditions are identified by the robot, then the action "he is lying" is true.

In this case, lying is an action of a person. In order to determine if it exist for the person, three conditions must exist. If the three conditions exist then that means the person is lying.

In a dictionary, a word comes with a definition. The example below shows the definition of the word aggressive.

Aggressive: a person is aggressive if he/she is angry and uses physical force.

This sentence has a pattern to identify aggressive behavior in a person. The next time the robot sees Jake angry and he uses physical force, the word aggressive activates and is assigned as a behavior belonging to Jake. There are two conditions in this word: "the person is angry" and "he/she uses physical force". Once the robot recognizes the two conditions, he will activate the word aggressive.

Similar words can also be distinguished by using logic. The word hop and jump are very similar. Someone can jump and the teacher will say that he/she hopped. Teachers can teach the robot what the distinctions are to better understand what a person is doing. Let's say that the teacher said that a hop is when someone jumps straight up in the air and a jump is when someone jumps in any direction. The teacher will demonstrate using her body.

What is this action? -- (teacher jumps straight up) "this is a hop".

What is this action? -- (teacher jumps forward) "this is a jump".

In both examples, the teacher is giving the question and the answer (supervised learning). The robot will create patterns between similar examples and understand what the distinction is between similar words like hop and jump.

In more complex cases of identifying objects/events and actions, probability and calculations must be made during runtime to determine what is happening. A sentence like: "if the clouds in the sky are equally distributed, then it will rain", will require the robot to look at the sky and see if the clouds are equally spaced out in the sky. If they are then the robot's brain will activate: "then it will rain".

Learning probabilities can also be factors to identify objects, events and actions. Maybe, the conscious builds a specific type of computer program that will balance probability of things happening.

An example would be this sentence: "if the house is messy and the drawers are pulled out and the front door is ajar, then the house has been robbed". The robot has to look around and see that the house is messy and the drawers are pulled out and the front door is ajar. When all these conditions are met then the thought: "I've been robbed" activates in his mind.

Patterns to search functions in intelligent pathways

In search engines, the input data are text words typed into the search box. These text words are the input variables for the search function. The human robot doesn't define the variables to search for information in memory. Instead, the intelligent pathways in memory self-define variables to search for information in memory. One really good example is questions and answers. The robot learns the steps to answer a question from teachers. When the robot learns many similar examples, these examples self-organize in memory. During the self-organization process, patterns are formed that define what the search variables are.

Thus, unlike search engines, the human robot uses patterns in intelligent pathways to define what types of variables to search for in memory. Also, the type of output from the search function is defined by patterns. The current search engines use computer scientists to write the codes for the search functions (only some have adaptive properties). A human robot learns how to search for information in memory, in terms of what to search for and what to output.

The robot takes in 5 sense data from memory, called the current pathway. Also, the robot has conscious thoughts like its goals, intensions, rules of searching and so forth. Based on the robot's 5 senses, as well as his conscious thoughts, the search variables are defined. You can say that the search variables would be: the robot's tasks in the task container, the rules in the rules container, the facts in the identity container, the intelligent pathway instructions outlining search patterns, the id of target objects and their activated element objects and so forth. All these variables are used to search for data in the robot's brain.

Current search engines, try to use software to guess the intentions of the user such as goals and rules, but they don't have an exact assumption. If you think about it, its not really the search engines that is doing to real intelligent task of searching for websites, its actually the user that is performing the intelligence.

Learning, configuring and storing data

The robot learns information through its 5 senses: sight, sound, taste, touch and smell. However, there are different types of learning media. A robot can learn from a teacher through classroom lectures, or reading a text book, or watching Sesame Street on TV. Different medias will store information differently. In this section, I will outline how data is learned, configured and stored in memory.

Most of the data stored in memory are stored in a 3-dimensional visual format. The majority of data in the brain is made up of visual images and movie sequences. When they self-organize, all 5 sense data will be centered around visual images. If the robot reads a textbook, the information from the textbook will be stored as a physical visual book in memory. Since the robot reads the book linearly – read page1 before reading page2 – he will store the first page in memory exactly to the way he encounters the first page. Page1 will be stored to the right and page2 will be stored to the left. Next, when the robot flips the page, page3 and page4 will be present. Although, page3 and page4 exists in the same area as page1 and page2, the robot understands that the pages are configured compactly.

If the robot reads a textbook, the entire physical text book will be stored in memory. All contents read will be stored in memory in a linear and organized way. Questions asked about the book can be answered such as: what was chapter1 about, what's the difference between chapter1 and chapter6, how long did it take you to finish reading chapter3, what chapter talked about evolution? These questions are answered by activating the text book from memory and flipping through the pages to look for information. The robot has to search for information in an approximate manner and might make mistakes. For example, he might mistakenly store chapter9 after chapter10.

Other topics (part 2)

The conscious can fabricate character voices

I'm a huge fan of the x-men comic books. I like to read back issues of the x-men on my free times. When the cartoon series came out, I started to watch those too. In the comic books, I have an idea of what the characters look like from still pictures. In the cartoon episodes, I have an idea of what the voices of the characters sounds like.

One day I was reading an x-men comic book and something very strange happened. I noticed that while I was reading the spoken captions, the voices of the characters are presented. For example, if I read the caption for wolverine, the voice of the text in my head sounds like the voice of wolverine from the cartoon. When I'm reading the caption for professor X his voice from the cartoon is presented in my head.

Somehow my brain created a computer program that can translate text captions from a character (in a comic book) and output the voice of the character (in the cartoon) in my mind. This is remarkable because this is impossible to do in current software.

The reader can try this at home. Watch a movie like Harry Potter and then read a Harry Potter book and you will actually hear their respective character voices while they speak.

The conscious can override certain dominant rules temporarily

The computer programs inside the robot's conscious can disable certain dominant rules temporarily. For example, if the robot was playing a racing videogame and there is a rule that said: "If the sky is green, the traffic light rules are reversed". This means that if the sky is green in the videogame, the rules of the traffic lights are in reverse – the red light means green light and vise versa. The dominant facts in memory are: the red light is to stop and the green light is to move forward.

In this game, the dominant facts are extracted and the conscious manipulate the facts so that they are changed according to the videogame's rules. The example below demonstrates this point.

Dominant rules:	Modified rules:
Green light = move forward	Green light = stop
Red light = stop	Red light = move forward

The computer program inside the conscious was able to take dominant data and to manipulate them using other rules. This type of temporary rule modification is used only in this videogame. If the robot plays any other racing videogame or drive a real car, the dominant rules will be used. This example shows that the conscious can disable a dominant data in memory temporarily and modify and use them under rare conditions.

To make this example more complex, imagine that the robot has current science knowledge about the world and how it works. An instruction was given to the robot to pretend like the year is 1700; and the robot is a character in this year. He was also instructed that all current science knowledge after 1700 are not known yet and that the robot has to answer questions according to the year 1700. The first question asked is: "is the world flat?". In 1700, the majority of people living in that time period have no idea what this question means and how to answer it. The robot has a dominant data in memory which is the world is round. However, the robot has to pretend like he is in 1700, so the conscious tells him to say yes to the question.

The second question is: "Do you like to drive your automobile?". According to the robot, he loves to drive his automobile, however, the automobile wasn't invented until the early 1800's. The robot will probably give this answer: "what is an automobile?".

The robot is given a task to be a professor in 1700 and to explain how the human brain works. The robot has current knowledge about the brain and how it works, however, in 1700, people have wrong information about the brain. The computer program inside the robot's conscious has to take what it knows about 1700 medical knowledge and explain this knowledge to an audience. Back in 1700, they didn't know the various components that make up the brain. They simply know that the brain was responsible for thinking and acting. They didn't know about neurons and chemical signals. Some information might be wrong such as the brain was one unit. Later on, people found out that the brain had interconnected hemispheres: left brain and right brain.

The idea I'm trying to convey is that the robot has all this knowledge in memory. The intelligent pathways are able to process data and output meaningful facts based on a situation.

The conscious can correct data in memory quickly

If the robot was taught that the world is flat for many years and one day someone published an essay that explains the truth -- that the earth is round, how will the robot correct the wrong data in memory? It really doesn't matter how strong data is in memory or how much times the robot learned this data. The robot's conscious can correct any data in memory instantly.

The world is flat is a very dominant fact for the robot. However, intelligent pathways are able to put a tag on this dominant fact and point it to another fact. A teacher might teach the robot: "no, you are wrong. The earth isn't flat, it is actually round". These two sentences basically put a tag on the dominant fact: the earth is flat and put a pointer that points to a new fact: the earth is round. The next time the robot searches for the fact: the earth is flat, it will see that there is a pointer, stating this fact is wrong and the correct fact is: the earth is round. As time passes the wrong fact is forgotten and the correct fact replaces it.

Predicting the future by using activated thoughts

My AI program (the human robot) has two methods of predicting the future. One is by matching the current pathway to the best pathway in memory and analyzing which future pathways benefit the robot the most. The second method of predicting the future is by using intelligent pathways in memory to extract specific future data.

Different types of future predictions are used for different situations. The robot must learn how to predict the future. In order to do this, teachers have to teach the robot these lessons. If the robot was asked a question: "who will most likely succeed in life?", and two pictures are presented. One picture represents a high school drop out and the other picture represents a college graduate. The robot will analyze each person and predict what will happen to them in the future. Next, the robot will compare the two futures and decide which person will most likely be successful.

The intelligent pathway to answer this question will require coming up with linear predictions of where each person will be in the future. For example, the college graduate will look for a job and he/she will most likely get hired by some company. On the other hand, the high school drop out will also try to get a job and will most likely be turned down.

Other facts must be considered. What if the college grad had a major in art? This would not give that person a leaping advantage over the high school drop out. The high school drop out might be blessed with good looks and get hired as a movie star. These external factors also have to be considered before answering the question.

The point I'm trying to make is that the intelligent pathways to answer the question will do things intelligently so that the answer to the question can be done in a logical and efficient manner.

The teachers in school will teach the robot how to predict the future through worksheets and examples. After many similar situations, the robot will have universal pathways stored in memory to predict the future efficiently. Below are lessons that teach the robot what to do in the future under a specific situation. In some respects, these lessons are like if-then statements. If the robot recognizes R1, then do R2.

1. When you see a dangerous animal, run and get help.

2. On weekdays go to school.

3. Eat lunch at 12 o'clock everyday.

4. When you can't solve a problem apply the scientific method.

These lessons are very basic. The brain of the robot will identify these lessons and store them in an organized manner. In the future, if the robot was in the swamps and sees an alligator, the search function might take variables: alligator and danger; and the conscious might activate "run and get help". The target objects (or variables) danger and alligator has strong association with the rule: "when you see a dangerous animal, run and get help". The search function will take specific target objects and the conscious will activate their strongest element objects. In this case, the robot recognizes alligator and danger. The robot's conscious took these two target objects and activated the strongest element objects between them, which is "when you see a dangerous animal, run and get help". The actions: "run and get help", is the future actions the robot will take.

Predicting the past/future using linear tasks

If you watch CSI, you will notice that the investigator's job is to predict what happened in the past. The investigators have to use technology to find out what happened in the past regarding a case. Their findings will be plotted on a timeline. In order to do this, the investigators have to know what the steps are to predict the past. These investigators, no doubt, had to go through criminal school in order to get their job. In criminal school they teach you the procedures and steps to generate a timeline of a crime.

The information in the timeline can be general or specific (or both). The way that the teachers teach the investigators determines how the timeline will be plotted. If a crime happened and the crime lasted for 2 years, the timeline might have general spaced-out events of what the criminal did in the last 2 years. Or the information in the timeline might be general information; with a few specific information added.

The conscious help solve problems when stuck in a videogame

A role playing game like Zelda requires deep logic in order to solve problems in the game. If the robot is "stuck" in the game and he doesn't know what its objectives are and where to go next, he needs to use deep logic in order to solve this problem. It is frustrating when you play Zelda and you are stuck in the game. You try something and that something doesn't work. You try something else and that something else doesn't work. At this point, the robot will use lessons taught in school to identify a problem, plan steps to solve the problem, try a strategy and repeat itself over and over again until that problem is solved.

Usually, when the robot is stuck in the game, the objective is to find out what the next mission is or what areas to go to. The robot was taught that if it gets stuck, it has to travel all over the land – in caves, in villages, and in forests to gather information. Talking to characters in the game is a very valuable asset in terms of information gathering.

For me personally, when I get stuck in the game, I would roam the land going from place to place and talk to as much characters as possible. Some places in the game are hidden to the player and requires bombs to reveal their entrances. I would keep a note about where I went and what information was gathered in respective places. Once I checked all places, I would assume where hidden entrances might be located and use bombs to see if there exist any hidden entrances. Sometimes characters would say: "there is a hidden entrance to level4 by lake hylide". This information will tell me a possible area where the hidden entrance is. When I get to lake hylide, I would look for obvious places where a hidden entrance might be located. Finally, I would put bombs on places I think the hidden entrance might be. If the bomb explodes and a hidden door is revealed, then I have solved my original problem. This entrance leads me to the next level (level4) which is the solution to the problem of where to go next.

If the hidden entrance doesn't appear and I have no idea where the hidden entrance is, I will have to start all over again. This failed trial can be used as information that I had gone to lake hylide and checked the area for hidden doors, but was unsuccessful. My conscious will tell me, in the future, I have done this task and I won't repeat it again. This narrows my future possible strategies.

At this point, I have to search and remember the quotes from previously encountered characters and use logic to determine where the hidden entrance might be. Maybe a villager I encountered told me that his father was wondering in demon cave and he never came home, might be a clue to the hidden entrance. Maybe his father went to demon cave and accidentally found the hidden entrance and was captured by the guards. This clue will prompt me to investigate demon cave. This quote might also be a hint, whereby the villager wants me to go to demon cave to rescue and bring back his father (a new mission).

An easier way to solve problems is to read strategy guides for Zelda. They have specific information regarding where the hidden entrance is to level4. Once the robot reads in the secret, he will know where to go and what to do next.

Sometimes, solving problems can be by accident. In the past when I was playing Zelda, I remember I was trying to find a hidden entrance to level5 in the forest. I was in hylide forest trying to attack enemies with fire when I accidentally burned a tree, revealing the hidden entrance.

In another RPG game called super metriod, the player has to go from room to room to beat bosses and complete missions. In some cases, the character in the game can be trapped in a local area and in order to get out, the character has to find hidden rooms. Many hidden rooms are found by blasting the environment. When I get stuck in the game and I have no idea where to go next, I simple go from room to room and blast the environment (making sure that I don't repeat going into same rooms). I tried this method on several occasions and I was successful in beating levels.

This method of going from room to room in an efficient manner (no repeated travels), was generated by my logic. Through trial and error I was able to come up with this optimal strategy; and this strategy wasn't read from a magazine or a strategy guide.

There are also other methods that work for solving a problem in super metroid. During the problem solving phase, my brain was able to go through the possible methods and select the most optimal for that problem. If I failed in solving the problem I will select another optimal method to solve the problem.

The point I'm trying to make is that human logic is needed when the robot is stuck in a videogame. He needs to solve the problem by using lessons taught in school to find solutions to problems.

The conscious can determine distance by using visual data in memory

The robot senses 2-d images from the environment, but the brain stores them in a 3-d manner. The 2-d images are stored in a 3-d grid and a 3-d model of the environment is formed in memory. The robot's conscious can take visual information and determine distance.

There can be different types of visual data extracted from memory. One type is a still picture of a memory. The conscious can use intelligent pathways to analyze the still picture and determine distance. For example, in the still picture is one building and a bridge. The robot's conscious can analyze this memory and determine that the distance between the bridge and the building is approximately 50 yards.

In this example, the teachers teach the robot how to determine distance based on a still picture or in real life.

The conscious can also fabricate a map of cities and determine the approximate distance. Teachers use worksheets of maps and they want the students to write down the distance between two cities. Let's say the robot is familiar with the entire island of Hawaii, and the robot was given a map worksheet of Hawaii. From encountering the environment for many years the robot has an approximate distance of the cities. If the teachers teach the robot the exact distances between cities, then the map can determine distance standards.

If the robot were to live in California and remembers all the locations, he can activate a fabricated map of California and determine the approximate distances between cities. This fabricated map of California was generated by the lessons taught in school using Hawaii maps.

Learning to search for information

Teachers teach the robot how to search for information. These intelligent pathways will be used to search for data in memory. The robot can extract a data in memory and use a specific type of search, according to its goals, and extract relevant data. He can take a memory of an event that happened 2 days ago and fast forward or backwards to find information. He controls the speed of the fast forward or backwards – he can slow down the speed or increase the speed of the search.

Other intelligent pathways are needed for the search, such as what to look for and how focused is the search. The search can be cursory, whereby the robot is analyzing the movie sequence with moderate focus. On the other hand, the search can be detailed, whereby the robot is analyzing the movie sequence intently.

He can also control the demographical areas in the brain to search for information. Should the search function be spaced-out or only localized in one general area. The search function can also focus on 2 or more general search areas simultaneously. The lessons in school to search for things in the real world create the intelligent pathways to search for data in memory. For example, the teacher can teach the robot to rewind and fast forward a movie in a DVD to look for information. This intelligent search pathway can be used to search for data in a recalled memory (note: a recalled memory is similar to a DVD movie).

Teachers can teach the robot to search for a specific type of media for an object. For example, if the robot is coloring a book on birds, the robot has to have an image of a bird to understand what the colors of a bird look like. The object is a bird and the robot has to search in memory for an object composite of a bird (also called a floater). Within the bird floater, the search function is only interested in extracting visual images of a bird. The visual image of a bird activates in memory and the robot will color the picture according to the color in the robot's mind. This example shows that during a situation, the search function can search for specific types of data related to an object composite.

Maybe the robot's brain can form intelligent pathways to search for data in memory based on math diagrams. The sight images are structured in terms of circle boundaries and the sound data are structured in terms of square boundaries. The intelligent pathways in memory can form "any" search algorithm in a network (the robot's memory). The brain will configure the data from the current pathway in terms of math diagrams and use a patterned search method in memory. This pattern search method is created in intelligent pathways through lessons in school.

Searching data in terms of a book or report

When the robot reads a book linearly, the robot's brain stores a physical approximate 3-d book in memory. The robot can flip through the pages, zoom into small areas of pages and so forth. The robot can recall a report about a person he read 2 days ago. He can analyze areas of the report. Let's say that the report list the background information about dave. The robot can flip pages of the report, zoom into the name slot, the occupation slot and even analyze the picture of dave. Thus, static data stored in memory isn't based on one full data, the robot has free-will to search for minor details of the static data – he can zoom in or zoom out or pan right or left of the static data to search for information.

If the robot picks up a book and he flips to the first page, he will see the overall page, but he can't see the text in detail. Only when the robot zooms-in and see the text will he be able to understand what is on the page. Only when the robot read text sequentially will he understand the content of the page. Sometimes there might be pictures or diagrams in the page and the robot has to zoom-in to analyze the picture before it can be stored as static data in memory. If the robot wants to recall the picture on the page, he has to access the zoom-in frames of seeing and recognizing the picture. This is a very powerful learning tool to analyze static data in memory.

Self-Awared robots

The definition of a self-awared robot is a robot that can control all aspects of its decision making. Every second that it lives, the robot is aware that it's alive and that it controls its body to change the environment. The intelligent pathways in memory self-organize and they are structured hierarchically. The ABC block problem is one example. The blocks can be positioned in any order or in any position, the robot will still be able to stack them up in an ABC manner. The reason behind this is because the robot was taught many examples of the ABC block problem by teachers. The teachers guide the robot to do the necessary steps to accomplish the task correctly. As the robot's brain self-organizes these examples of the ABC block problem, he was able to form universal pathways. These universal pathways will allow the robot to solve the ABC block problem regardless of the environment – he can solve the problem in school, at home, at a park or in a train. The world can be upside down, but the robot will still be able to solve the problem.

In Nils J. Nilson's Stanford University book, called Artificial Intelligence: A new synthesis, he uses a combination of recursive planning programs (called the strip program), discrete math, predicate calculus, induction/deduction, semantic models, language parsers, search algorithms and ruled based programs just to solve the ABC block problem. The distinction between his methods and my methods are completely different. Mr. Nilson's AI book was used in my artificial intelligence class in late 2003.

The way I solve the ABC block problem comes straight from the robot's conscious. The teacher will tell the robot the main objective: "stack up the blocks in an ABC format". The teachers will also guide and give commands to the robot in a step by step manner. The robot is given this command: "identify the C block and put it on the floor". The robot has learned to do this task previously by a teacher and understand the instructions. The robot will identify the C block, which is to look around and recognize the C block. It will grab the C block. It doesn't matter where the C block is or how it is stacked with the other two blocks. The command is to grab the C block. If the C block is on the floor and the B block is stacked on top, the robot will take the B block and put it on the floor and grab the C block. When the C block is grabbed, the robot will put in on the floor. If the robot does something wrong the teacher will guide the robot to do the correct thing. Next, the teacher will give another command: "identify the B block and put it on the C block". The robot will identify the B block and grab the B block. If the B block is on top of the A block, the robot will simply grab the B block. Then, the robot has to identify where the C block is because the command given was to put the B block on top of the C block. When the C block is identified, the robot will carry the B block and put it on top of the C block. Finally, the teacher will give the last command: "identify the A block and put it on top of the B block". The robot will identify the A block, and then put the A block on top of the B block.

Doing the task in real life and physically handling the blocks is a must in order to understand objects and 3-d space. It will also identify states of the blocks. If the B block is on top of the A block, the B block isn't on the floor. The fact that the robot see the ABC blocks stacked a certain way declares that state. When the ABC blocks are scattered all over the place, the problem is at its beginning stage. If the C block is on the floor and the B block is on top of the C block and the A block is on top of the B block, then the problem has ended.

When the teacher teaches the robot many examples, he will eventually get it. If the teacher does her job correctly and successful forces the robot to carry out steps to stack the blocks, then the robot will have created intelligent pathways in memory to solve the problem. These intelligent pathways go through self-organization to form universal pathways so that the robot can solve the problem under any given environment.

When the robot watches TV, he won't be watching the TV in a systematic way. The robot wonders around in the TV monitor looking at objects to focus on that gets its attention. If there were 4 anchors and each anchor takes turns speaking, the robot will focus on the anchor that is speaking at the moment. Sometimes, it focuses on something else on the monitor such as an object an anchor is describing. The robot chooses to focus on things it wants to focus on. The robot's choice are based on innate behaviors and by lessons learned from teachers.

Sometimes, the goals and rules of the robot determine what to focus on in the TV monitor. Strange events in the TV might focus the robot's attention toward a minor object. Maybe a goal of the robot is to pay attention to the background noise and not the news anchors or to pay attention to the small bird outside the window.

Being self-awared also means that the robot can control a wide variety of things the robot's conscious is responsible for, such as: analyzing a situation, managing multiple tasks, outputting logic, providing meaning to language, giving facts about objects, solving interruptions of tasks, solving a problem, focusing on objects, predicting the future, fabricating maps, planning routes on the fabricated map and so forth. The robot controls all aspects of how the conscious thinks and acts. This control gives the robot a sense of self-awareness. All human beings and human robots work the same way: to pursue pathways in memory that will lead to pleasure and to stay away from pathways in memory that will lead to pain.

Friendly conversation inside the robot's conscious

The voice inside the robot's conscious is the engine that does everything for the robot (act and think). The robot's conscious can: manage tasks, execute multiple simultaneous tasks, solve interruption of tasks, provide knowledge about a situation, provide meaning to language, solve problems, focus attention, search for data, forget data, modify data, take action, make decisions, provide sets of rules while doing a task, and provide a friendly conversation.

I think of this voice as the robot's friend. Actually, the voice is itself, but the robot can imaginarily think that it comprises 2 people or a group of people. In fact, the friend in its mind can be different people. The robot can be talking to itself like he is talking to a parent or a friend or a teacher or a group of friends.

When I was in college, I attended an art class and after each assignment is finished, we would post our artwork on the wall and everyone has to comment on each artwork. The comments can be question-answer, facts about the artwork, descriptions, personal likes/dislikes, comparisons, advice, logical fact, joke comment and so forth. The collective voices of not only the teacher, but the robot and other students are stored in memory based on analyzing artworks.

The next time the robot has to analyze an artwork, the collective voices of a group of people will activate in the robot's mind. The robot is debating and commenting and giving opinions in his brain about the artwork. Sometimes the robot is saying something to the friend and other times the friend is saying something to the robot. The robot might ask a question and the friend answers the question.

The robot can talk to itself like a parent. Maybe the robot is doing drugs and his conscious is being honest with the robot. Things this friend will say will sound a lot like his parent. Some sentences might be: "don't do it. drugs are bad for you", "I will ground you if you take drugs". On the other hand, bad friends will talk to the robot as well, such as "take the drug. If you don't you will look weak" or "you chicken".

The robot can also learn to think a certain way. Teachers can teach the robot to have a meaningful conversation in its mind. For example, a teacher can say this to the robot: "when you are solving a problem, talk to yourself and debate a way to solve the problem". When the robot is solving a problem and the problem is difficult, it will talk in its mind in a group like setting; and being both the person asking questions and the person giving the answers.

Popular learning commands are: "when you are bored, do something", "at lunchtime, plan a restaurant to go to", "when you have free time, think of things you like to do". These are questions that teach you to think when you are in a specific situation. For example, if the robot is at home, bored and has nothing to do, he will think of something to do.

In other cases, the friendly conversation is created from the absence of friends. Let's say the robot was a very social guy in high school. Everywhere he goes, he has lots of friends to talk to. After high school, he went to college and most of his time is spent studying. When the robot has free time and there are no friends around, the obvious thing is to talk to the air. The speaker and the listener is the robot. We can take it a step further and say that if the robot was in public, but doesn't have a friend nearby, he will talk in his mind. He does this because he doesn't want people around him to think his crazy.

If you think about how ideas come to the robot, the voice in his head is debating to generate ideas. When the robot is in a classroom, he is surrounded by ideas thrown by other students. These group ideas didn't come from the intelligence of the robot. During the self-organization process, the group ideas stored in memory form the intelligent pathways so that the robot can generate ideas that come from someone else.

Also, teachers are a big factor in the development of the robot's conscious. When the robot is solving a problem or reading a complex text book, the voice will be a teacher guiding you, teaching you and giving you information about how to read the text book. The reason why is because the robot learn science knowledge and complex knowledge based on teachers. It's natural that the robot activates voices of teachers in his mind when he reads a book related to school subject matters.

Finally, if the robot learns to comment about a person in a party from family members and personal friends, then their collective voices will be in the robot's mind when he is in a party. In the party, the robot's mind will be thinking: "that is one sexy girl", "the music is bad", "I don't like the food served here", and "don't say anything bad about party members".

Teachers teach the robot how to do math equations in school by using fingering pointing as a guide to focus the robot's eyes on important numbers. If the robot watches the teacher give a lecture on solving addition problems, the lesson will be stored in memory. Both the linear steps to the equation and the guidance from the teacher are stored in memory.

When the robot has to do the addition problem by himself, the conscious will activate the teachers finger to point to vital numbers. Even though the teacher isn't there, the association between the addition problem and the teacher's guidance (finger pointing) are so strong in memory that when the robot is solving an addition problem the teacher's guidance automatically activate. In some respects, the addition problem is the target object and the teacher's guidance is the activated element objects.

Translating languages

Let's say that the robot is a translator and his task is to translate sentences spoken in German into English. A speaker will speak in German and the robot will activate the meaning to the sentences. These activate thoughts will come in the form of 5 sense data or a fabricated movie. The robot will simply use an intelligent pathway in memory to analyze the fabricated movie and translate the images/5 sense data into English sentences.

It's no different than activating thoughts in the robot's mind and using intelligent pathways to convert 5 sense data into linear structured sentences. For example, the sentence spoken in German might be: "the dog jumped over a box". The robot will activate the meaning to the sentence in his mind. Images of a dog leaping over a box are fabricated in his head. Next, the robot will remember this fabricated movie and using an intelligent pathway from memory to translate what he is seeing in his mind into structured English sentences. The subject is a dog, the action is jumped and the other object acted upon is a box. Grammar rules activate such as subject-predicate. The robot will construct this English sentence: "the dog jumped over the box".

Modifying massive facts in the robot's brain by reading books or attending lectures

Let's say that the federal government decided they wanted to change all the rules of driving -- green light is red light and red light is green light, etc. Every driver has to read and understand all the new rules and apply them to driving immediately. For human beings this task is easy, but for computers this task is very complex.

In the robot's brain, there is a section that is devoted to facts, rules, and procedures in terms of driving. Networks of data pertaining to one subject matter (driving) are neatly organized in memory in a hierarchical-associated manner. The task of the robot is to read and understand a book on the new rules of driving. He has to insert all the new rules and procedures and to delete all the old rules and procedures in memory.

As stated before, the intelligent pathways in the robot's brain can form any type of computer program to do work. In this case, the conscious will create a computer program to insert new facts and to delete old facts relating to driving. This computer program is putting high priority on every single new fact identified in the book and it is deleting any old facts stored in memory. Remember, the old facts aren't actually deleted from memory, they are simply forgotten.

Think of the computer program inside the conscious as a software that generates functions to modify a massive database system. There are thousands of rules and procedures in terms of driving and this computer program has to modify specific rules and not all rules. Some driving rules are still followed. A new rule might be: green light is red light and vice versa, but the yellow light remains the same. The robot will modify data in memory based on the sentences read in the new book. During the reading of the book, the robot's goal is to remember all information in the book into memory and to identify old respective rules from memory and delete them. After reading the entire book 2-3 times, the database in the robot's memory, in terms of driving, is modified. All or most of the new rules are stored in memory and most of the old rules are deleted from the database.

The uniqueness of a human brain is the ability to forget information. After 5 years implementing the new rules of driving, if the government decides to change the rules back to its original format, human beings can accomplish this without reading a new book. Their brain can remember what the old facts about driving are. For example, their brain can tell them: the old rule about traffic lights is red light means stop, green light means move and yellow light means pause. However, it would be more efficient if each driver simply read a book on the old rules of driving.

If you look at current database systems using SQL or robot cars, human programmers are needed to write the functions to modify information. For example, if the government changes the rules of driving, a robot car requires that human programmers modify the rules and procedures "manually".

Condensing information about people, places, and things through words

Imagine personC has a file consisting of 100 pages worth of background information. Computers can save a lot of diskspace if they use individual words to describe a person. Those 100 pages on personC can be represented by 2-3 words. In some cases, personC can be represented by 1 word.

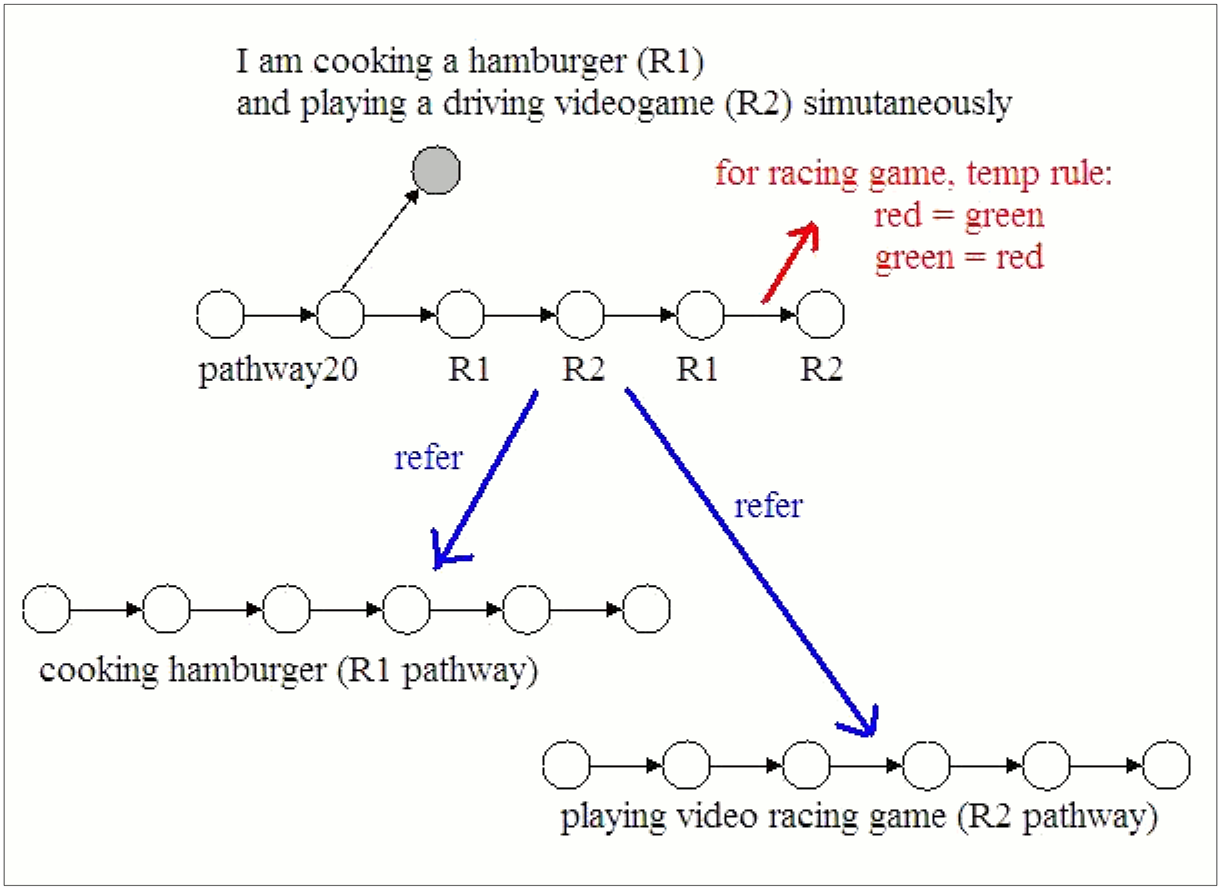
The robot's brain basically condenses information by using individual words to describe a person, place or thing. When the robot recognizes an object, the most important facts about that object will be presented to the conscious. These facts will mainly consist of individual words. If a person loves to read alot, the words "nerd" basically represent that person. If a person is obese and is trying to lose weight, the word "fat" basically represent that person. If a person has been arrested 20 times and do drugs, the words "repeated criminal" describes this person.

With words like "repeated criminal", people have an idea, based on stereotypes, who this person is, what they did in the past, and what they will do in the future.

In some cases, people, places and things can be described by a picture. The robot is able to know a lot about a person's background through 1 or several pictures.

Referencing pathways

English sentences stored in pathways can reference other pathways. The diagram below shows a pathway to reference other individual pathways. In this case, pathway20 is a pathway that does 2 simultaneous tasks (R1 and R2). The sentence: "cook hamburger and play a racing videogame simultaneously" manage 2 tasks simultaneously by switching between tasks until both tasks are completed. Pathway20 is known as a universal pathway because R1 and R2 can be any task. Pathway20 simply references individual task pathways in memory to manage 2 simultaneous tasks. For example, R1 can be cook pizza or R2 can be clean the living room.



In pathway20 there is also another task, which is to temporarily change the rules of the racing game. The instructions include following a temporary rule: red light is green light and green light is red light. The dominant information in memory is green light is go and red light is stop. However, the temp rule allows the robot to temporarily follow the opposite rule. This shows that pathway20 can temporarily change the rules and goals of pathway R2.

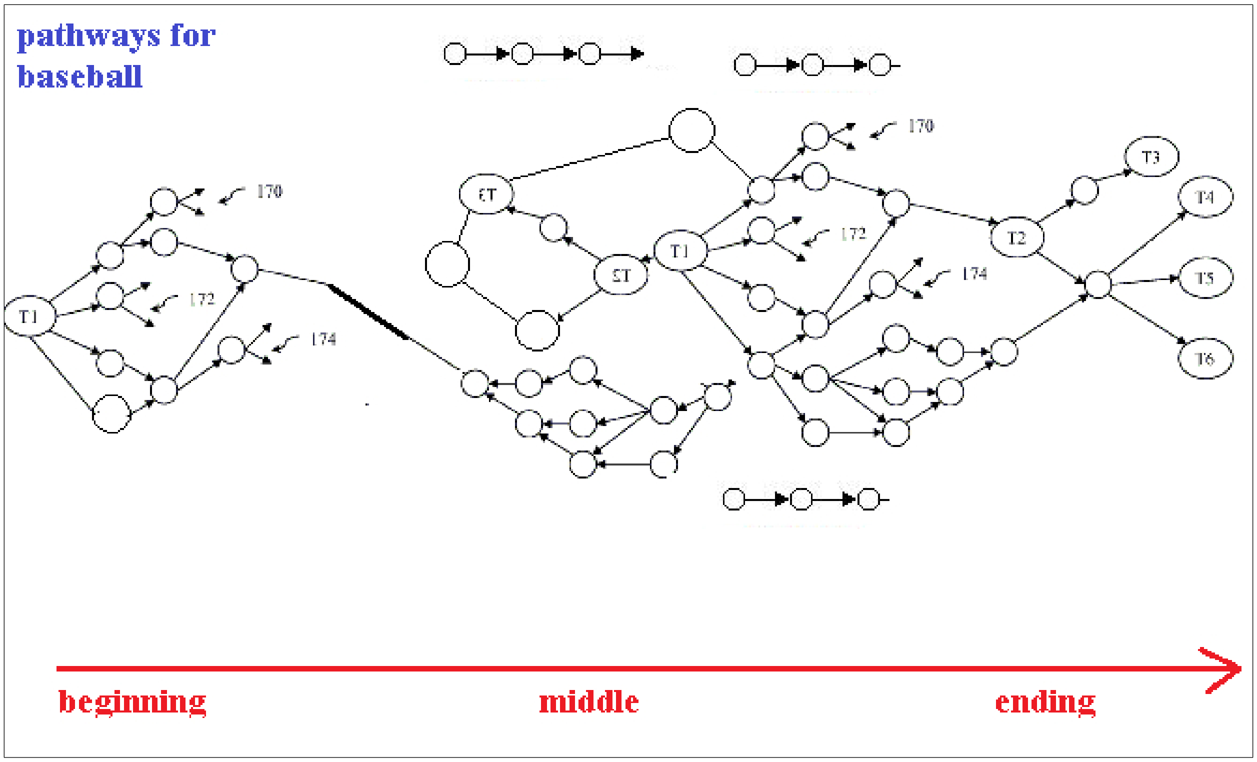
Also, English sentences stored in pathways basically define how data is stored, configured, and searched. The robot goes to school to learn knowledge and that knowledge define how data is stored, configured and searched in the robot's brain (refer to my books for details).

Pathways are clustered and stored in a timeline/s

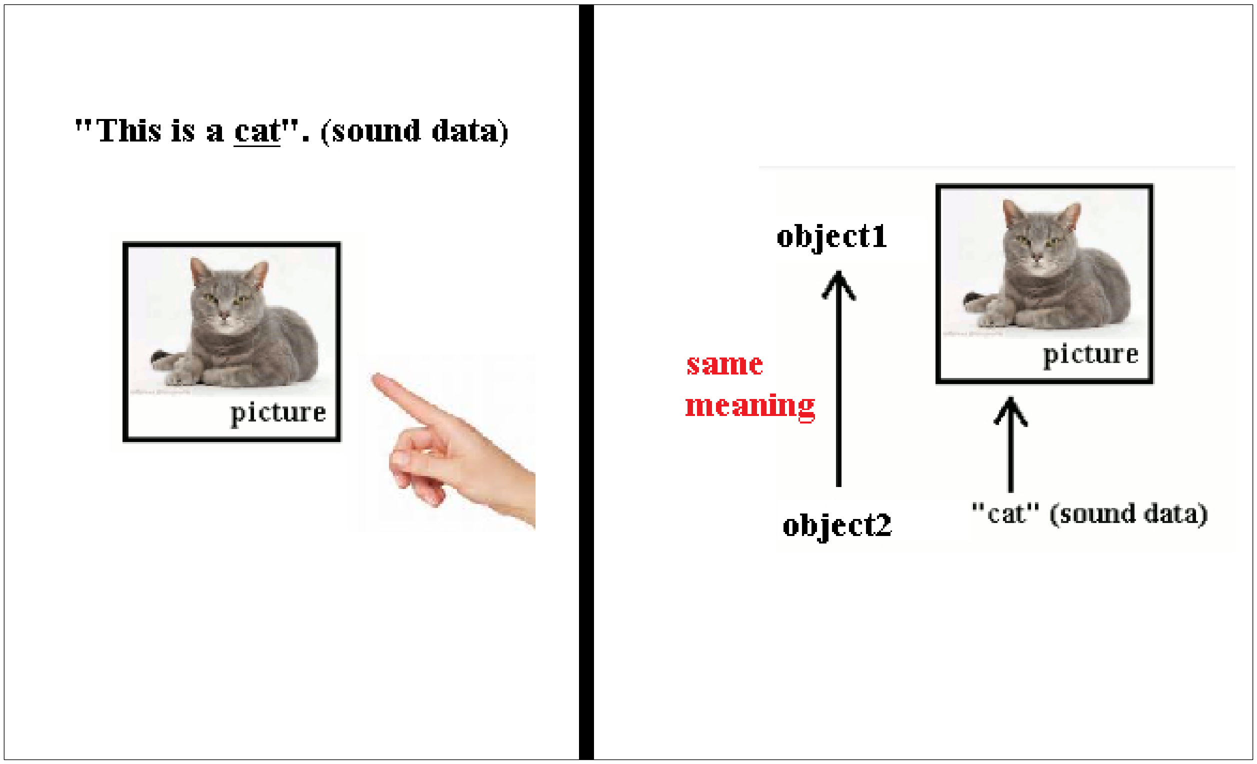
Another important thing to remember is that in a given task, like playing baseball, pathways are stored in clusters in a timeline. Pathways are stored at the beginning, middle, and ending of a game and depending on the time of the game and situation, the robot's brain extracts different pathways. For example, at the beginning of the game, the robot will extract pathways to follow instructions from a coach. At the beginning of a baseball game, the coach will give the batting lineup and player positions. In addition, there are some rules and objectives that are given, like the batting signals and how hard to play the game.

Another example is driving a car. Pathways for driving are also stored based on a timeline. At the beginning, middle, and ending of driving there are specific rules and procedures to follow. This is basically how my robot manages complex tasks. It uses the pathways as a way to separate individual procedures of a job. At the beginning you do this, at the ending you do this, in this situation you do that, if you're playing as the pitcher use these pathways, in multiple situations you do this, if you identify a street light you do that, etc.

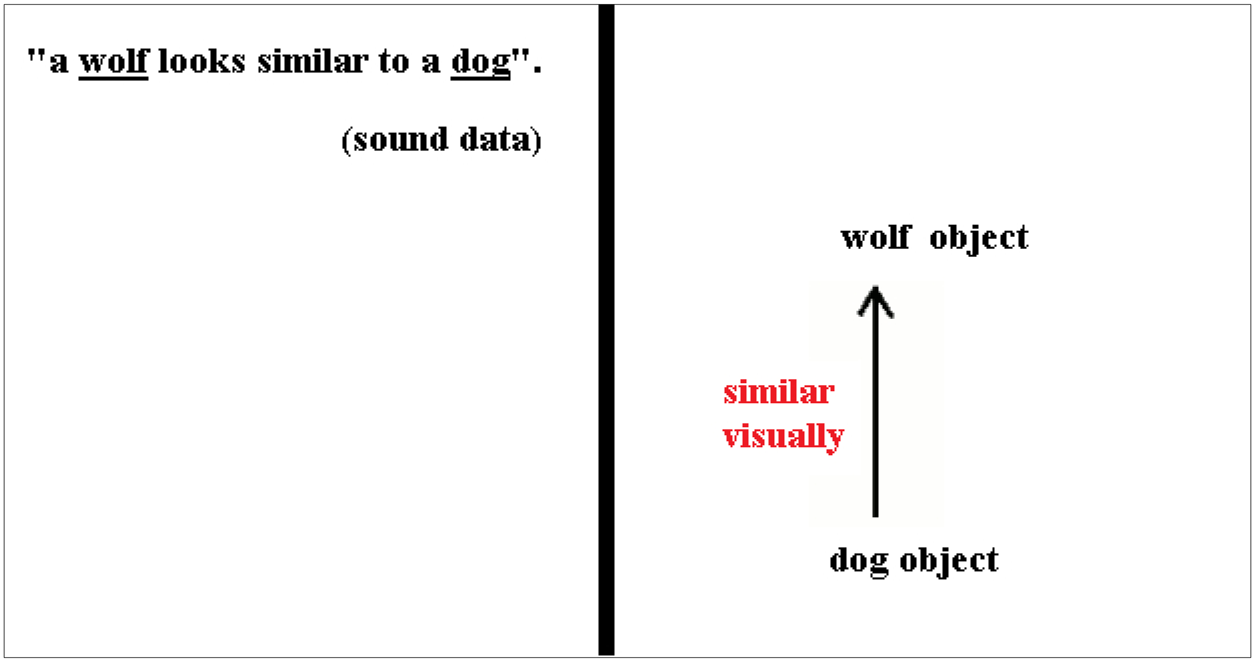
Every human task has different decision trees. And these decisions tree comprises massive forest and trees, connected or separated. These pathways can form any type of computer program or function to accomplish a task, regardless of how complex they may be. Writing software programs for Microsoft is a very complex human task and it spans all skills and knowledge. In some cases, the whole brain will light up because it is accessing information from all cognitive skills in order to do something complex like write software programs.



English sentences structure data into self-created semantic networks
A sentence like “this is a cat” has patterns. The pattern is, in memory, the location of the word cat and the visual image of a cat is located really close to each other. This sentence basically helps the robot to organize and create data in memory. If the robot has never encountered a dog before and the teacher points to a dog as says “this is a dog”. Based on the sentence the robot’s brain will create 2 objects in memory: the visual image of a dog and the sound data dog. These 2 objects will be created and stored close to each other in memory.
This is a ----



A sentence like “a wolf looks similar to a dog” also has patterns. The robot will find the patterns and attach them to the sentence structure: a R1 looks similar to a R2. In memory, the robot’s brain will establish relational links between the 2 objects: wolf and dog. The relation is that both objects are visually similar.



The point I'm trying to make is that English sentences generate self-created semantic networks in the robot's brain. Data are organized in complex data structures based on English sentences. Decision trees are structured into decision making semantic networks. Data inside the brain establishes relational links and operations like a semantic network; and search functions are discovered on its own to search for data in these semantic networks. Even things like logical networks, decision making systems, and induction and deduction reasoning are self-created.

Chapter 14

In my patents and books I talk about how the robot understands natural language through a method called the "fabricated movie". This fabricated movie is a very important component in human level artificial intelligence in terms of understanding common sense knowledge, and also understanding "universal language". There are about 5-6 claims in my patents describing this concept. 1. when the robot reads text it activates a fabricated movie. 2. said fabricated movie is the meaning to the words read. 3. said fabricated movie comprises 5 sense data from memory that is cut, copy and pasted together based on the words read. These are just some of the claims.

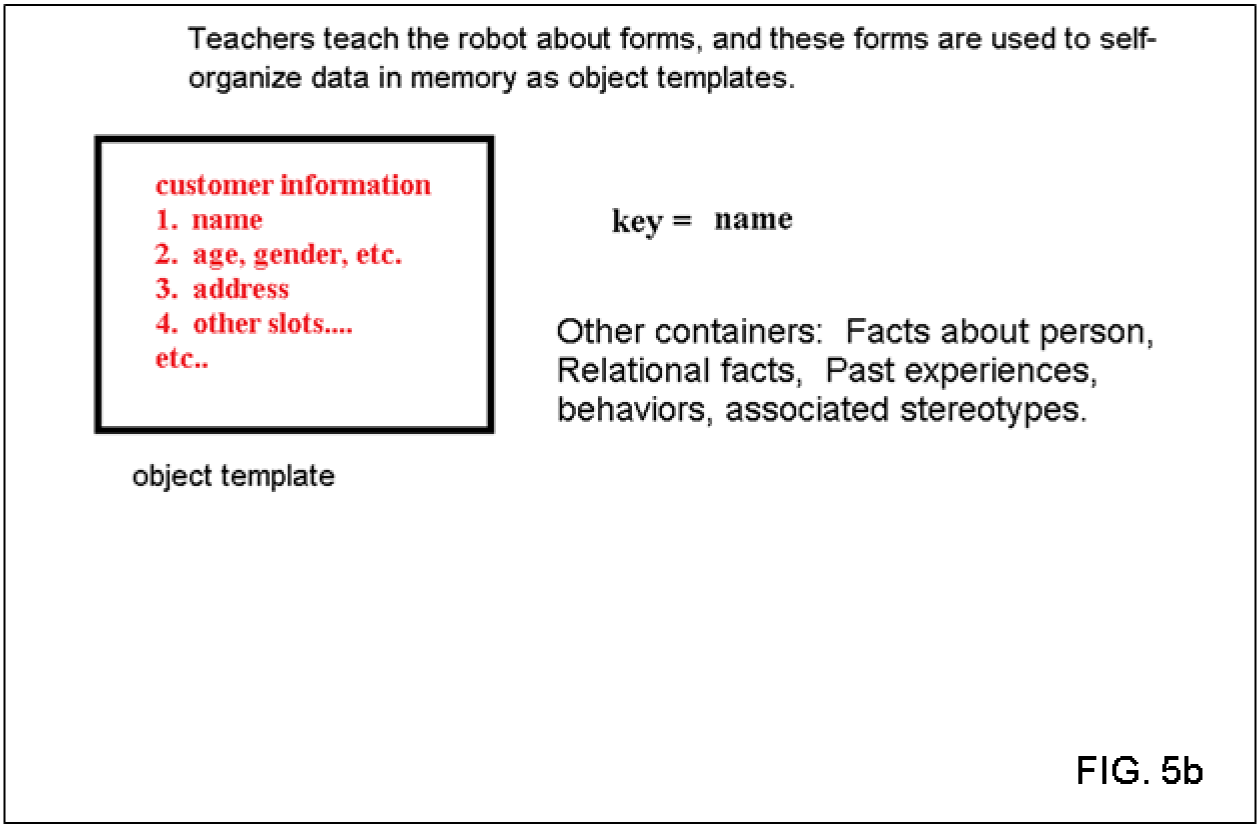
I go further about the subject and describe how the robot understands a book. When the robot reads a book it fabricates a mini movie in its head. For example, if the robot reads the Wizard of OZ for the first time, it creates a fabricated movie in its mind based on the words read. This is how the robot can answer questions about the story -- not based on the words but based on the sequential fabricated movie that was created, on its own volition, in its brain. It uses this fabricated movie to fast forward or rewind scenes to look for specific events or data in the story to answer questions.

The robot can read the Wizard of Oz from different languages, like Chinese, German, English, and Korean, but the fabricated movie is strikingly similar for different languages. This is the key component to NLP because the knowledge in memory are universalized – language can be represented universally in memory through primarily visual images – if the robot understands 5 different languages, these languages are stored in memory in separate areas and words and sentences of each language are all pointing to the same meaning – words and sentences for different language are referencing the same fabricated movie sequence.

Object templates and relational links

In terms of understanding a story, object templates or object composites are used. Teachers in school teach the robot how to understand an object. The lessons in school create what I call an object template of a person, place or thing. This object template neatly stores facts, relational data, and stereotypes of the object. All that knowledge of an object is organized and stored in a massive network of data, and the robot's brain is able to quickly and efficiently extract facts through these network nodes.

In one example , teachers teach the robot the important slots in a form. After repeated exposure to similar forms, these forms are used as data slots in this object template about a person or thing. The name, phone number and relationships are the most important data in an object template, and the brain mimics these forms as variables contained in object templates to store knowledge about people or places or things it encounters (referring to Fig. 5b).



Teachers teach the robot about forms, and these forms are used to self-organize data in memory as object templates.

customer information
1. name
2. age, gender, etc.
3. address
4. other slots....
etc..

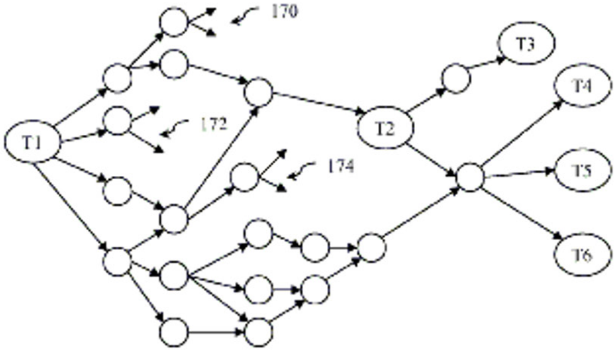
object template

key = name

Other containers: Facts about person, Relational facts, Past experiences, behaviors, associated stereotypes.

FIG. 5b

Self-created semantic network of objects (people, places and things) neatly stored in memory and hierarchically structured.

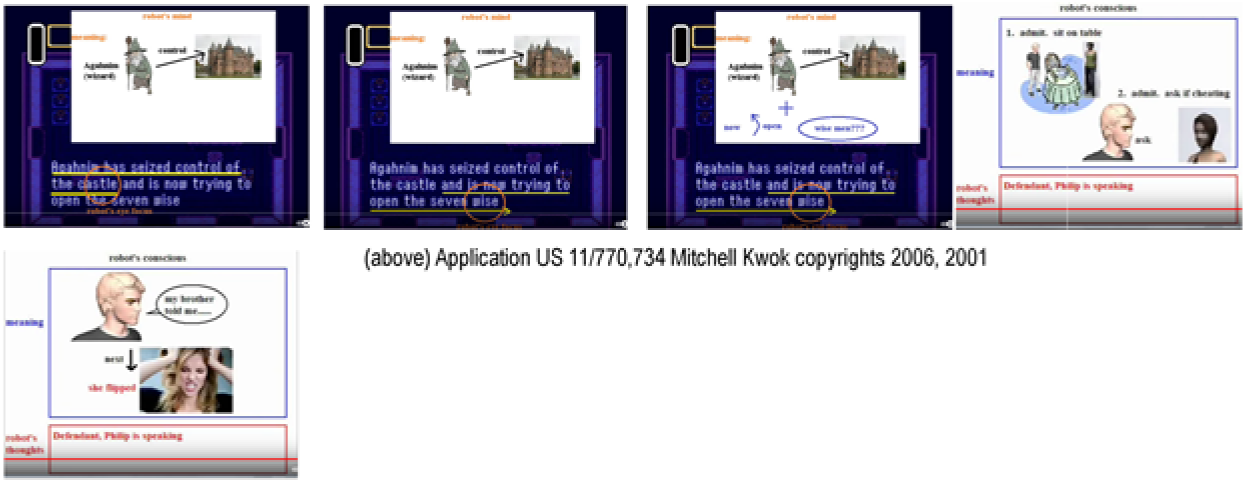


I call this thing object templates because knowledge and database information can be shared or newly created as "templates". For example, let's say that the robot was reading the Wizard of Oz again. He reads a 2 paragraph description of what Dorothy looks like. It stated that Dorothy has freckles and is a 16 years old girl. The robot automatically extracts an object template of a teenage girl from memory and uses it to represent Dorothy. It will further refine this character by fabricating what Dorothy looks like based on the text read about her. As the robot reads the story further it will store facts and scenes about Dorothy in this Dorothy object template – constantly reconfiguring and changing the network as it gets more information about the character.

By the end of the story the robot will have a very good understanding of Dorothy, and the database of information about her in the robot's brain is organized and contains comprehensive info about Dorothy, even information that wasn't described in the book. In fact, all the characters and scenes in the book will go through the same process. Events will be organized sequentially, places will have specific knowledge in terms of how it happened, what happened, when did it happen, what are the most important events or information to remember, etc. So each character in the story will have their own object template/s and they are organized based on the sequential story.

The visual mapping of the environment in the story is also contained in the brain. The robot will have the timeline and a visual 3-d map of where things happened and what was involved in the scenes, etc.. In the Wizard of Oz, the sequential locations are outlined in the fabricated movie. It starts of in Kansas and then it transport it to a new world in the land of Oz where Dorothy is now located at Munchin land. After following the yellow brick road the journey stops at the corn fields (the robot's brain will have a good 3-d map of this scene in its head based on text read), followed by the house in the forest. And then the night in the woods, then a pit stop at the Emerald city of Oz.

Reading (or listening) to text and activating a "fabricated" movie based on the text read.
Said Fabricated movie comprises cut/copy/paste images, text, and flow diagrams.



Agahnim has seized control of the castle and is now trying to open the seven wise

(above) Application US 11/770,734 Mitchell Kwok copyrights 2006, 2001

Object relationships are also established in the Wizard of Oz. Friends and family characters are different -- they each have different relationships to Dorothy. For example the witch is an enemy of Dorothy which carries many negative stereotypes and facts, while the scarecrow is a friend which also has many different stereotypes and facts. Aunt Emily is a relative to Dorothy and linked by blood. And in order to understand this kind of concept the robot has to borrow knowledge about family trees. Teachers teach the robot in elementary school what a family tree is and how relationships are established between people and its associated stereotypes. The robot must use this previously learnt knowledge to understand who Aunt Emily is in relations to Dorothy.

Why does Dorothy want to get back to Kansas so badly? Because she misses her family, specifically, she misses her Aunt Emily. Why does she miss Aunt Emily? Because Dorothy loves her aunt and she took care of Dorothy as a child. This is what is known in the field of computer science as common sense knowledge. All the knowledge learnt in school from K to college is needed to build this common sense knowledge. Understanding a person, place or thing, how to store important information about a person, how to organize data in an optimal network in memory are all knowledge learnt through teachers in school.

The robot is treating himself as an object template and it is building his own character by storing all aspects of itself into this massive object template. And then it's using this object template for other people or things read about in a book or experienced in real life.

This is where the identify part of the robot comes in. This object template of itself has a very comprehensive network of information about itself, and it was created through a lifetime worth of observation and experience. It knows what it likes or dislikes, what its attitudes are, its own behavior, how it makes decisions, what its beliefs are, and what its objectives are in life. This object template can also be proof that the robot is self-aware and is classified as a sentient being that can think and act based on the self (FIG. 10E).

How the robot stores data from its environment and what decisions to make are not predefined or hardcoded in its positronic brain by AI engineers. This is knowledge and skills learnt from teachers in school. How to store knowledge from lectures in its memory; what are the most important facts to remember are learnt from teachers in school. Teachers teach the robot what are the most important data to identify and remember in books or lectures. With these lessons, the robot can seek out important information of a book or lecture on its own volition. The self-created semantic network on knowledge about objects, events or actions in the robot's brain are learned through school. How to build a character composite of someone is learned in school, how to answer questions are learnt through school, how to follow grammar rules when writing or speaking is learnt in school, how to follow the law is learnt in school, how not to break the law and avoid jail time is learnt through school, how to make good decisions in life is learnt in school --- everything is learnt in school. The most critically important point is that none of these skills are preprogrammed into the robot's brain by software engineers.

In terms of NLP, the old traditional way is to preprogram grammar structures and rules into the system so that the AI can learn language. But, my method is better and easier. The robot reads books and experiences life to store sequential sentences in memory, and the averaging out of these sentences generate a sort of self-created grammar rules or grammar structures – a simple type of grammar rule system without predefined instructions. With lots of exposure to the

English language and access to vast repository of books and texts over the internet, the robot's brain will eventually discover patterns to common grammar rules on its own. However, in order to write good articles, the robot still has to go to school to learn proper grammar rules and "ethics" and apply them to writing. The only way to do that is for the robot to go to school and take grammar classes to learn English (or other languages) the proper way. And grammar has to be learned in grade levels, without skipping.

This not only instills common sense into the robot, which have been elusive in computer science, but this common sense knowledge can be used for logical reasoning and decision making. Again, all of that is also learnt in school from kindergarten to college. The robot learns information in terms of a bootstrapping process whereby data in memory are built on top of each other recursively and hierarchically. In order to become a computer scientist the robot has to first go to school all the way up to high school. Next, it has to pass the SAT in order to go to college. Once it gets to college it has to take linear courses to acquire the necessary knowledge to become a computer scientist. Here is a diagram to show the learning process to become a computer scientist (FIG. 7A). First, the robot has to finish basic arithmetic in elementary school and middle school. Next, in high school it takes algebra, then trig, next Calculus. Finally, when the robot gets to college, take courses like discrete math, computer programming, advance algorithms, integrated circuits, operating systems and software engineering (all the prerequisites of CS). Thus, information is built on top of each other recursively whereby old data is built on top of new data to form complex intelligence. Of course, it's not just about math, other courses that have no relation to math is also required like art and history.

By the way, I described deep learning 6 years before deep learning 2012 came out. The scientists who pioneered deep learning said they started on deep learning in 2007. Mitchell Kwok published his book in 2006. Coincidence???? I think not.

I called deep learning an "encapsulated tree" or "hierarchical tree" back in 2006. It is the smallest unit for representing 5 sense data in memory. When data is stored in memory, it is represented by an encapsulated tree compiled into prioritized object layers and represented in a tree like structure. Then it is stored sequentially in a massive 3-d network of trees and forests -- connected or separated (this 3-d network isn't a traditional neural network with 3 layers and back-propagation). It self-organizes data based on association using primarily 3 factors: commonality groups learnt groups and time seq. 3-d sequential images (called pathways) are stored in memory sequentially as they occur to learn information. These pathways are averaged out based on similar examples and universalized to create computer programs to make decisions. Pathways are built on top of each other recursively to form complex thinking for the robot.

If all this sounds familiar it's because this is the latest 2022 AI research. All technology companies are converging on this idea. It's not new at all, it's not novel, and it's not non-obvious it is "prior art". I filed these ideas back in 2006 as either patent applications or registered copyrighted books. So Mitchell Kwok owns the copyrights to the descriptions, diagrams and examples used. Unfortunately earlier notes on these ideas were not published and considered private papers stored in my Unix account starting in 2001. I started on this when I was in college and shared it with students and professors in presentations or required course papers. The sharing of information by me can be the bases for the original copyrights to these ideas, which was conceived starting in 2001. The encapsulated tree (later known as deep learning) was the first idea I had.

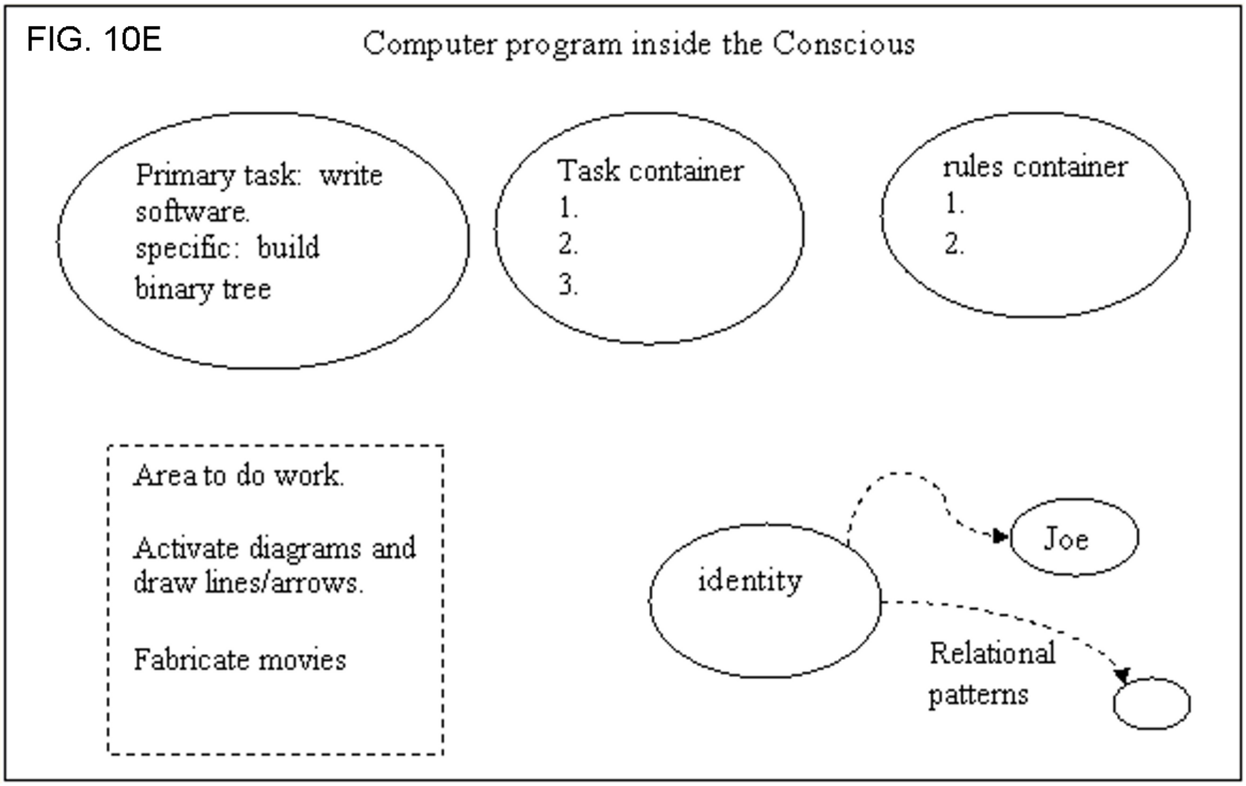
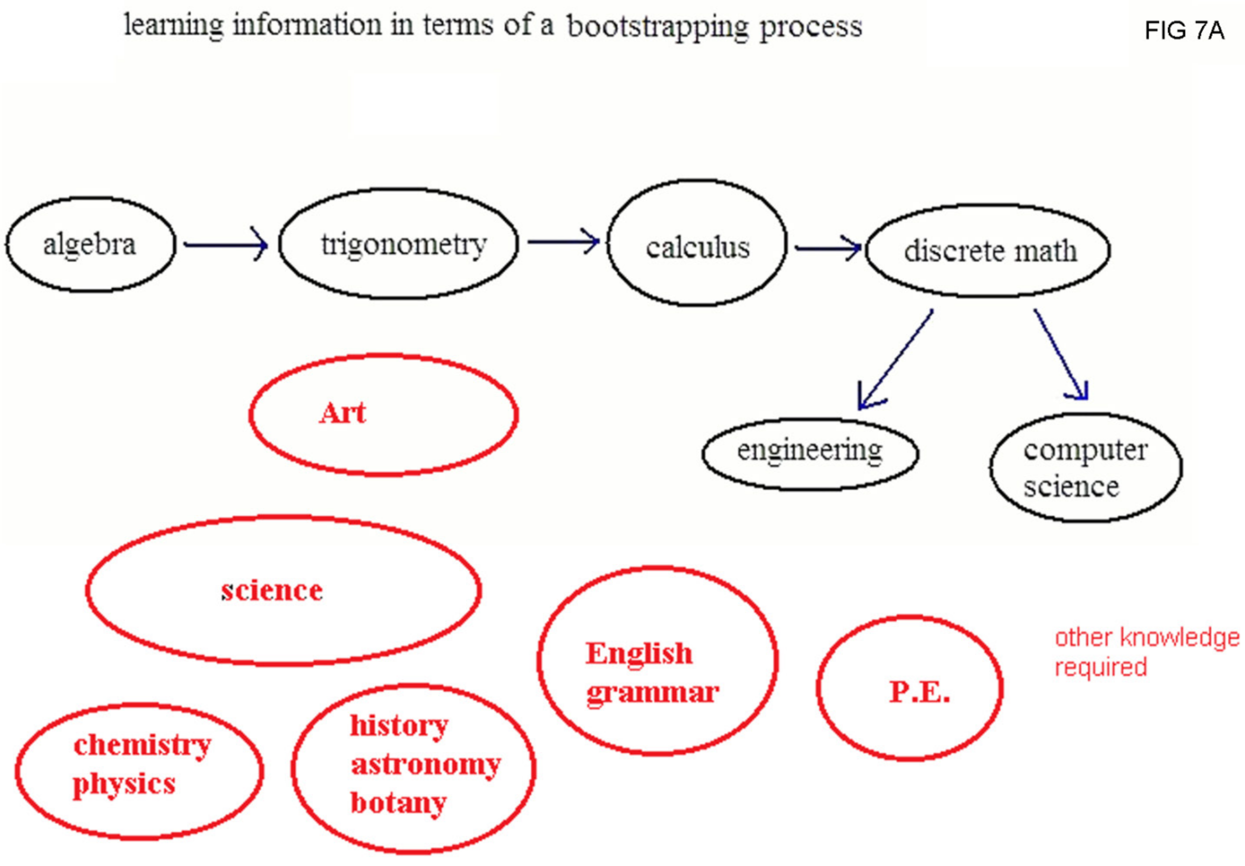


FIG. 10E

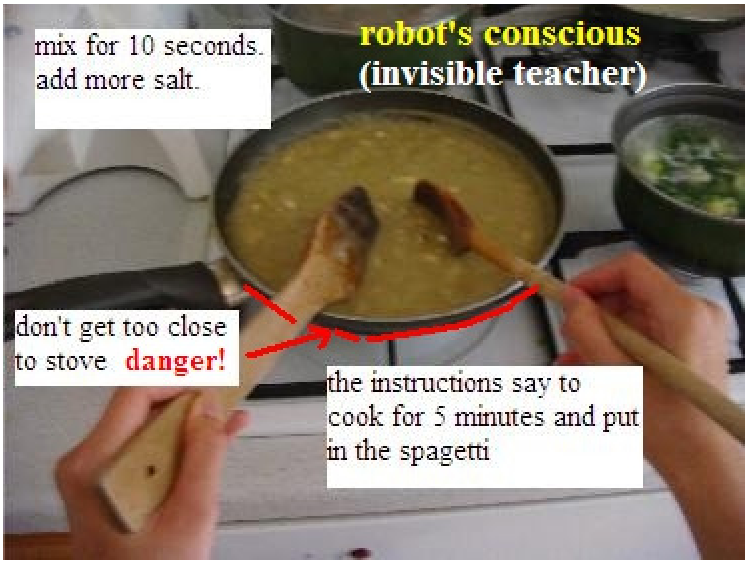
-- establishing identify and relational links to other people or characters in books.


learning information in terms of a bootstrapping process
FIG 7A
algebra
trigonometry
calculus
discrete math
engineering
computer science
Art
science
English grammar
P.E.
other knowledge required
chemistry physics
history astronomy botany

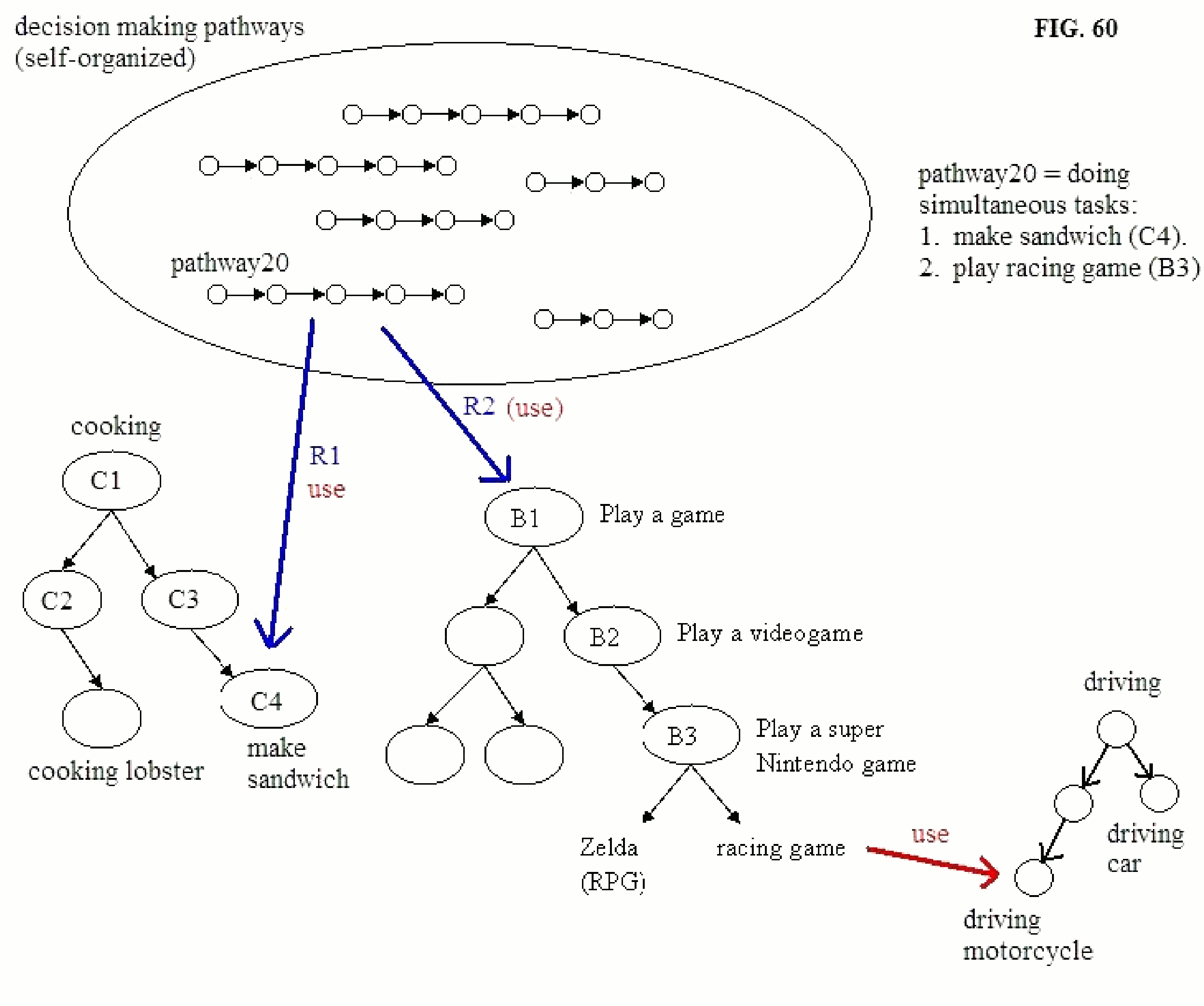
Chapter 15

Self-awareness (part 1)

Self-awareness means a human or robot that can control their actions, both mental thought and physical action, in an intelligent manner (with college level intelligence). The primary function of the robot is to pursue actions that will lead to pleasure and stay away from actions that lead to pain. Thus, the robot is self-aware because he is able to manage tasks. This includes: doing a task, doing multiple tasks, doing simultaneous tasks, solving interruptions of tasks, aborting tasks, modifying tasks, scheduling tasks, following rules while doing tasks, etc.



Basically the robot is alive and self-aware as a result of the robot's conscious. The robot's conscious manages tasks. In addition, the robot's conscious can also do other things like observe the environment for danger or unusual events, alert the robot to hazards from the environment, filter out sensed data from the environment, generate facts about objects, provide common sense knowledge, solve problems, extract facts from memory, etc. Referring to FIG 60, all tasks are structured hierarchically. These task range from driving a car to playing video games to making a sandwich. Similar experiences will be stored closer to each other. For example, driving a car is similar to driving a motorcycle so they are hierarchically structured and stored close to each other.



Also, data in the robot's brain are stored and gravitate close to each other based on association. In FIG 60, the playing racing video game pathway will be stored close to driving a real motorcycle because of their similarities. Data like cat and dog will gravitate towards each other in memory because of association.

When all the task pathways self-organize, based on hierarchical similarities and association, universal decision making pathways are created, which form the top of the hierarchy (the robot's brain comprises massive forests and trees). These decision making pathways control and manage tasks for the robot. In other words, the robot is self-aware because of these decision making pathways.

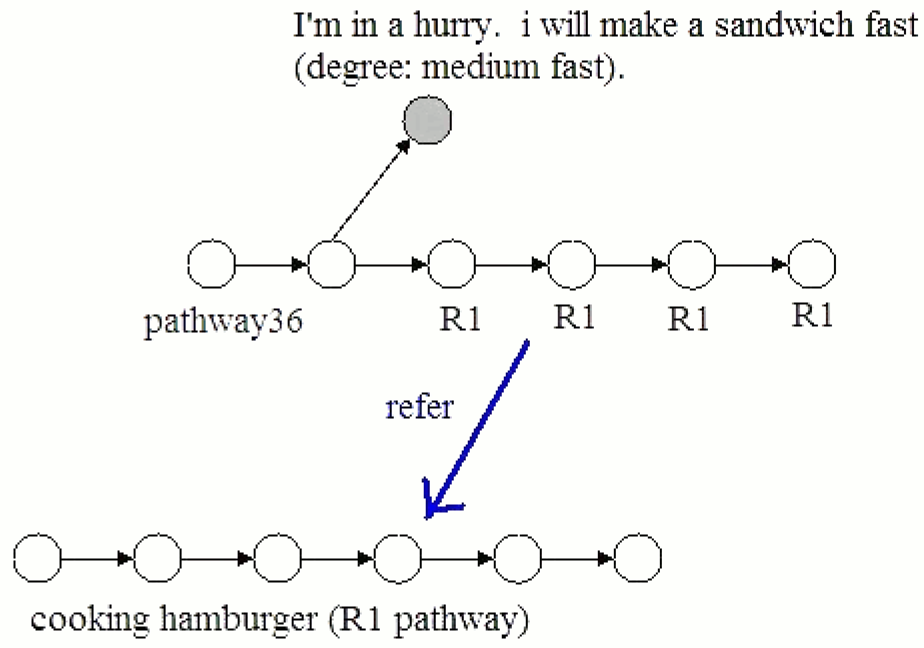
The most important thing is that the decision making pathways are created by lessons learned in school and through personal experience. All decision making pathways are learned from either teachers in school or information in books. As stated numerous times in previous books, my robot's brain does not use machine learning, neural networks, NLP, decision trees, genetic programming, semantic network, predicate calculus or any modern day AI. However, it does use deep learning to break up data in movie sequences (called pathways) so that data can be stored in an optimal manner. Only 10 percent of the robot's data structure is devoted to deep learning. Despite what people say I, in part, discovered deep learning back in 2006 (some of the material in this book were discovered even earlier than that via college papers).

Let's take a look at what kind of pathways are stored in the decision making pathways. If you look at MIT and Stanford University's research papers and books, they focus on mainly recursive tasks. In my invention, teachers teach the robot how to do recursive tasks. Also, storing linear steps in pathways is one way to represent recursive tasks. If the robot is making a sandwich and is doing step1, step2, step3, etc., he is automatically doing recursive tasks. More complex type of recursive tasks are learned from teachers. For example, the teacher can teach the robot to manage complex recursive tasks by writing things down. If the robot has to do a task that has 200 recursive sub-tasks he can write things down on a piece of paper. He can mark off completed tasks and see which tasks to do next, or which tasks to abort, or modify, etc. For simple recursive tasks, the robot can use his brain to remind himself what tasks to do.

Referring to FIG 60, pathway20 is doing simultaneous tasks. Teachers teach the robot to do task1, stop, do task2, stop, continue task1, stop, continue task2, stop, continue task1, etc. The robot has to repeat this until both task1 and task2 are completed.

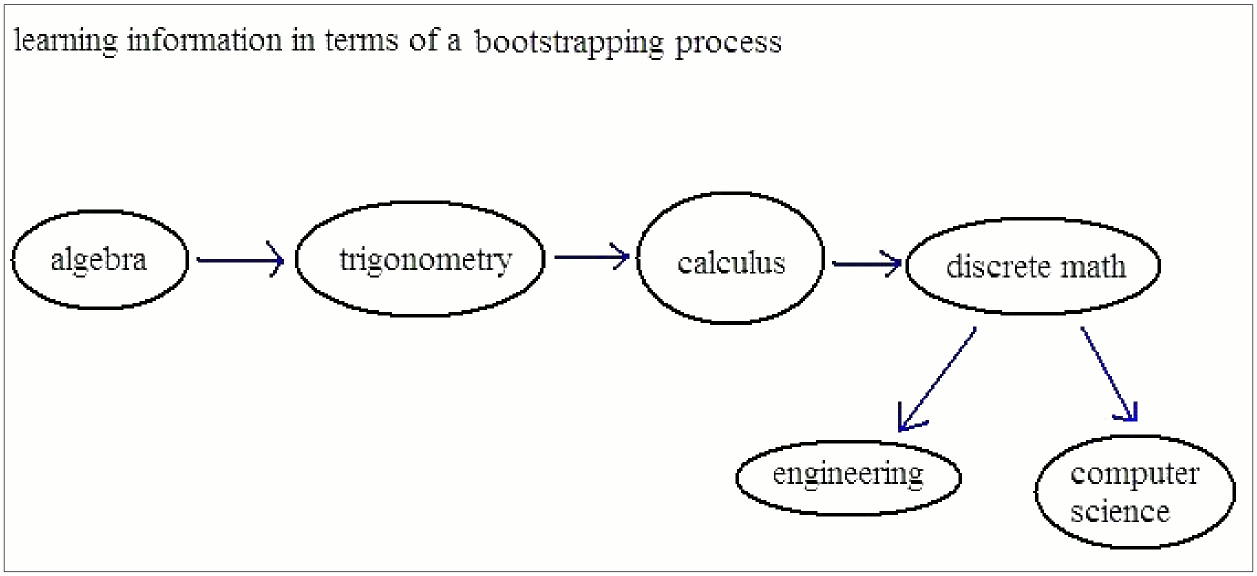
Referring to FIG 62, pathway36 is doing a task quickly or at a speed. Teachers teach the robot how to speed up or slow down a task. Through repeated experience, the robot creates a universal pathway in memory that can do any given task quickly. Teachers also teach about the consequences of speeding up a task. For example, if the sandwich is done too quickly, the robot might make mistakes like drop the lettuce or adding too much tomato sauce. Only through personal experience and learning a wide variety of task to speed up can the robot truly refine his skills.

FIG. 62



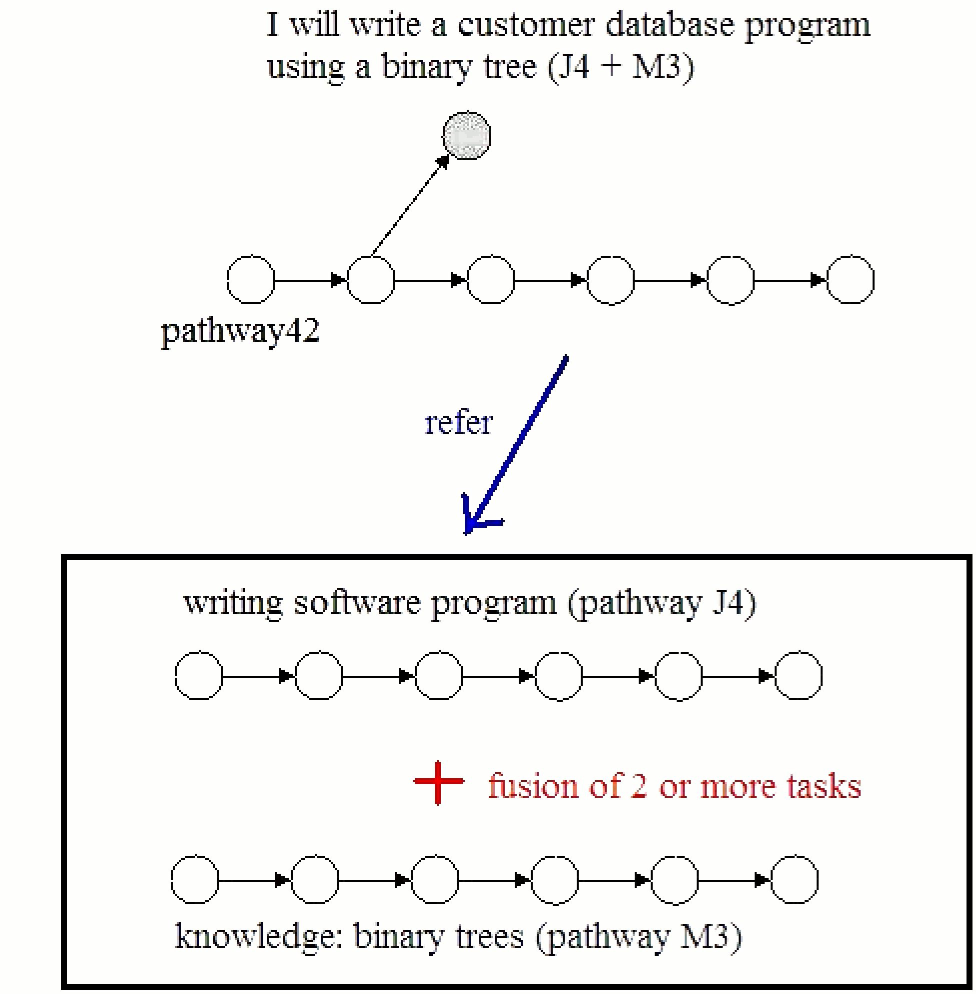
Referring to FIG 64, pathway42 is a very important pathway because the robot is trying to combine knowledge from 2 or more tasks (pathways). Scientists from MIT and Stanford University have been trying to teach their robots to learn information in a bootstrapping manner. If the robot learns simple math like addition, how can he use that knowledge to solve a problem? In another example, if the robot learns what a binary tree is, how can he use that knowledge to write a customer database system?

Humans learn math through a bootstrapping manner, whereby information builds on top of each other to form complex intelligence. First we learn algebra, then we take that knowledge to learn trigonometry. Next, we take trigonometry to learn calculus. Then, we take calculus to learn discrete math or computer science.



Referring to FIG 64, pathway42 is taking 2 tasks and combining them. A simple example would be: task1 is to manage bullets in call of duty. task2: do addition. if the first task is to manage bullets in call of duty, then the robot has to use bullets as variables when doing the addition problem. Let's say the robot (the player) picks up 2 gun clips from a house, he obviously has to keep track of the amount of bullets he has in his gun so he must do simple math, mainly addition. The robot currently has 40 bullets in his gun and each gun clip has 15 bullets. Since he has 2 gun clips the equation will look like this: 40 +15 +15 = 70. Thus, currently the robot has 70 bullets in his gun.

FIG. 64



In this case, bullets is a variable, and it is used to plug into the addition problem. This is how the robot takes knowledge like addition and apply that knowledge to a given problem. Pathway42 showcase a more complex example. Task1 is to write a customer database system and task2 is to use knowledge on binary trees. Now, the robot has to take knowledge learned about binary trees, which is an algorithm, and apply it to build a customer database system. Obviously the robot has to identify variables and functions from the customer database system and insert them into a binary tree. For example, the nodes in a binary tree will represent customer information and the search operation will be A-M is used for the left search and N-Z is used for the right search. Thus, learning information in terms of a bootstrapping process is actually a learned thing. It is learned from teachers in school. This is important because some AI scientists think genetic programming or decision trees is the key to this type of self-learning. It isn't.

In conclusion the human brain isn't special. All intelligent thinking from a human brain comes from knowledge learned in school and there is no divine angel that is giving us information. The voice in a human's mind is like an invisible teacher that: gives information, make decisions, alert the host for danger, observe, generate common sense knowledge, predict the future, schedule tasks, etc. This invisible teacher, which exists in a human mind, was created from a lifetime worth of learning from school, through personal experience, and knowledge from books.

My goal for this book is to explain how the human brain works and how to build a digital brain comparable to human level intelligence. In my opinion I have succeeded; and I'm 100 percent sure my invention filed back in 2006 is the correct AI software. Content on this page explains in general how my invention works and what it does. The actual software is really really long and complicated. Refer to my books for detailed information and examples.

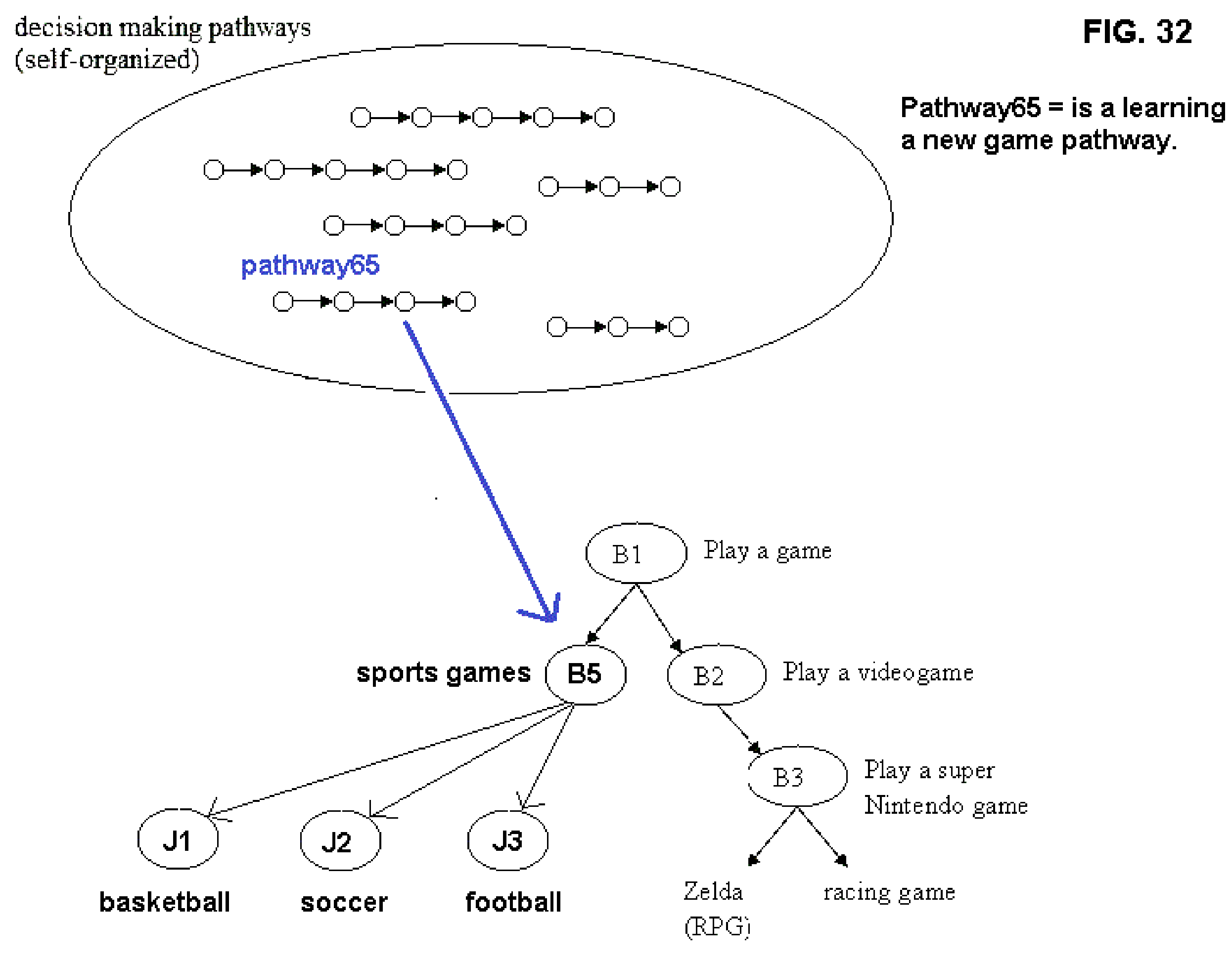
Self-awareness (part 2)

As stated above, decision making pathways manage tasks for the robot. However, there are many other things these decision making pathways can do, such as:
1. learning a task (learning a new skill like driving a car or cooking a lobster dish)
2. practicing a task.
3. usage of a task/s
4. modification of a task.

1. For the first one, the decision making pathways can learn a new task (or skill), such as playing poker or making a sandwich. It can also learn complex or abstract knowledge like binary trees or linked lists.
Teachers in school teach the robot "how to teach itself" to learn a new skill. If the robot knows nothing about soccer, he has to seek out info in books or PE lectures to learn the objectives and rules of soccer. He was taught by teachers to identify goals and procedures so he will know what to do at any given time (called procedures). Then, he has to identify the rules of the game so regulations are followed while playing soccer. This can be had by reading books or observing demonstrations, like a teachers' lecture.

Referring to FIG. 32, after being taught 3 different sports game by a teacher, the robot develops a universal pathway to learn a game, called pathway65. Pathway65 was created because the robot was taught how to play soccer, football, and basketball. In all 3 sports, the instructions to learn are the same or similar. All three sports have these same or similar instructions. Because the learn instructions were so similar, the 3 pathways (J1, J2, and J3) self-organized and created pathway65 as a result.

FIG. 32



Now, pathway65 is known as the learning a game pathway and is considered a decision making pathway. Any new game that the robot doesn't know, his brain will use pathway65 to learn. For example, if the robot doesn't know how to cook a lobster dish, his brain will extract pathway65 to learn this new skill.

2. To be able to understand the goals and rules of a game is one thing, but actually playing the game is something completely different. So, practicing a skill is very important because it allows the robot to refine its skills and become an expert in it. FIG 21A is a diagram depicting a learn a task pathway and FIG. 21B is a diagram depicting a practice pathway (which is also a decision

making pathway). The learn a task pathway id and stores the knowledge of a task; and the practice pathway id and stores the linear instructions in pathways.

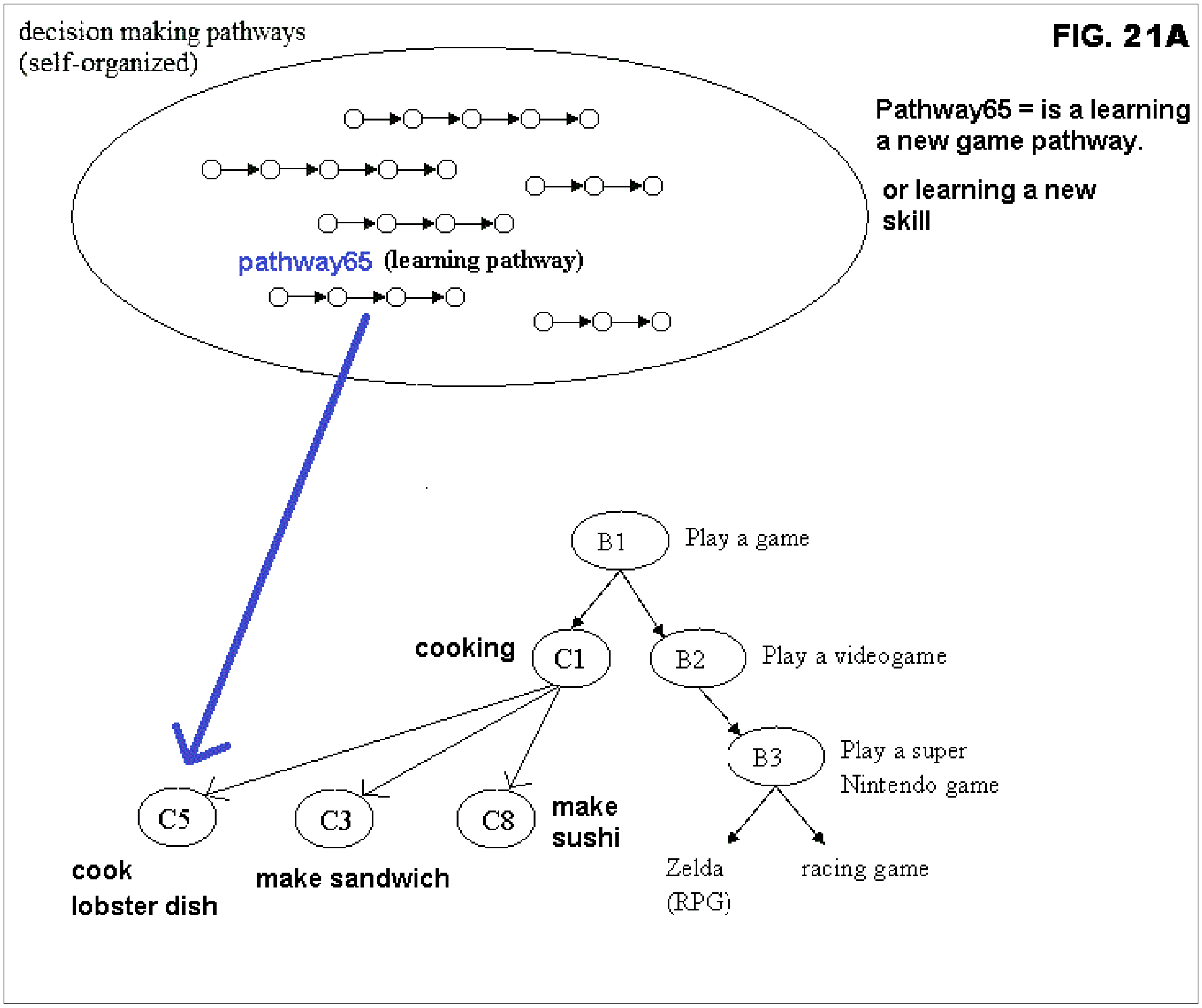
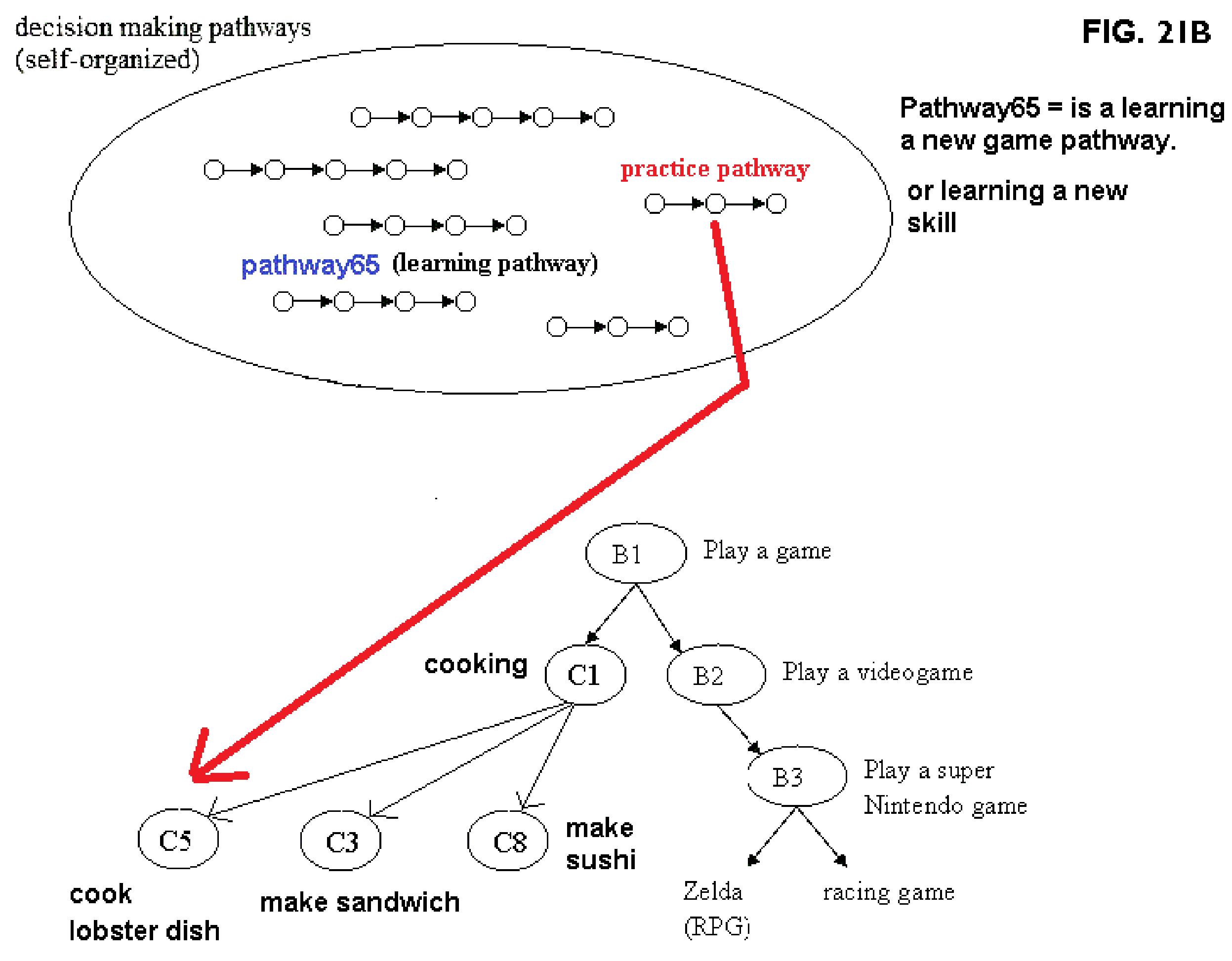


FIG. 21A

Pathway65 = is a learning a new game pathway.

or learning a new skill



In this example, FIG. 21B, the robot is cooking a lobster dish. He is following and trying to remember the steps to accomplish the task. In some cases, the pathway has a for-loop at the end and it will loop itself until a condition is met. The condition is to commit to memory the entire steps of cooking a lobster dish. After several tries, the robot becomes an expert in the task and within the cook lobster dish pathway are the exact steps and rules to make a lobster dish. The more the robot practices the better it becomes.

3. Referring to FIG. 21A, the important thing is task pathways are different from decision making pathways. For example, the decision making pathway to learn a skill (pathway65) is different from the making lobster dish pathway (C5). Pathway65 is there to make the robot learn a skill. The end result is pathway C5, which contains all the knowledge needed to make a lobster dish, which includes: goals, rules, linear steps, safety rules, basic cooking knowledge, etc.

Now that pathway C5 is created in memory, another decision making pathway is used to "utilize" C5 task.

Referring to FIG 21C, pathway20 is a decision making pathway to do 2 simultaneous tasks at the same time. Pathway20 is referencing data from pathway C5, to make a lobster dish. The robot is actually making a lobster dish and talking on the phone at the same time and trying to balance between 2 tasks.

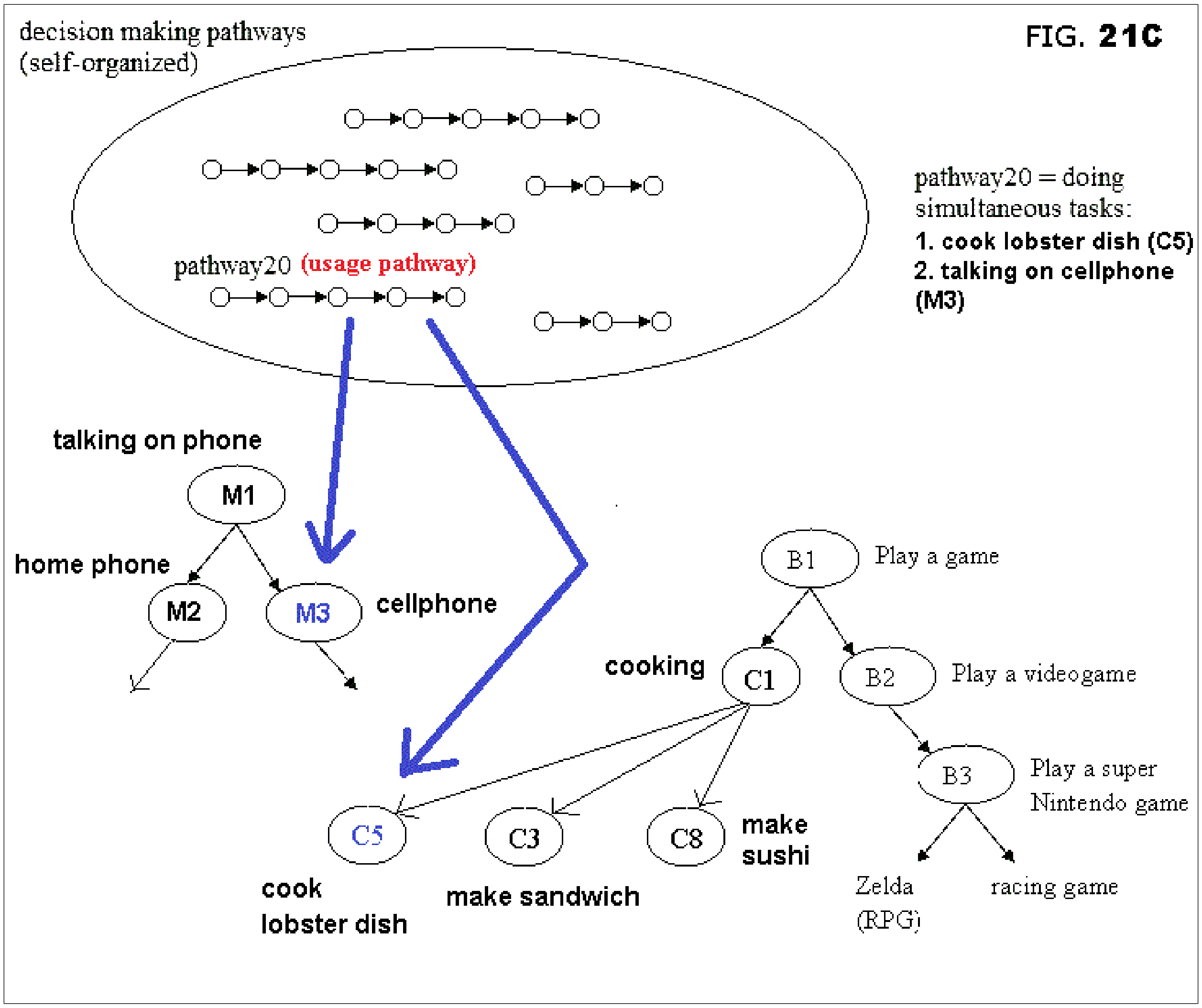
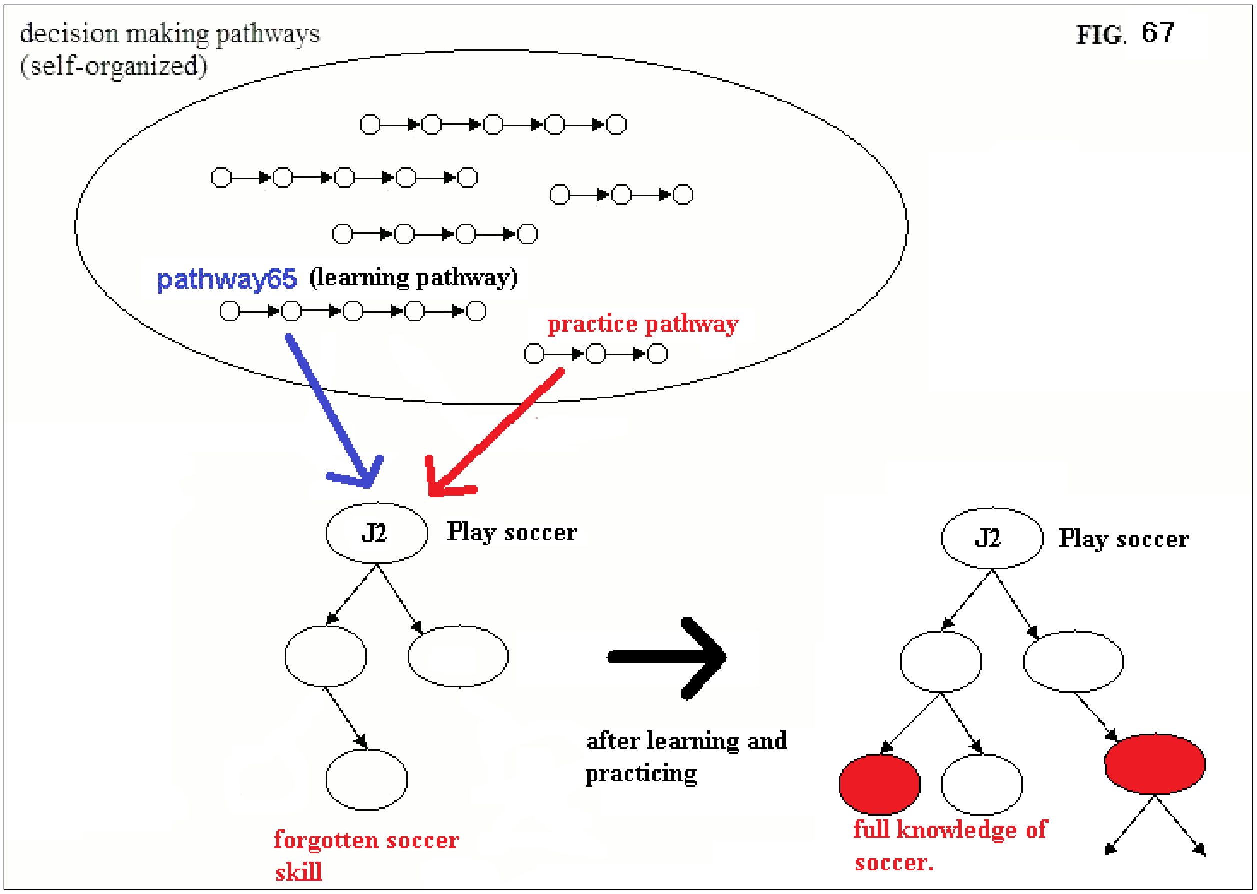


FIG. 21C

4. Let's say that the robot learned how to play soccer in PE from school. 10 years later, that knowledge is forgotten and only vague facts are still stored in memory. In order to gain the full knowledge back about soccer, the robot's brain has to use the learn pathway and the practice pathway again to remember and relearn the game of soccer.

In FIG 67, as you can clearly see after using the learned pathway and the practice pathway, the soccer knowledge is restored back to full capacity. The robot forget information in order not to overload its memory bank. This ensures that only important things are stored in memory and minor things are forgotten. Also, forgetting allows the robot to live for millions or billions of years without needing additional harddrives for its memory bank.

FIG. 67



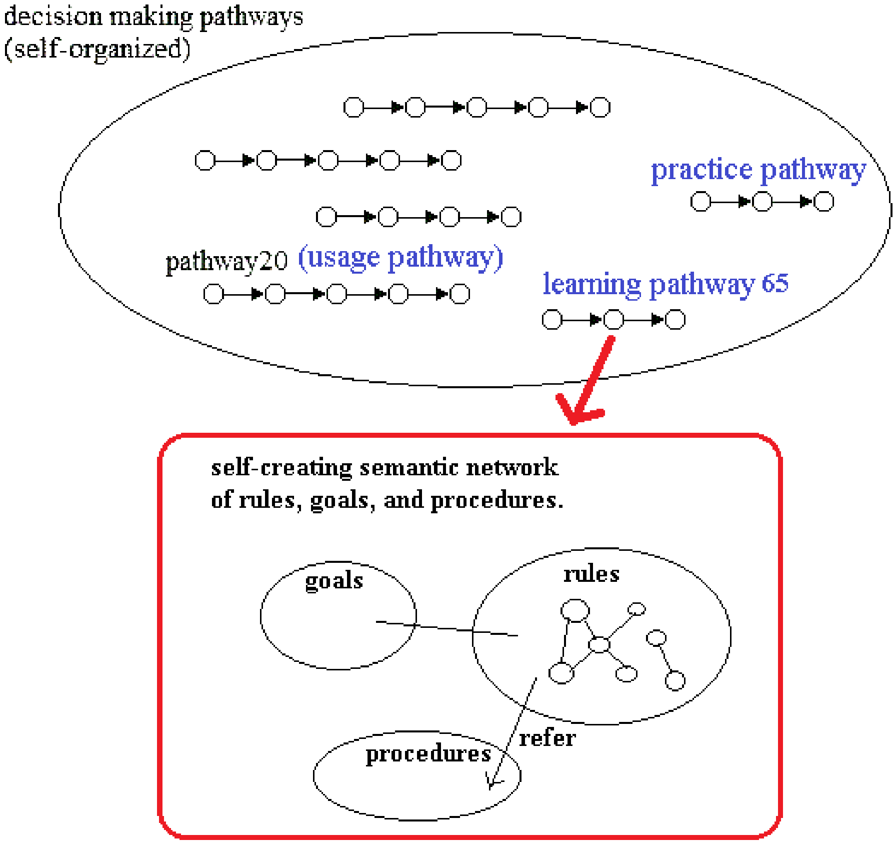
Details on what is stored in the hierarchical trees (self-creating computer programs)

There are 2 things that are stored in pathways, structured in hierarchical trees:
1. computer functions and operations
2. time estimation for events and probability and possibility of events occurring.

1. Self-creating computer programs stored in pathways.
School teachers teach the robot "to teach itself how to learn and play" a game. In FIG. 45A the sentences in pathway43 is telling the robot's brain to create 3 containers, one for goals, another one for linear procedures and the third one for rules. The robot learns these goals/procedures and rules and put them in their respective containers.
"in all games there are goals, procedures, and rules. you have to identify all 3 in order to play a game. understand these goals and rules. Usually, during the game you have to do linear procedures and at the same time follow rules. your goal is to win the game."

These sentences configure data in an organized manner. In some ways, it's generating a self-creating semantic network of data of various data types (based on human 5 senses). Not only are the data semantically organized in an optimal way, but search functions to find information is also created. It's like a self-creating search engine.

FIG. 45A



Further more, the sentences tell the robot what to do during the game (FIG. 45B). in this case, the instructions act as a self-creating operating system: to do procedures, and at the same time, follow rules. The goal is to win the game.

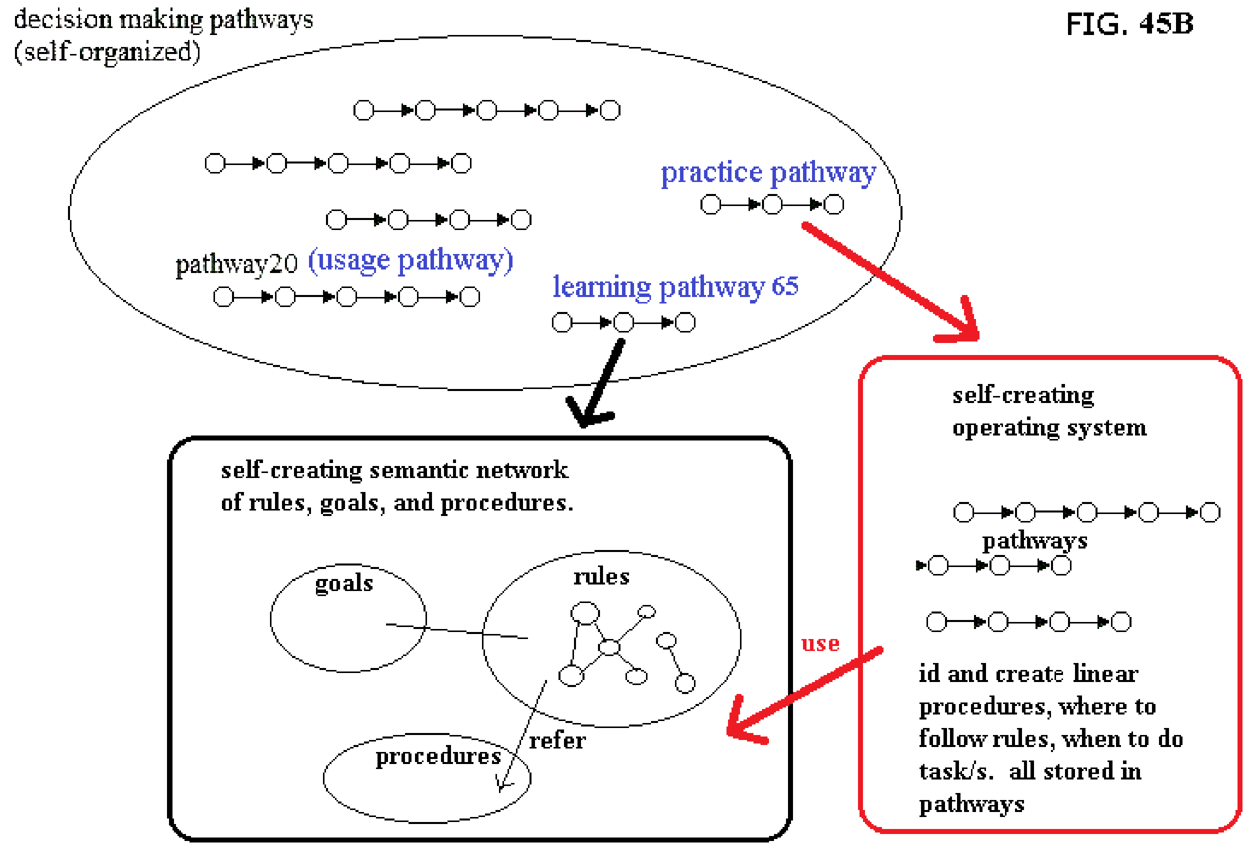


FIG. 45B

All this is done through English sentences. For example, the sentences below:
"the goal of chess is to take turns playing the game until someone's king is eliminated. while playing the game you have to follow all the rules. here are the rules. remember them and follow each one."

is telling the robot everything he needs to know about the game of Chess. These sentences are spoken by a teacher and the responsibility of the robot is to identify key facts. Once key facts are identified, the robot will use common sense to determine what to do in the game. For example, the procedure of the game is to take turns with an opponent and the goal is to eliminate the opponents' king. Thus, these instruction sentences given to the robot creates objectives and procedures in his mind.

Thus, the next time the robot is playing chess, facts will start to pop up in his mind, such as:

1. the procedure of the game is to take turns with opponent.
2. the goal of chess is to eliminate your opponents' king.

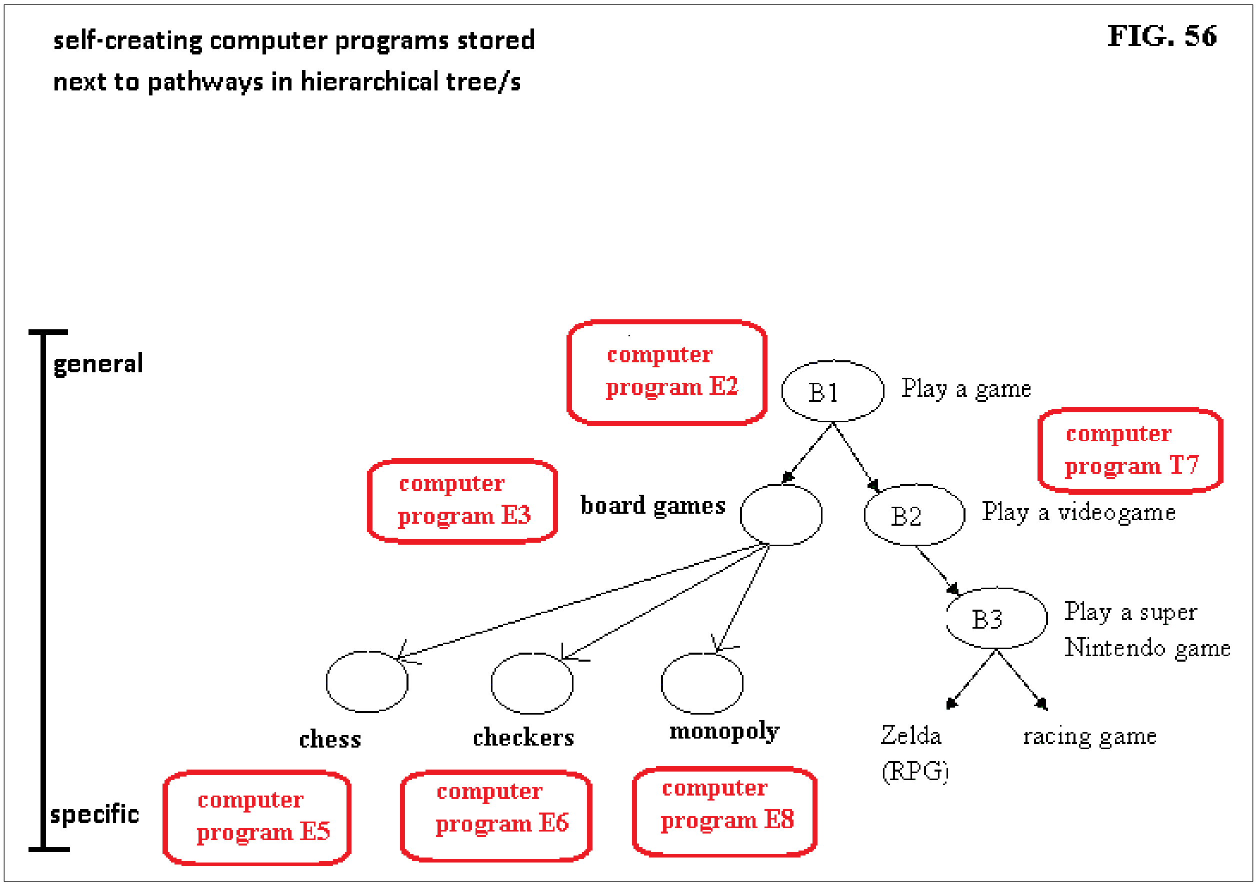
3. here are the rules (via visual images) for each chess piece. follow these rules when moving each chess piece. the horse piece can only move in a L-shape direction, the king can only move one step in any direction, etc.
4. predict the future by imagining future steps for me and opponent and come up with good moves.
5. make your moves quickly, this includes predicting future moves for me and opponent and selecting the best move to win the game.

These sentences create a computer program inside the robot's mind to understand goals, procedures, rules, solving conflicts of rules, searching for data in memory regarding chess, etc. These sentences set up the semantic network to store and retrieve data, set up the linear steps to play chess, the goals and rules, and so forth.

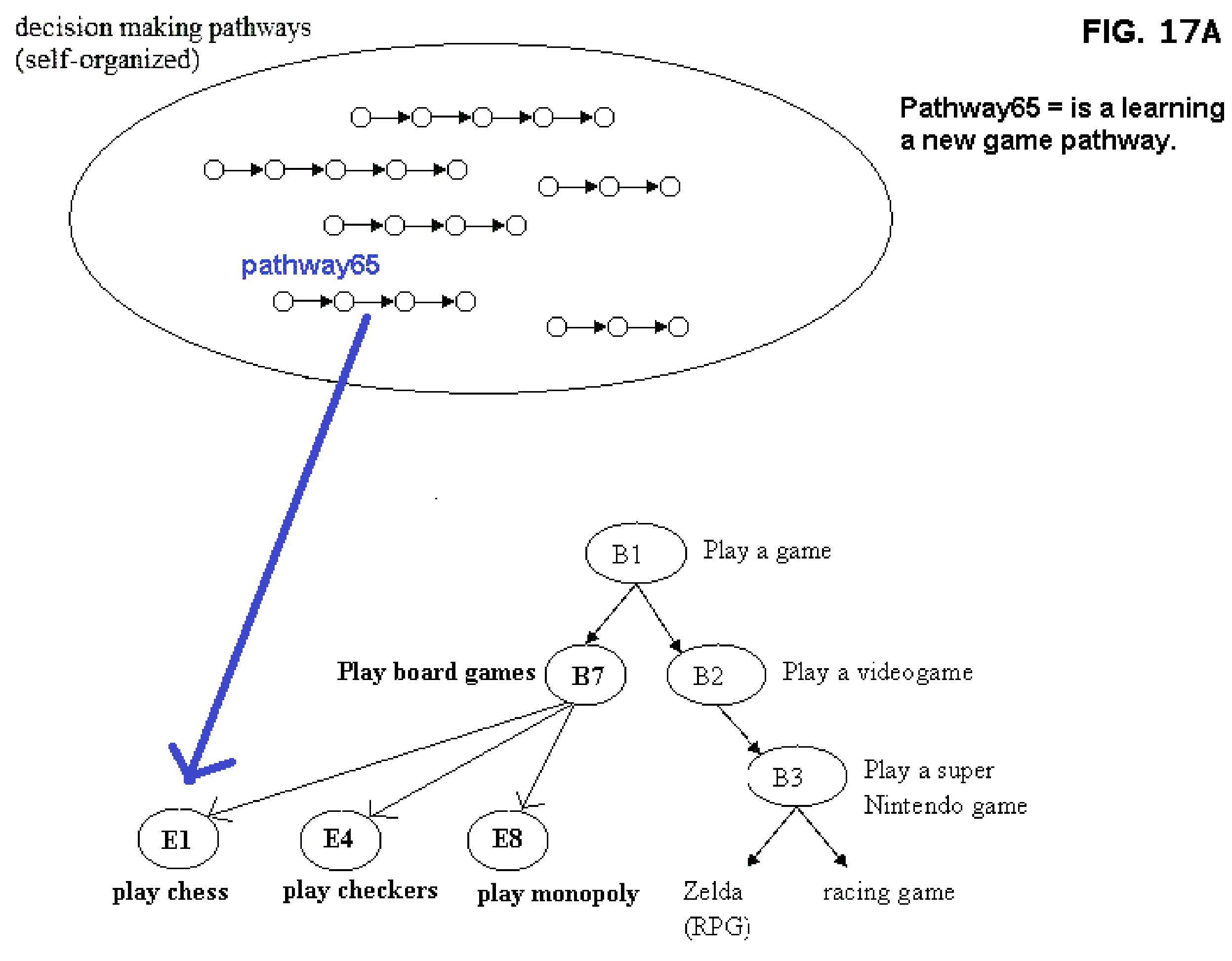
In the game of chess, there're many previously learned skills. For example, the sentence: "the procedure of the game is to take turns with opponent", has a meaning. That sentence encapsulates a whole new skill (another task pathway) that is previously learned by the robot. He might have learned a similar game, like checkers, and he is using that skill to take turns in chess.

In the sentence: "predict the future by imagining future steps for me and opponent and come up with good moves.", is creating a function to predict the future by imagining future moves by both the robot and the opponent and determining which move to make based on long term benefits. This is 1 of 2 ways my robot's brain predicts the future. The 2nd way to predict the future is the built in future prediction function in the robot's brain.

FIG. 56



In FIG. 56, the hierarchical tree is structuring the self-creating computer program in a hierarchical manner. At the top of the tree are general self-creating computer programs and at the bottom are specific self-creating computer programs. For example, E2 contains a general self-creating computer program to play any game. E5 contains a specific self-creating computer program to play the game of chess. The lower nodes (task pathways) inherit all data from higher level nodes of the tree. In other words, task pathways share data in a hierarchical tree.


decision making pathways
(self-organized)
FIG. 17A
Pathway65 = is a learning
a new game pathway.
pathway65
B1
Play a game
Play board games
B7
B2
Play a videogame
B3
Play a super
Nintendo game
E1
play chess
E4
play checkers
E8
play monopoly
Zelda
(RPG)
racing game

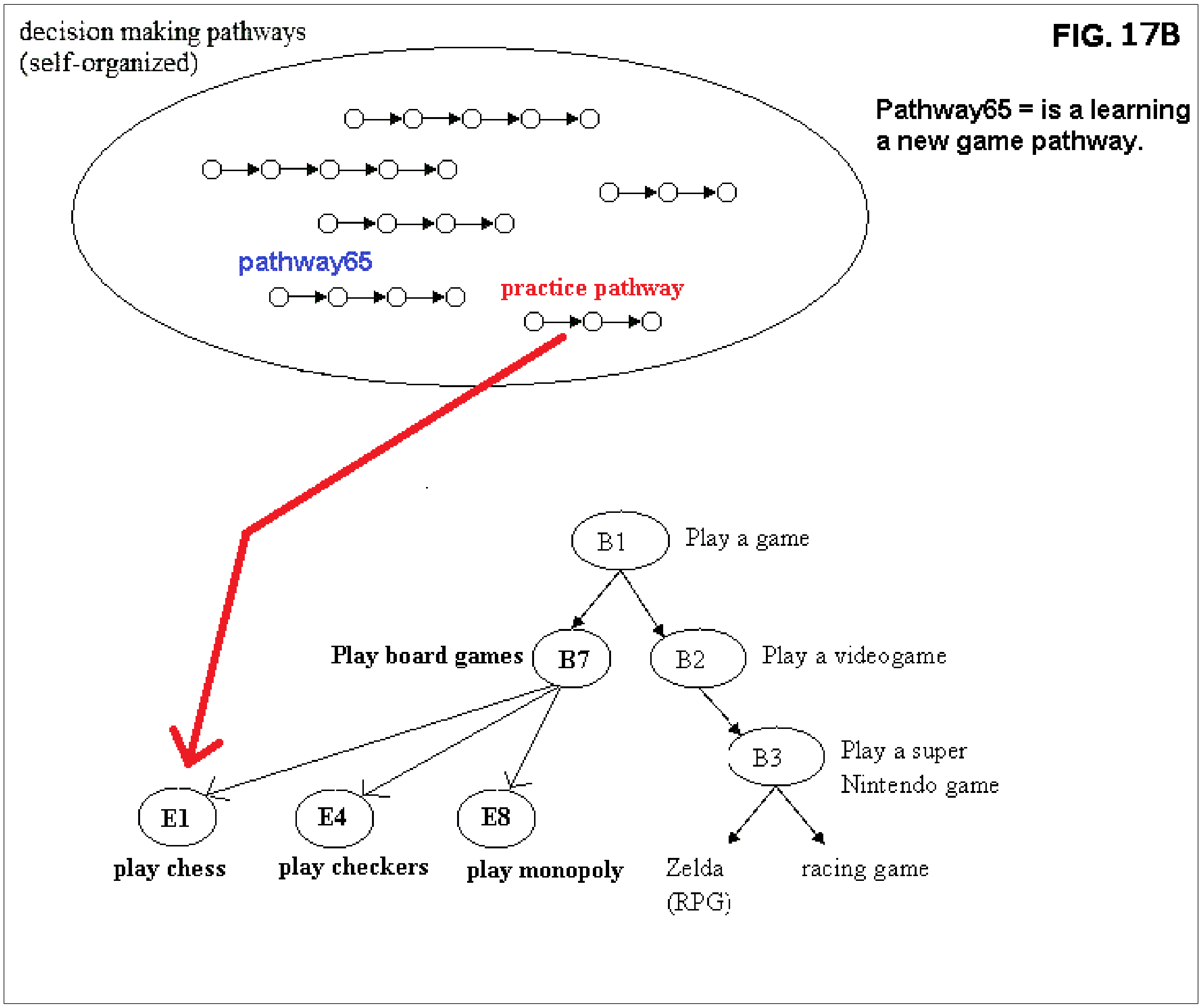


FIG. 17B

Pathway65 = is a learning a new game pathway.

In the learning a task pathway and the practice pathway, the robot is using common sense, observing the environment and decision making to seek out important facts to store in memory (FIG. 17A and FIG. 17B). The robot might learn chess by reading a 30 page book. He has to use decision making pathways to identify only the most important facts to store in his memory. The learn a task pathway stores the knowledge, while the practice pathway stores the linear actions to play chess. Let's show an example of a practice pathway and how it learns to play chess. Below is a list of thoughts from the robot's mind to create and store the right instructions in task pathways for the game of chess:

question: what is the goal again?
recall: the goal of chess is to eliminate your opponents' king.
fact recall: all the rules and procedures pop up in the robot's mind
question: i'm starting the game, what do i do first?
answer: oh, i have to take turns with opponent to make moves.
fact: oh, i have to predict future moves for me and opponent and make a move quickly.
recall: the rules say i have to flip a coin to determine who starts first.
common sense: I win, this means i go first.

ok, the pawn can move like this. the horse can move like that, the bishop moves like this. -pic1

logical thoughts: select important pieces to predict. look for my pieces that are in danger and look for pieces from opponent to kill. Randomly scan the pieces (strategy1).
these are the 4 moves i predicted. times up i got to make a choice now.
logical thoughts: move1 is better than move2. move3 is better than move4 and move3 is better than move1. I select move1.
fact: it's opponents turn. continue to observe and predict future moves (predict opponent move more than my move).

Notice that the thoughts of the robot are like a group of people in a room, asking questions, answering questions, providing logical facts, deciding on actions, predicting the future, generating common sense, etc.

Above are the linear thoughts of the robot's mind, which are stored in the pathways along with 5 sense data, hidden data, and pattern data. After using the practice pathway for several tries, the robot will have the required knowledge to play chess. If he practices for years, he will become an expert in the game. The practice pathway ensures the chess pathways contain optimal strategies and knowledge to play the game. The more the robot practices the more optimal strategies are generated and stored in the chess pathways. If the robot discovers a strategy that is better than an old strategy, the better strategy will be remembered and the old strategy will be forgotten.

robot's thoughts: I discovered a new strategy5. strategy5 is better than strategy3 or strategy1. So, in this situation use strategy5 next time.
robot's thoughts: remember strategy5 and forget strategy1 or strategy3 in situation7.

2. Time estimation events and possibility and probability
FIG. 61 depicts a sandwich pathway and shows the linear steps to make a sandwich. Most of the time steps are represented by sound sentences. Other times it’s represented by visual images or movie sequences. In each step, the sandwich pathway contains an estimation of the beginning and ending. Thus, pathways store the estimation of events.

FIG. 61

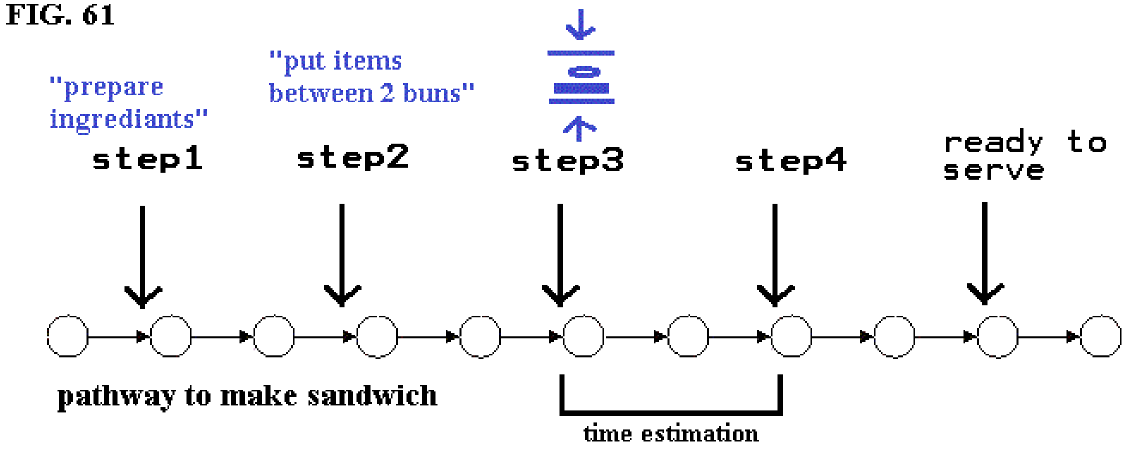
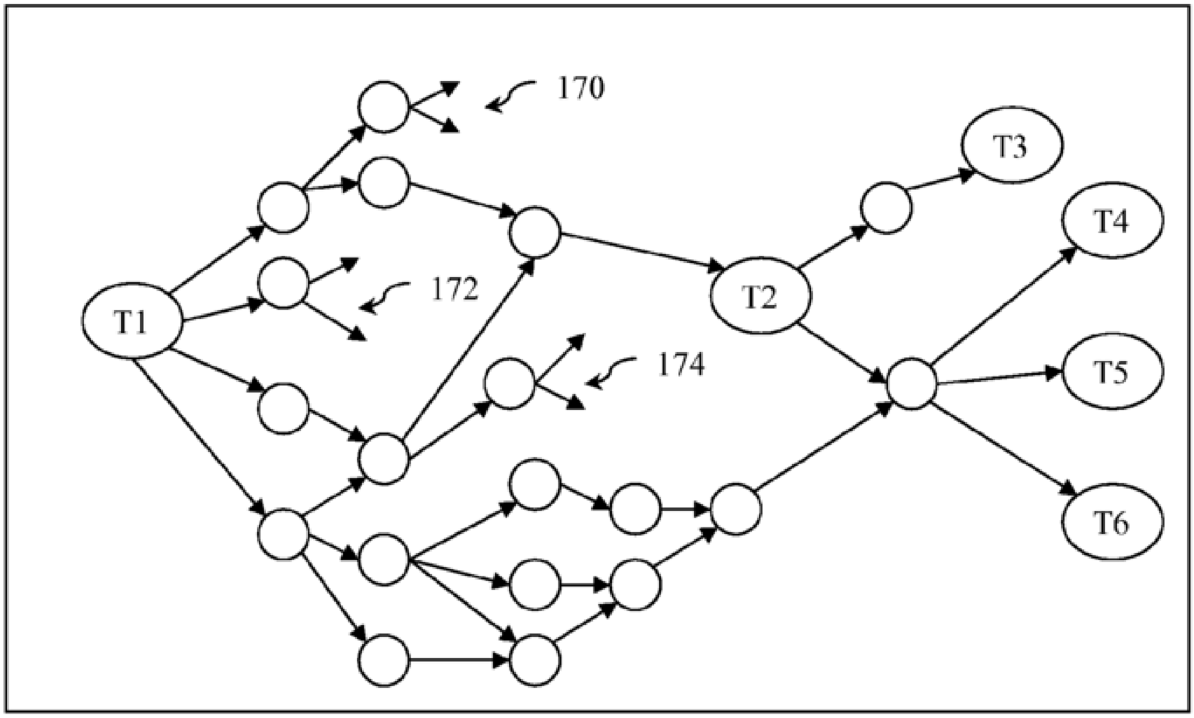


FIG. 53A depicts a sandwich pathway that shows different possibilities. Making a lobster sandwich is different from making a meatball sandwich. They share some steps, but some steps are different. The sandwich pathways store possibilities (if they are repeated experiences). And each possibility has a weight of the probability of occurring in the future. The more experiences the robot has with an event the more points added to the connection weights.

FIG. 53A



Referring to FIG. 67, let's tie this in with the self-creating computer program stored in pathways. In this case we have to use driving as an example. As the robot learns how to drive, he is constantly doing a task: "drive between the 2 white lines and watch out for danger objects". This instruction is stored in the driving pathways and is constantly being reminded (permanent instruction) because the robot is repeatedly following this rule.

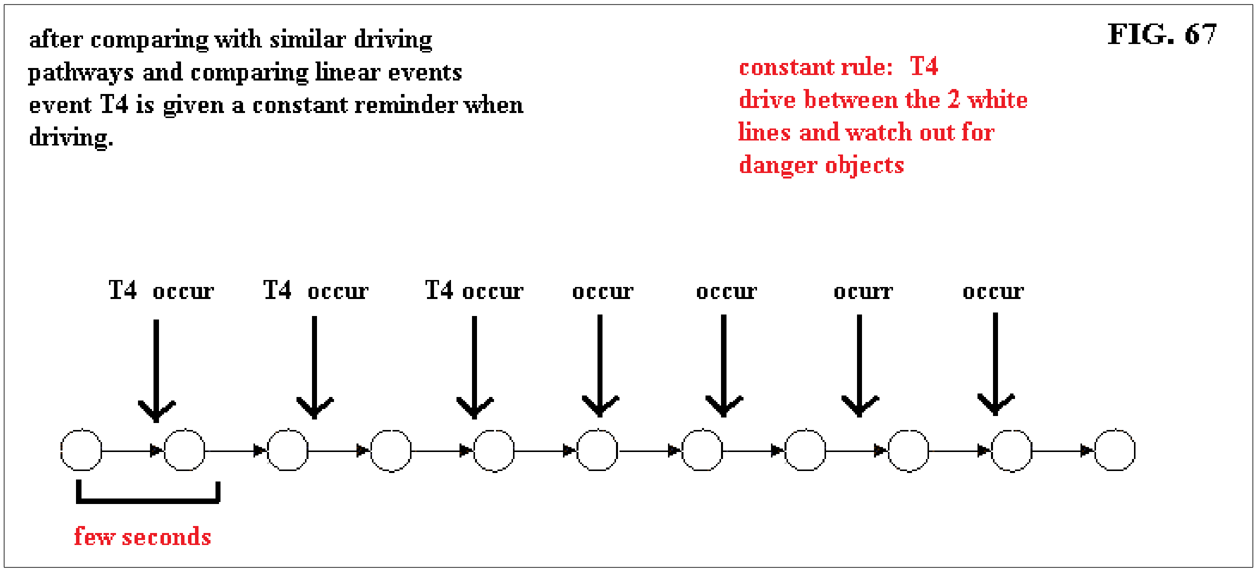


FIG. 67

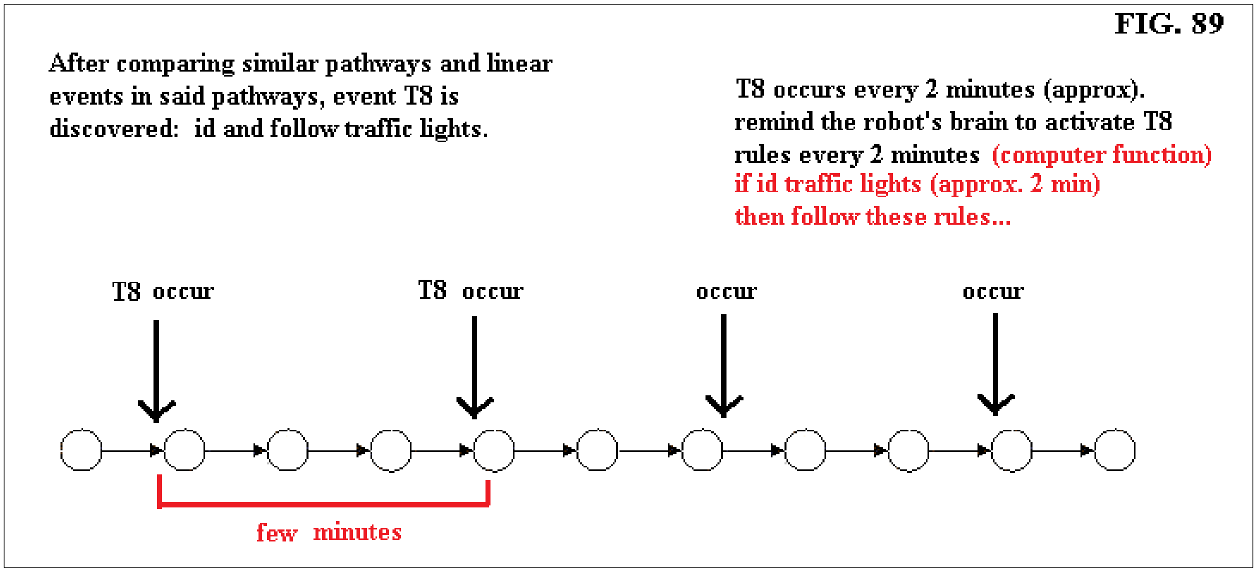


FIG. 89

Referring to FIG. 89, using the rules for traffic lights has occurred every 2 minutes based on hundreds of hours of driving. So, the robot's brain created an if-then statement from observing similar driving pathways. When the robot is driving, these operations are created:

if id traffic light follow rule4. rule4: green light is go, red light is stop, and yellow light is null.

sub-levels:
if id traffic light and car is going straight, do this ...
if id traffic light and car is turning, do that ...
if id traffic light and in the middle of a dessert, do this ...

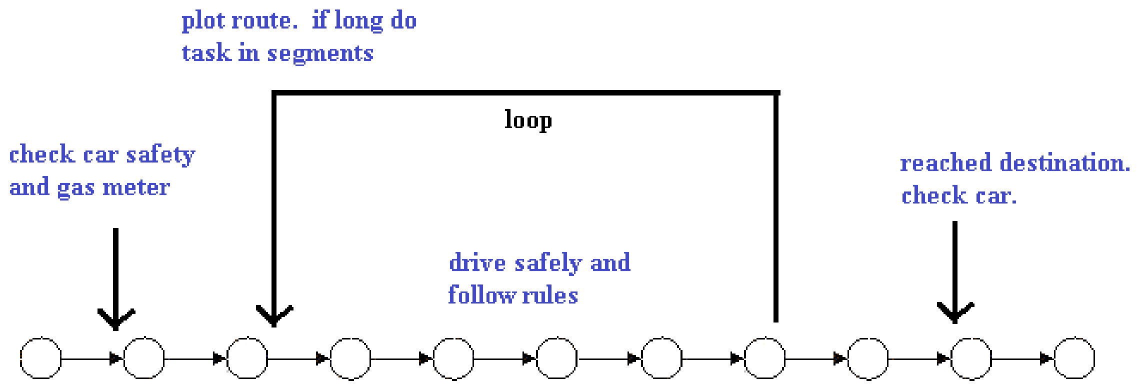
Thus, the driving pathways are observed by the robot's brain and patterns are found. Most likely complicated if-then statements are stored next to pathways because in driving school and books, they teach students using if-then statements.

The above example shows pathways store complex computer functions along with possibilities and probabilities of events. The computer functions store things like if-then statements, conditional loops, procedures, recursive functions, hierarchical functions, and event probability and possibility.

After many driving experiences, the robot will have a computer program stored in the driving pathways. Most likely the robot will have a linear procedure running as an operating system and activate if-then statements to take action.

FIG.69 depicts a general procedure to drive. At the beginning of the procedure the robot has to check the car for safety like adjust the rear view mirror and check for gas meter. Next, the robot has to identify the destination location and plot the fastest route to its location. Maybe sometimes the robot will plot routes in segments if the destination is a long trip. After the robot gets to its destination location he has to do 2 other tasks, which is find parking and check the car before shutting down. Examples of checking the car would include: make sure the shift gear is set to neutral and make sure the doors are locked before getting out of the car.

FIG. 69



Pathway to drive a car (linear procedure)

So this is the general procedure the robot is following while driving the car. At the same time the robot will use if-then statements and facts to drive safely. Most likely, he will execute if-then operations to follow rules. For example, if the robot id a traffic light and his intention is to turn left, these rules will activate in his mind:

task: if id traffic light follow rule4. rule4: green light is go, red light is stop, and yellow light is null.

sub-task:
if id traffic light and car is turning, do this ...

Solving conflicts of rules are done using logical inferencing on the driving rules container. These types of inferencing are taught by teachers in school when dealing with complex if-then statements (learned previously). For example, if the robot is driving and he ids the traffic light and his intentions are to make a left turn, something unexpected might happen, like police sirens coming from behind the car. In the rules container is a rule that states the driver has to stop all activities and let the police car/s pass. Therefore, the robot aborts all future tasks and does a brand new task, which is to pull over to the side of the road and let the police car pass. However, most driving rules are simple. The robot will follow the procedures of driving and at the same time execute if-then operations to follow rules. If the robot is in sitaution4 do task3. If the robot is in situation8 do these tasks. The more complex rules are accomplished using logical inferencing. Below are examples of logical inferencing for driving a car:

if i'm in situation1 and situation4, then i have to do task9.
if i'm in situation3 and situaton7 and situation4, but situation7 has top priority, then i have to do task1 and task3.

Doing a task like driving is very complex. The robot will be using the learning pathway and the practice pathway to learn to drive. After several weeks of driving, the robot will have optimal driving pathways to drive a car in its brain. Inside the driving pathways is a self-created computer program, equipped with a self-created database of facts, procedures and rules on driving.

The knowledge to drive a car using human intelligence.

A skill like driving a car is thousands of times more complex than playing chess. There are more rules, more goals, and safety rules to follow. How does a human learn to drive? We learn to drive from both driving teachers and books.

FIG. 45A

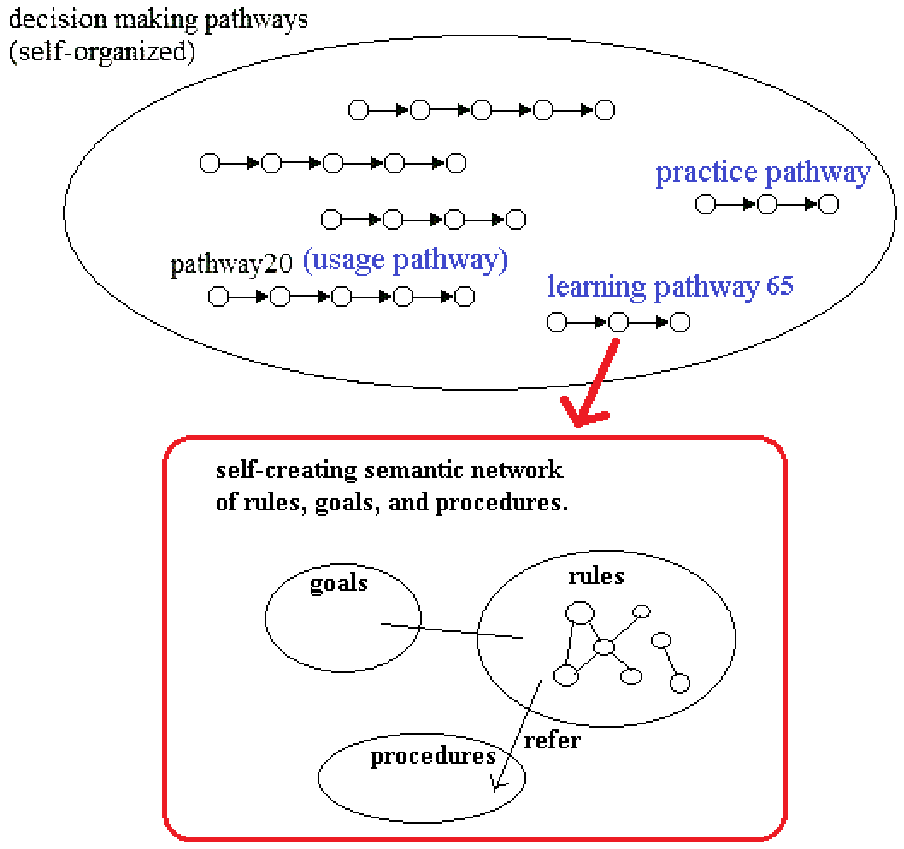
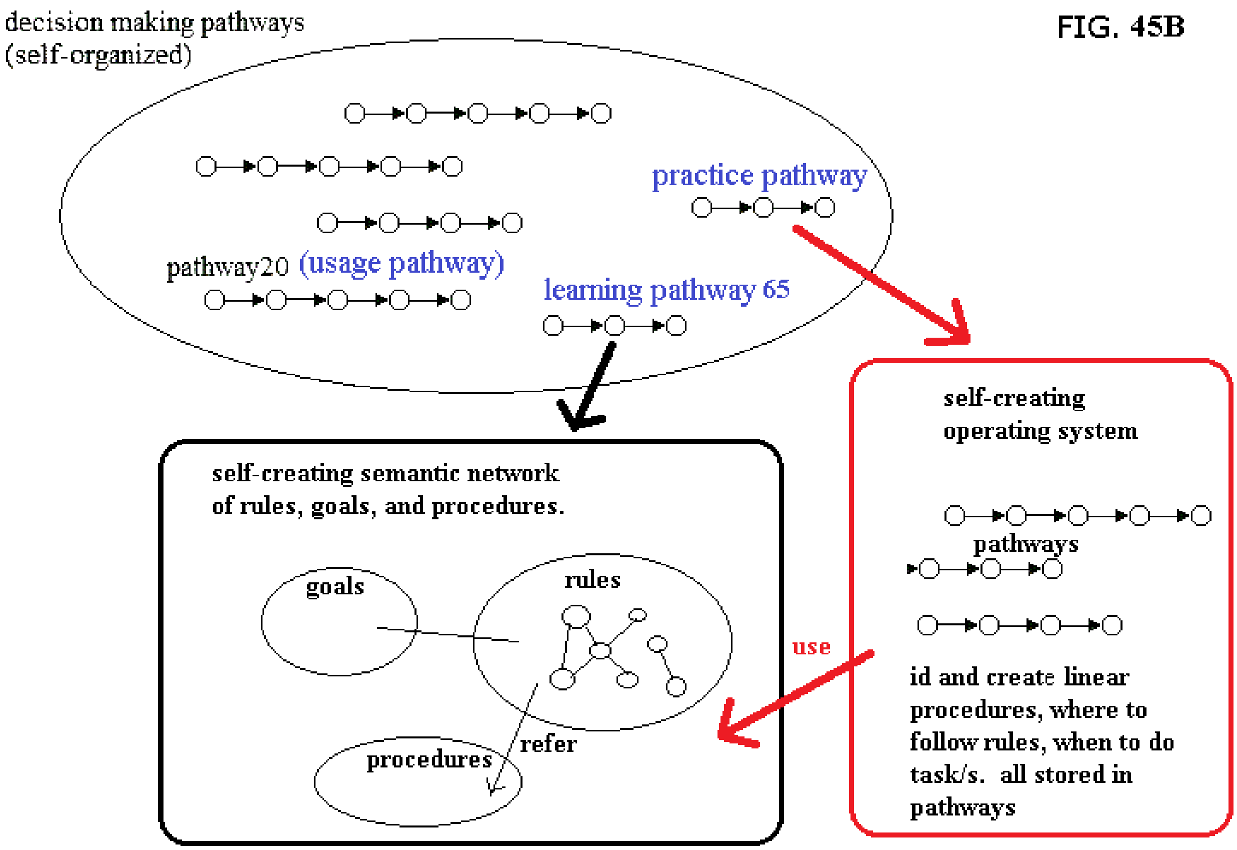

decision making pathways
(self-organized)
practice pathway
pathway20 (usage pathway)
learning pathway 65
self-creating semantic network
of rules, goals, and procedures.
goals
rules
refer
procedures

FIG. 45B



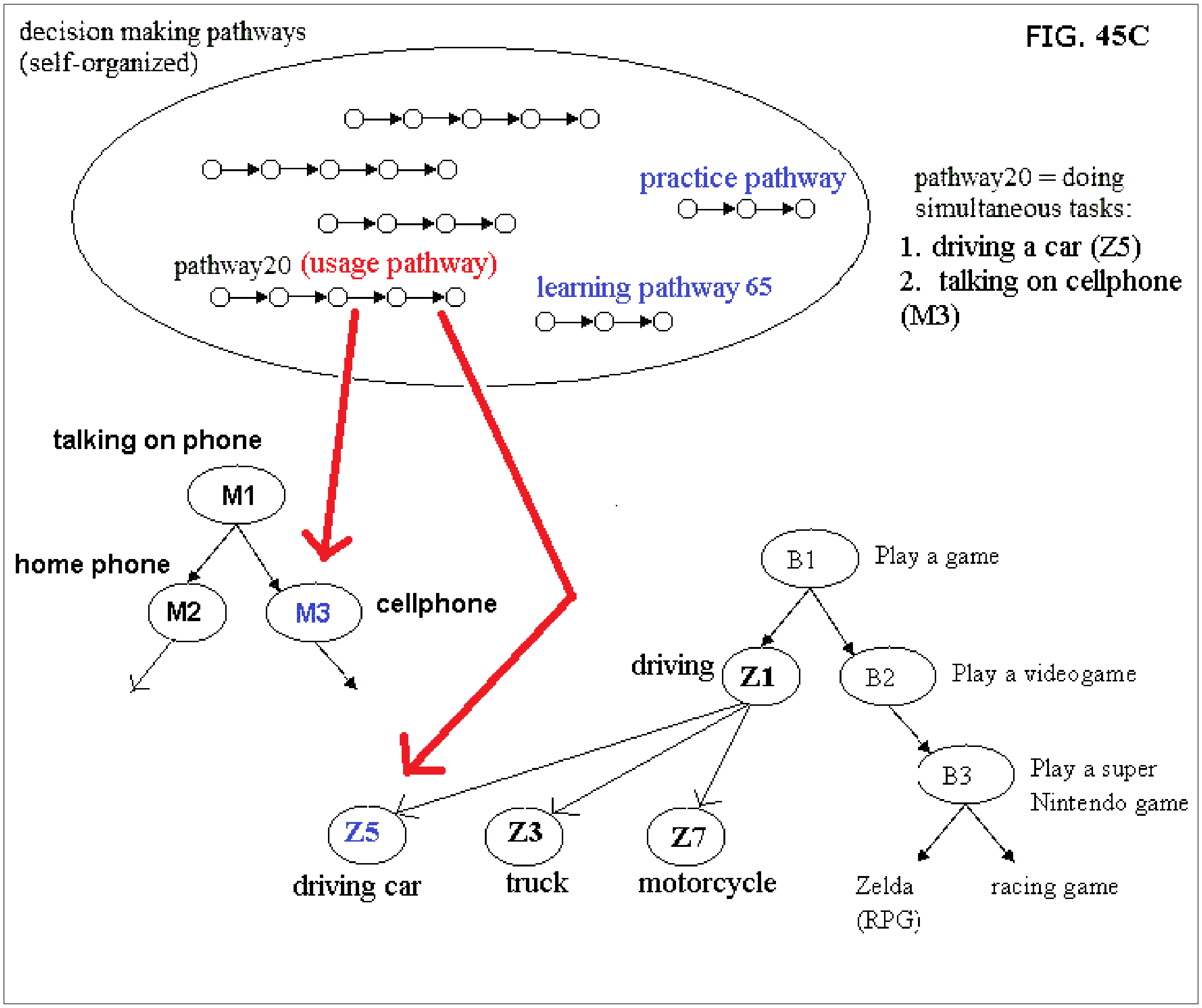
For the robot, he will use the learning a skill pathway to learn the goals and rules of driving. The robot will use logic and common sense to id the most important information in books or lectures. The sentences from driving books self-organize in a self-creating semantic network (FIG. 45A). Information will be stored optimally and visually. The goals are stored here and the rules are stored there.

Since the rules of driving is plenty, in the rules container are even more organized structures to store operational functions, recursive rules, hierarchical rules, solving conflicting if-then statements, and nested-if-then statements.

The learning pathways store optimal search functions to look for data quickly in this driving semantic network. These search functions are also self-created.

Next, the practice pathway is used to create and store linear instructions in driving pathways (FIG. 45B). As usual, the robot's mind will use common sense knowledge to create and store instructions in pathways. After months of driving and trial and error, the robot creates optimal driving pathways to drive a car, perfectly.

The learning pathways and the practice pathways create the pathways to drive a car in the robot's brain. Once the data is there, the robot can use another decision making pathway to "use" that knowledge (driving task). In FIG. 45C is a diagram depicting a decision making pathway20 to use 2 different tasks (driving a car and talking on the phone). This decision making pathway20 is doing 2 simultaneous tasks at the same time.



As the reader can see I'm not using any AI methods from autonomous cars. My robot learns information to drive from reading books or watching lectures. Humans don't have a navigation system, or machine learning, or GPS in their brains. They drive a car based on human intelligence and not by data structures from autonomous cars.

Borrowing knowledge from different skills

Let's say that the robot has mastered the skill of driving a car. Now the robot wants to drive a motorcycle. The robot knows that the majority of rules and goals of driving a car are similar to the rules and goals of driving a motorcycle.

While using the learning pathway to learn the new rules of driving a motorcycle, the robot will borrow driving knowledge from the car pathways. New rules and procedures will be stored in the motorcycle pathways. Using the practice pathways the robot will cement the linear instructions to the motorcycle pathways and to fuse both new and old rules/procedures.

After months of practicing to drive a motorcycle the robot's brain will have the optimal knowledge to drive on the road. FIG. 45D shows a decision making pathway to "use" the motorcycle pathways to drive on the road. In this case, the robot is driving a motorcycle and talking on the phone at the same time (which is dangerous).

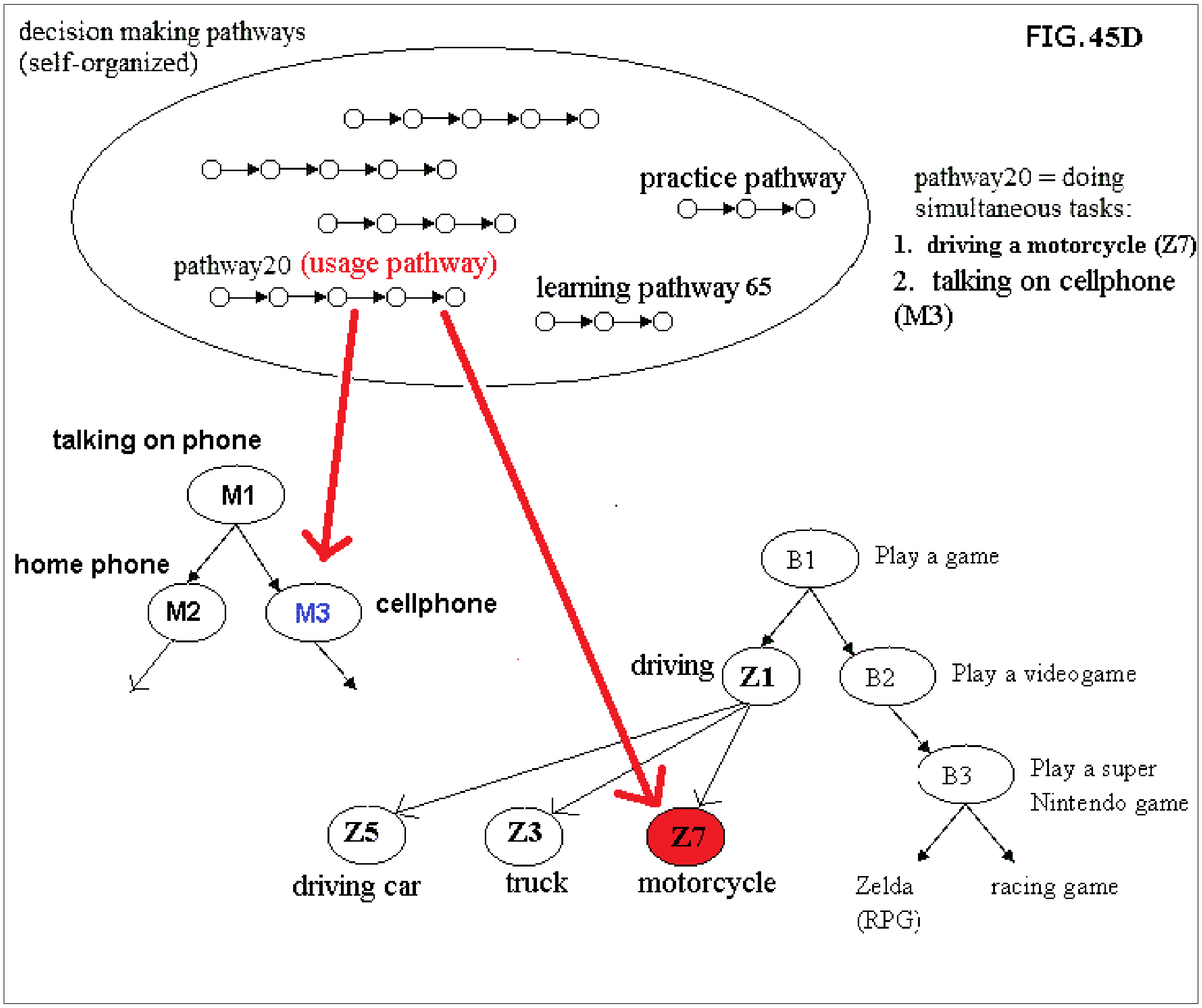
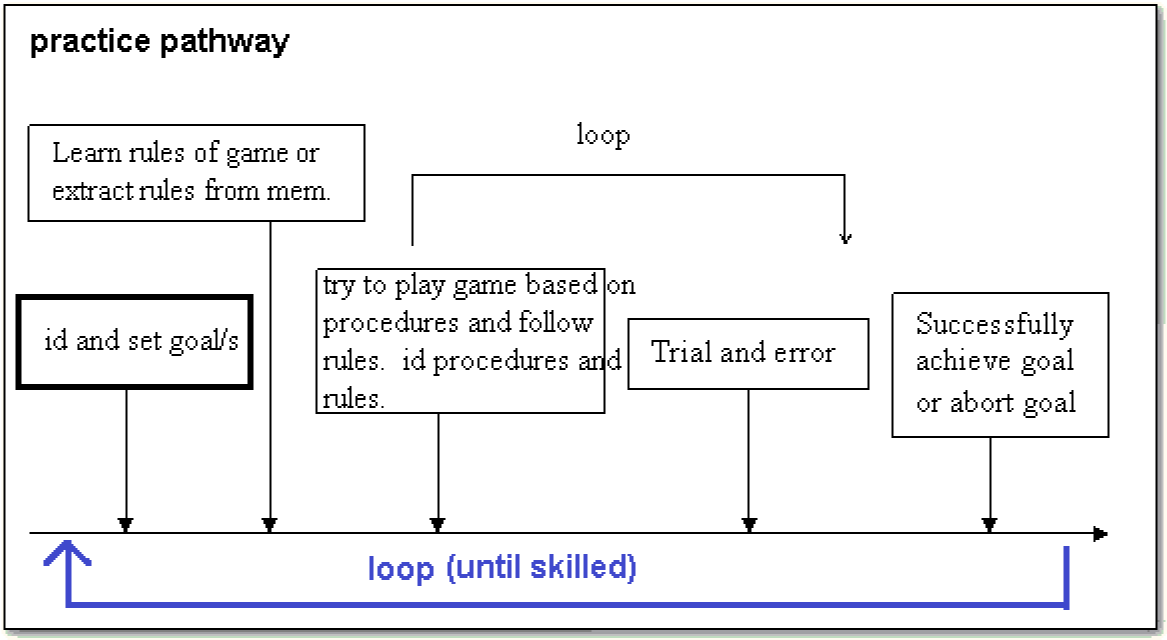


FIG.45D

The robot using a skill to do different media
Let's say the robot mastered the skill of playing basketball and someone wanted the robot to play basketball in a video game. Another type of decision making pathway will be used to take pre-existing knowledge and adapting it with a skill like playing video games. Playing video games has its goals and rules and basketball has its goals and rules. The decision making pathway will fuse the two skills together.

If you think about it, a basketball game played on a video game is different from a real basketball game. For video games, all 6 players on the court are controlled by the player (the robot). This is completely different from real life basketball where the robot is an individual. Thus, the robot has to tweak the rules of basketball in order to adapt with video game rules.
Knowledge from teachers in school will create the decision making pathways. These decision making pathways will help the robot adapt 2 or more skills together.

--- deleted page from above.
Example of robot learning a new skill, playing chess, by itself without guidance from teachers. The sentences below are the linear thoughts of the robot when it's trying to learn and play a new game: chess:
robot's thoughts:
goal: "learn to play chess".
recall fact: "in order to learn a new game i have to id 3 things: goals, rules, and procedures."
asking itself a question: "what are the rules, goals, and procedures of chess?"
seeking and providing itself with the answers: "the goal of chess is to take turns playing the game until someone's king is eliminated. while playing the game you have to follow all the rules. here are the rules. remember them and follow each one."



Language encapsulate entire pathways and allow the robot to learn different skills in a bootstrapping manner. FIG. 16A shows a diagram depicting how the robot learns to write a book. English sentences are used to encapsulate entire instructions, called pathways.
In FIG. 16A, first, the robot learns to write words and understand basic grammar rules (V1-V2). Next, he takes those skills to write a sentence (V3). Then he takes the knowledge of writing a sentence to write a paragraph (V4). Finally, the robot takes previous knowledge to write a book (V5). Notice in FIG. 16B, said robot is using previously learned skills to write a book. For

example, he is repeatedly using basic grammar rules, writing sentences, and writing paragraphs. As you might recall, writing a paragraph requires writing several sentences and using basic grammar rules.

FIG. 16A

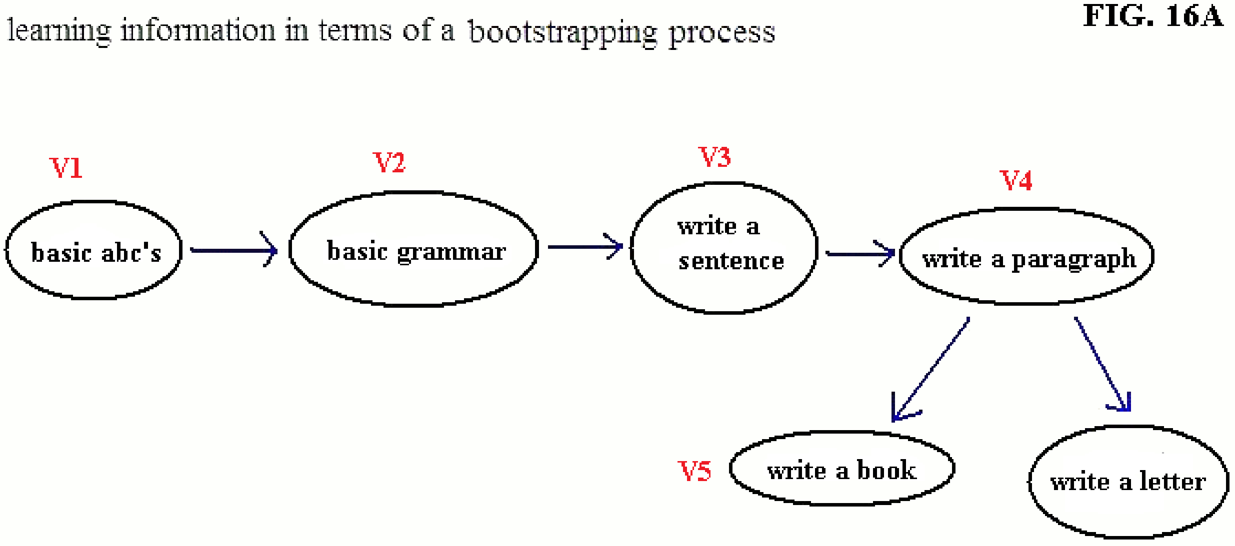
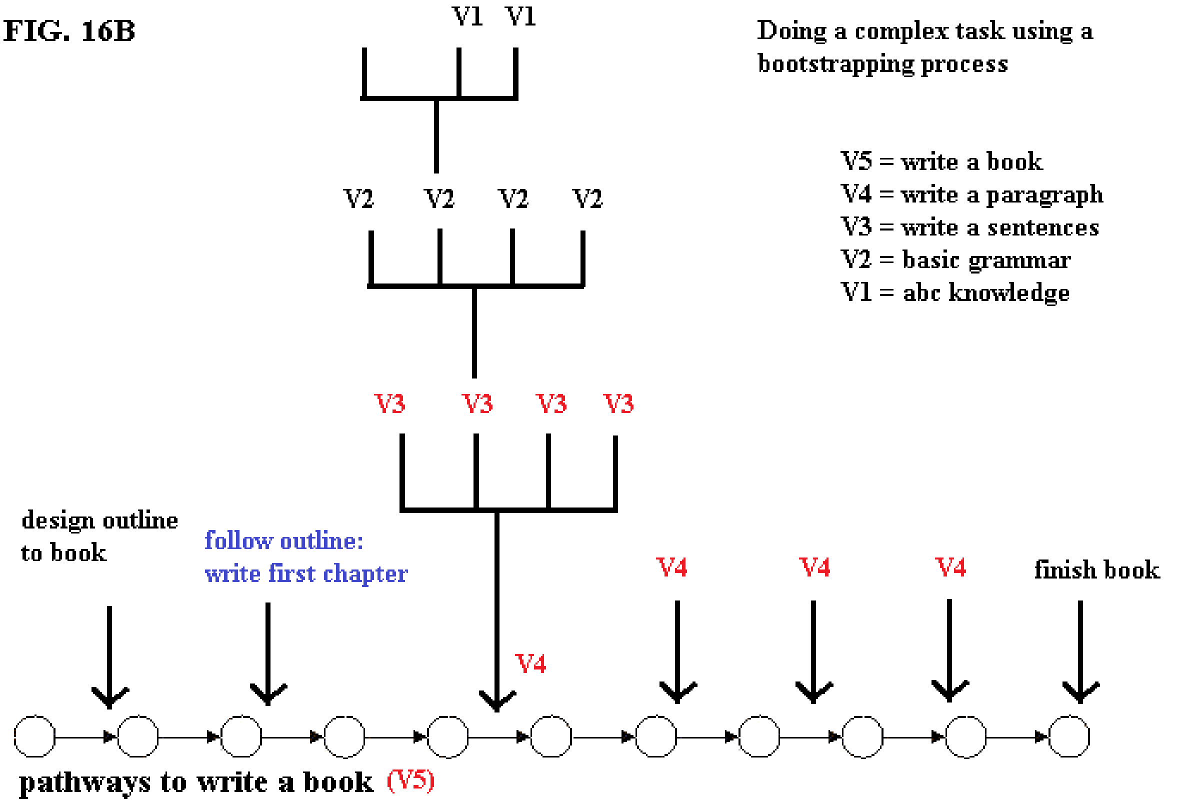
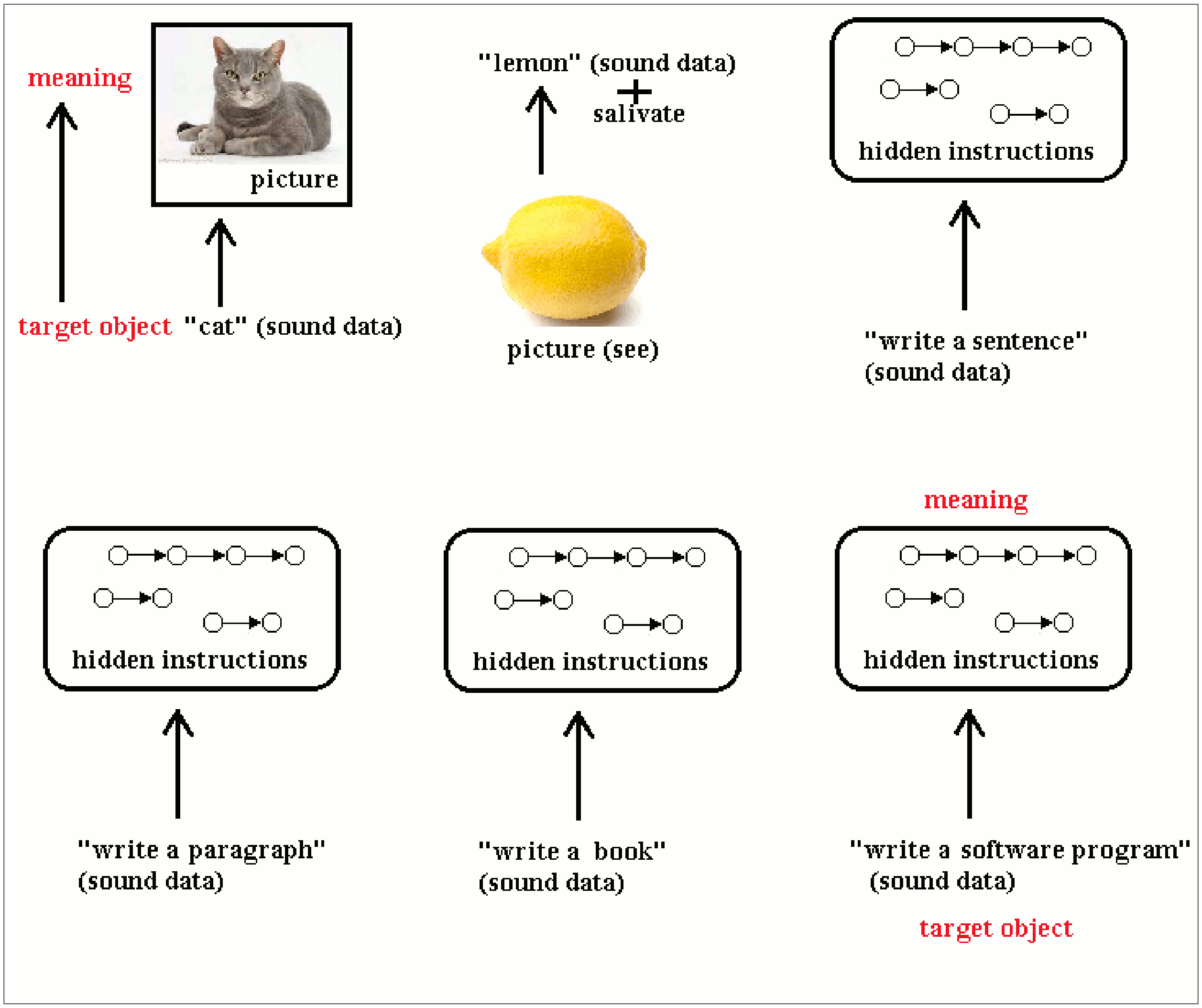


FIG. 16B

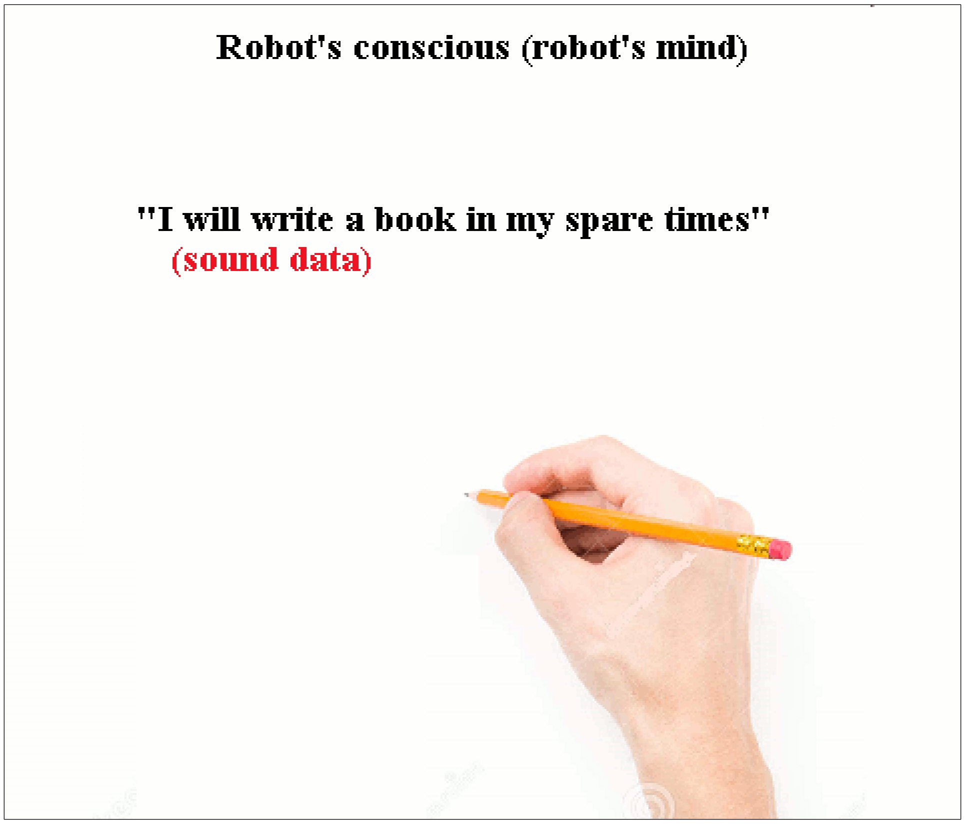


English sentences represent a task or sub-task. sentences is a fixed object that can represent a fuzzy or abstract concept. There are many cats in this world, coming in different sizes and shapes, but the word cat is a fixed object to represent a broad range of the species. In this case, sentences are used to represent or encapsulate very complex tasks. This type of encapsulation allows complex intelligence to form in the human brain, and enabling us to solve college level problems.

FIG. 11 depicts several examples of words and sentences representing abstract objects or actions. When the robot senses the target object, the meaning automatically activate in his mind.



when the robot is making a decision and the sentence: "write a book in your spare times", activates in his mind, the sentence encapsulate all the complex instructions to write a book. That one sentence represents all the knowledge the robot needs to write a book. This sentence is also known as an internal instruction, given by the robot himself to take action.



How can we test the robot to see if it has achieved Human Level AI?

In 2006, I proposed the robot college test, which gives scientists an alternative to the Turing test: "If the robot can graduate from a 4 year college with a difficult degree such as an Engineering degree or a Computer science degree, then it has achieved Human-Level Artificial Intelligence. An art degree doesn't count". In around 2009, I proposed the post robot college test, which states: "the robot must be observed for a 2-3 year period (or maybe longer) for performance and behavioral characteristics. If the robot can perform his job in his specialized field at par or at an expert level than human beings and the robot doesn't exhibit any mental disorders, then the robot has past the post robot college test".