Gen-I-Sys: a basic framework for a generalized intelligent system

Gaurav Gupta
Institute for the Study of Accelerating Change

gandalf_gaurav@yahoo.com
gandalf_gaurav@accelerating.org

Abstract

What may be the fundamental flaws behind early stage AI research that has led to widespread disappointment in terms of the achievement of AGI? What may be an appropriate framework to follow in order to design good AGI systems? This paper discusses the nature and interpretations of "intelligence" and "learning" with regard to the human brain and neural system. It also presents an analysis of the perceived differences between computer processing and that of the human brain. Gen-I-Sys is an independently initiated project that has been underway for almost two years. Gen-I-Sys utilizes the concepts of integration of sensory modalities, integration of technologies to cohesively process the sensory inputs, and the gradual abstraction of the input data over several layers where the highest or most abstract layers hold the assimilated information to which humans may most closely relate. The basis of this project is presented as a viable framework following which good AGI systems may be designed. Preliminary evidence strongly suggests that the framework is a powerful one for the purposes of AGI.

1: Artificial Intelligence and certain Philosophical Issues

For the purposes of AGI, the computer should be used as a tool to facilitate the recursive and autonomous handling and processing of vast amounts of individual input data, not as an embodiment of preprogrammed rules and logic constrained by the amount of effort put by programmers into entering the rules making up its basis for reasoning. That is, the computer should be viewed as an empty databank waiting for internal knowledge to be built by the integrated entering of individual sensory data.

1.1: Humans and Computers

The argument of whether truly generalized AI is at all a possibility or not is one that has been raging for ages, and one which is yet to be resolved. Those arguing for AI ultimately being within our reach believe that the biological brain functions on the basis of rules determined by its physical construct and properties, and that these rules may be simulated on a computer given sufficient knowledge regarding the brain. Those arguing against the possibility of AI say that there is more to human functioning than just the brain. They introduce as a separate concept the 'mind' - which they vaguely define to be some sort of a paranormal entity or phenomenon. According to them, this 'mind' is something that would take godly powers to create, being of course beyond the reach of human designs. The 'mind' gives us our intelligence, our emotions, lets us communicate easily through natural language, and enables us to be self-aware. Without it, they say, no machine will ever be truly intelligent.

But are humans and computers indeed so disparate? Humans are said to go beyond logic and scientific explanation (and into the realms of emotion, spirituality, and feeling) in their actions. Computers indeed follow nothing but logic. Do humans not seem to surpass logic only because of the complexity of millions of individual logical 'transactions' or transformations occurring in the brain? Might it not be that humans, being unable to comprehend the exact series of logical mental events (by which behavior results) due to the sheer size and complex nature of the whole process, attribute the result to something surpassing rational scientific explanation? And might not a similarly overwhelming series of logical events occurring in a computer, in the form of electrical instructions sent to the processor (analogous to neurons firing down axons and synapses in the biological brain), seem equally unexplainable? One of the reasons for the inconsistencies between the interpretations of the actions of humans and computers is that the basis for a computer's functioning is known to an extent by most, while little about the brain's exact functioning is known. It could also be that relative temporal scale differences create misperceptions regarding the nature of biological data and information processing as compared to the nature of such processing in a computational device. Even modern computers are unable to get close to the awesome power and speed of the brain (for that immensely complicated sets of parallel processors would be required). In the functional timescale of the computer, the brain is able to achieve at least several million times the cycle speed of the modern computer. By the term ‘functional timescale’ I mean the total amount of data transferred per second between different processing elements. While desktop computers reach speeds of 2 Gigahertz or more via a single transmission channel connected to a single processor, the human brain reaches much greater speeds simply by having millions of neurons transmitting simultaneously. Although modern technology can easily outperform a single neuron’s transfer rate, the combined throughput of many

massive arrays of neurons is incredibly difficult to achieve without having a similarly large number of processors working in parallel. The fact that this performance gap is due to the massively-parallel processing nature of the brain most often goes unperceived by the human consciousness. The complicated series of steps involved in a single human action or thought commonly appears to humans to be one large illogical input to output transformation. During the same time, a computer running software executes a number of instructions that, although still numbering maybe in the millions, is comprehensible to humans through the abstraction of sets of instructions into functional modules. Thus, humans are popularly known to be unbound to logic, exercising it often but not always. What is not considered is the possibility that our inability to fully comprehend the brain's functional complexity may be clouding our judgment of the logic of its input-output transformational processes.

In many respects, humans and computers are actually quite similar. Just as computers are programmed with instructions to open, save, and close files as well as to preserve the integrity of its file system by guarding important system files, humans are programmed with such things like the undesirable nature of pain and the need to reproduce. For instance, humans try to avoid painful encounters such as being burnt not because of the pain in itself. Rather, "pain" is the term humans have allocated to those sets of values, provided by gauges of internal state, that indicate an undesirable environmental situation in terms of physical structural integrity. Humans are genetically programmed to avoid such situations. We 'do not want' physical discomfort because the behavior is programmed into us, just like a computer's file-system preservation behavior is programmed into it; it too executes certain safety checks and displays warnings whether or not the end user tells it to. Unless its programming is altered (analogous to modifying the neural systems of humans), it will do what its designed to do. For the sake of comparison, we can describe the processes of avoiding pain (by humans) and preserving integrity (by computers) to be very similar. Humans 'do not want' pain; the computer can be said to 'not want' logical or structural damage. For the computer, deletion of an important core-system file by a user is its mechanical equivalent of biological pain, and sure enough, it will display a warning message to try to stop the user from causing damage.

1.2: Processing Power Vs. Intelligence Engineering Skills & Knowledge

There are reasons, for systems not performing successfully as AGI systems, which have nothing to do with the lack of processing power. Our computers are getting faster and more powerful by the day and, at this moment, it would be technologically feasible for a major organization to pool in funds towards the creation of a massively powerful parallel computer much like the human brain. Once we derive a method that shows us how to go about building good AGI systems, the necessary research and development funding will follow relatively easily. Our knowledge base in electronics and computer engineering is advanced enough to give us what we want in terms of AGI provided our technology methodology or Intelligence Engineering methodology is appropriate.

While it is hard to say exactly what is lacking in systems that fail to perform at levels comparable to the biological brain, it may be possible to hazard a few informed guesses at some of the possible reasons for their failure. For one thing, few AI systems in the past have attempted to work with integrated multiple sensory inputs. Any processing or analysis was carried out on one specific type of input at a time. Once these inputs had been individually processed, only then was thought given to integrating the information from

various sources in order to produce a more complete picture. Furthermore, the nature of the processing itself may be too predetermined and limited in terms of scope. It is common to encounter AI programs that apply pattern recognition techniques such as edge detection in order to create a meaningful internal representation of the visual circumstances. Similarly, many programs apply relevant pattern recognition techniques to auditory data in attempts to filter out useful information from the jumble of input data. The dependency on the application of these very techniques themselves, whether to visual, auditory or other data, may be what is restricting the potential of such systems for intelligence.

Specific techniques have a specific range of possible inputs that can be processed as well as a range of specific outputs that can be arrived at, especially when they are applied to one type of sensory input at a time. In order to clarify this argument, let us consider an AI system that uses elements of image analysis and processing as its basis of functioning. The system accepts pictures of its surroundings, upon which it applies line and edge detection, amongst other such techniques, to the images in order to try to isolate meaningful elements of the picture such as squares, rectangles, lines, edges and so on. If the system is successful in its image analysis endeavors, then the result is a collection of individual shapes and object elements, the total set of which correspond to the instance (at which the picture was taken). However, what is to do with this collection? It has no other environmental information to which it may correlate this information. It is only able to carry out the assignment of the gathered and processed information to symbols (or other representatives) that its programmers have preprogrammed into it. That is, it is only able to restrictedly enrich its predefined metadictionary, which in this case consists of pattern to symbol mappings. In addition, it is not that this is enrichment in the true sense and can go on indefinitely. It assigns patterns to symbols that are then used to create a form of internal representation of the instance. Here the system hits a sort of dead end. Its programming is based almost entirely on the predefined symbols. There is only so much that a system may be able to accomplish if given a limited set of parameters from which to work. The system is unable to increase its basic knowledge (for example, by adding symbols or symbolic representations) and thus is restricted in the versatility of action. Besides, while the system's programmers would have known what to do in response to the occurrence of any particular symbol, the system itself has no such real world knowledge to help it formulate actions, and therefore is not expected to be able to make profitable use of anything it itself adds to its basic knowledge set.

Although some advanced AI systems, such as intelligent surveillance and security systems, do indeed deal with complex sets of varied inputs and calculate an appropriate output for them, even these do not use wide enough range of sensory modalities as compared to humans and do not integrate generally enough the ones that they do use. For instance, the integration of sight and sound often involves preprogramming such as "IF input_image == image1 AND sound1 == food THEN move_forward". The system is not given a way to discover for itself that moving forward is a smart thing given a particular image that correlates temporally the sound of the word "food". If that were to be the case then system might have learnt that for an input instance where both the image "image1" and the sound "food" are present, then moving forward leads to increased energy levels. Although it may have taken the system a few rounds of trial and error in order to figure out that moving forward is the best option given image1 and food, it would eventually itself have generated the code "IF input_image == image1 AND sound1 == food THEN move_forward".

1.3: The need for the senses

Intelligence is very obviously dependent on the integration of multiple sensory inputs. It is unreasonable expect to hear a blind and deaf person critically discussing the latest movie! The lack of sight would imply the absence of the concepts of color, shade, hue, and depth. The lack of hearing would similarly imply the absence of concepts such as loudness and melody. The lack of these concepts would in turn result in reduced intelligence arising from the deprivation of many aspects of 'common sense'. Our common sense arises from our ability to attribute cause and effect to instances and phenomena. For example, we know that 'seeing' (visual sensory input) the shape and color of a flame means that getting too close to it could lead to feeling pain (touch sensory input). Thus, the correlation of different sensory inputs is the basis of intelligence as we know it. AI systems that do not involve the integrated use of multiple inputs should not be expected to exhibit any level of intelligence beyond the rules and logic preprogrammed into them.

1.4: Learning and Understanding – sensory correlations

Both learning and understanding are phenomena that involve processes of correlating different sensory inputs [see 6]. It does not seem possible that a system processing in isolation any individual sensory input (whether auditory, visual or other) will be able to learn, understand and evaluate much from its environment. The human experience of life is arguably a series of linkages and correlations between input data at discrete instances. Knowledge exists through association. To 'know' what a dog is, for instance, is to have an internal correlation between the sound of the word 'dog', the image of a dog (an approximate human brain equivalent of a pixel pattern of certain colors), and maybe the sounds and images associated with a dog's barking or nature of movement. Any isolated sensory pattern would have no meaning. Consider a strange sounding and completely unfamiliar word. Unless associated with another sensory input, the word would make no sense whatsoever (other than just to be a strange sounding word). With no other information about it, it could potentially be a noun, a verb, an adjective, or some other concept. To understand what it means, there would have to be a corresponding set of visual or other sensory data. If associated with visuals of movement, it would imply an action. If associated with an object, it would imply a noun. Even explaining its meaning with the use of another word would in the end cause information associations to form. In this case, the corresponding other sensory data for the word being used for explanation would be assigned to the new word. If considering understanding or learning as the formation of new information correlations, then the association of different types of inputs is something that cannot be escaped. Developments in Embedded Sensor Networks (ESNs) show high dependency on multi-modal sensor fusion [13] for better information mapping.

1.5: Reasoning

The term "reasoning" can have many interpretations. In some cases, it involves the identification of links or relationships between two objects or phenomena. If we see a man or a woman walking the same dog every evening in the local park then we assume, or reason, that the man/woman is the owner/guardian of the dog. In other cases, reasoning involves the logical linking of sequences of actions towards the formulation of plans - such as planning to use a window as an escape route in the case of a fire. Thus, we have associative reasoning (linking, via

causal or non-causal relationships, static images, entities, etc.) and deliberate reasoning (forming logical sequences of actions that will lead from a given state to a goal state – or planning). Deliberate reasoning almost always involves associative reasoning. In the second instance given above, the development of the reasoning of using the window as an escape route is preceded by the associative reasoning which associates fire with burns and bodily injury as well as by another associative reasoning which associates the window with a suitable escape path. Associative reasoning is thus the foundation for successful deliberate reasoning because (with reference to the example) associating a solid wall with an exit would be disastrous. Both (associative and deliberate) however involve links or relationships between somewhat abstracted concepts, ideas such as the objects: "window" and "fire", and the actions: "escape", and "getting burnt". Therefore, in the case of an AGIS, reasoning as a process cannot work on inputs taken directly from the environment, but must utilize refined information fed out from more advanced levels that carry out the information abstraction from raw input data.

1.6: The basis for the computer's functioning and use that may be getting in the way of machine intelligence

The authors of the book 'Mind over Machine' (Dreyfus, Dreyfus) [4] stress the importance of intuition in intelligence. They argue that computers that are used purely as traditional logic machines may never be able to demonstrate intelligence as we know it. This is something that has probably been learnt the hard way by many developers of expert systems who have found that it is near impossible to program in enough rules for a system to react and adapt flexibly to a changing environment. The world is too broad and diverse. Unless a system has provisions to assimilate any situation, not just those foreseen by its developers, and to update and enrich its basic knowledge set unfettered by programmed symbolic definition sets, it will never be truly intelligent. There are ways for a system to be able to stretch the bounds of such predefined data dictionaries, but this is finite. If run for long enough, a time will come when a particular situation will not match anything the system has been given to know, and there it will get stuck. Hence, if machines are ever to be truly intelligent, they cannot have strictly preprogrammed instruction sets.

However, certain neuronal reactions to inputs are indeed in a way preprogrammed. It is known that, in the brain, certain neurons fire given certain circumstances. The physics, biology and chemistry of the system (the brain) decide the behavior of the neurons. The complex interactions between neurons, which in essence function with the strictest logic (considering the physical dynamics of the system), in turn produce the complex reactionary behavior of the brain. Thus, the programming of the brain exists at the micro level. Although reactions to entire real world instances and phenomena are not predefined, the reactions of neurons to individual bits of sensory input data indeed are. The combined characteristics of micro level behavioral predefinition and the complex interaction of those behaviors within a super system gives the brain the necessary versatility to deal with almost any kind of input data in some coherent manner.

The brain deals with the full range of input values for the senses and these probably number in the billions if not more. The study of artificial intelligence has so far largely attempted to program in reactions to permutations and combinations of a (still massive) subset of those billions of pieces of information. A preprogrammed description of an object is a combination of several bits of individual data. A description of another similar object is a

different permutation of the same combination of individual data. An entirely different object requires a combination of almost entirely different data. If trying to account for most or all of the possible permutations and combinations of the full range of sensory values, we end up with figures the size of which cannot even be accommodated in the memory of the world's most expensive and most advanced computer. It is no wonder then that developers of expert systems, or indeed of any type of AI system using a predefined knowledge base, have failed miserably in their efforts at achieving generalized artificial intelligence. Maybe the predefinition of responses to compound instances or circumstances should not be attempted; instead, we might attempt to deal with the autonomous generation of the rules themselves that govern the outputs with respect to the input data and experience. A good approach indicated for synthesizing an AGIS appears to be the design of an engine which is intended for recursive self reprogramming, not in terms of altering its base commands but in the sense of creating new "functions" or "operations" out of the existing library of base commands. Efficient engineering of such a machine would involve the use of a concise but powerful library of base commands, each of the components of which lends itself naturally to the process of recombination with others for the synthesis of newer and complicated operation strings. The Gen-I-Sys engine, discussed later, in fact does this in a subtle way.

2: The likely nature and structure of a good AGI System and the Gen-I-Sys machine

Before starting to design or build an AGI system, we must build up a description of the system's intended capabilities (functional requirements). In all engineering applications, the intended capabilities lie in the purpose of the system to be constructed. The problem here is the fact that we want the system to demonstrate general intelligence, a term that has produced much disagreement and argument amongst concerned researchers. What hope then have we to establish clearly the requirements of the system?

2.1: The intended capabilities of a good AGI system

Let us approach this issue from a less abstract perspective than that taken by most earlier researchers. Instead of trying to lay down constituents or ingredients of general intelligence itself, let us identify the broad categories of actions or activities of which humans are capable. Then we may initiate the attempt to create these capabilities in our AGI system.

Humans can (and thus the AGI system should be able to)*:
1. Move body parts about physically
2. Hear
3. See
4. Smell
5. Feel (here I refer to the feel of touch and not emotional feelings)
6. Vocalize or Speak (in the sense of making noises via the vocal chords and not linguistic speech)

This much at least is what we can firmly establish by simply observing the external interface that humans present to the world. If we consider the entire human system to be a black box, then these characteristics we may conclude to be the basic constituents of human behavior. The attributes of "understanding", "learning", and "emotion" are not considered here, as they are internal to the human system. These do not *constitute* behavior - they *contribute* to it. Taking behavior to be "a regularity observed in the interaction dynamics between the characteristics and processes of a system and the characteristics and processes of an environment" (Luc Steels quoted by Aron Malkine [1]), the behavioral constitutive elements correspond to what is being done while the behavioral contributive elements correspond to how the behavior arises.

The "internally coded, inheritable information" [7], or Genotype, carried by all living organisms, holds the critical instructions that are used and interpreted by the cellular machinery to produce the "outward, physical manifestation", or Phenotype of the organism.

The set of behavioral constitutive elements for a system then clearly comprises its basic phenotype. A basic phenotype may be differentiated from an advanced phenotype in the sense that, while the basic phenotype defines the simplest actions available to the system, the advanced phenotype defines the behavioral responses of the system that are produced, via complex internal processes, through its interaction with its environment. Thus movement, hearing, sight, smell, feel and vocalization, all correspond to the basic phenotype while characteristics such as recognition of objects and faces, display of emotion, and learnt skills (e.g. walking, talking, etc.) correspond to the advanced

phenotype. In this context, I take the genotype to be the processes that control and decide learning, understanding and reacting with respect to the advanced phenotype, with the genotype accepting raw data inputs via elements of the basic phenotype and expressing itself to the environment via the same. That is, the genotype describes the rules that govern the behavioral reaction of a system to its environment.

In the list given before*, we have established the behavioral constitutive elements (i.e. the basic phenotype) for our intended AGI system. Extrapolation of the basic phenotype with regard to certain facts about general human behavior would help us to draw up a (perhaps incomplete) description of the behavioral contributive elements (i.e. advanced phenotype) that the AGIS would have to demonstrate. Then we have to figure out a suitable equivalent representing something of a genotype.

In order to arrive at the required advanced phenotype for the AGIS let us again approach via broad categories of human abilities.
Humans can (and thus the AGI system should be able to):
1.Recognize objects and faces
2.Understand spoken language
3.Generate linguistic speech
4.Modulate efforts and display moods according to some internal metrics of state or well-being

To correctly arrive at the genotype for the AGIS, we must first ascertain the nature of the correlation between the input elements of the basic phenotype, the elements of the advanced phenotype, and the output elements of the basic phenotype.

Input elements of the basic phenotype:
1.Hearing
2.Sight
3.Smell
4.Feel

Elements of the advanced phenotype:
1.Recognition of objects and faces (broad pattern recognition)
2.Understanding spoken language (speech recognition)
3.Generation of linguistic speech (speech synthesis)
4.Modulation of efforts and the display of moods according to some internal metrics of state or well-being (awareness of internal state and reflective behavior)

Output elements of the basic phenotype:
1.Physical Movement of system parts
2.Vocalization (generation of sound)

Note that the six elements comprising the basic phenotype are split up between the two categories of input elements and output elements. The correlations shown in the diagram below represent the conceptual structural requirements for the genotype of the AGIS.

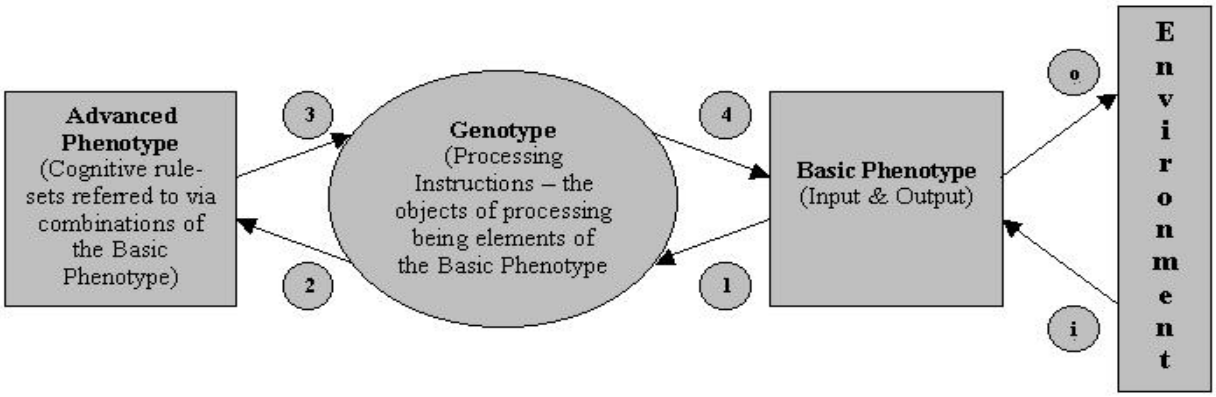


Diagram 1

i >> Inputs to the system from the environment
1 >> Input elements of the basic phenotype must be registered as images, sounds and other sensory data in data files and are sent into the genotype area
2 >> The genotype area must filter or do some preliminary processing of the raw input data and then send the filtered information to the advanced phenotype area
3 >> The advanced phenotype area must extract abstract data from the information and, based on certain preprogrammed guidelines, send a high-level description of the actions now required to be executed by the system to the genotype area
4 >> The genotype area must translate the high-level information into low-level output data and then send the appropriate instructions to the output elements of the basic phenotype, which in turn must trigger the appropriate physical events
o >> Outputs from the system to the environment

Under this framework, the system has the capacity to take in input data, process it, and output either movement or sounds. However, what is to be the nature of the processing that will contextually enable the system to respond to speech with speech, learn simple and complex actions, and build up internal maps of physical space for navigation? That is, what is to be the composition and structure of the genotype of the AGIS?

While we may not know exactly what comprises intelligence, knowledge or understanding, we do know of three basic ways by which a human being possesses these. Knowledge is clearly the basis of both intelligence and understanding, and humans either 1. have knowledge already built into their systems, or 2. discover the knowledge by imitating actions of others, or 3. discover the knowledge by trying out random actions. Amongst the several types of learning that are described in popular taxonomies [10], the two categories of learning that shall be considered relevant here are 1) inductive learning and learning from analogy, and 2) learning by experimentation and discovery.

How much knowledge is prewired into us and in what way are questions to which we have few answers given the current developmental stage of the sciences. Thus, the options left to us for the design of the AGIS are to enable the system to learn by imitating and to enable the system to learn by executing random actions and then checking the outcome of those actions. This means that we must program into our AGIS the means of imitating observed inputs (actions and sounds) in order for it to check the results of imitating particular inputs (in terms of system power levels or similar metrics). In addition, we must program into our AGIS the means of executing random actions and again checking relevant metrics to determine the usefulness of that action. It is required that these checks of

internal indicators of well-being return contextual results. That is, if the system moves backwards (whether by copying or by initiating random actions) when there is a red box in its field of vision and this produces increased levels of energy, then both the visual data as well as the action must be remembered. Storing both pieces of information as a record would allow the system to autonomously move back again the next time it sees a red box. Simply searching through its records of past successful actions in the context of a red box visual would lead it to the right (desirable) action. It is also a requirement that the system be able to overlap outputs that are separately identified as being desirable by its internal processes. For instance, given a particular situation, one search process might return the action of moving forward and another search process might return another action of generating a particular sound. In this case, the system must initiate both as soon as they are identified. Therefore, the effect may be such that the system moves forward and 'says' something at the same time.

The integration of multiple sensory inputs and the gradual abstraction of the input data over several processing layers are also important requirements. See the "Learning and Understanding – sensory correlations" section presented earlier for a discussion of the importance of the integration of multiple sensory modalities. Regarding data abstraction, no system that even pretends to be generally intelligent may operate exclusively on its raw input data in a single step transformational process in order to get screened outputs. Our system here must thus have several layers or levels that involve themselves in various orders of data abstraction (in synchronicity with Paul Bush 1996 [3]).

This simple model of incorporating 'learning' into the AGIS is more versatile than it first seems. The AGIS is able to do a lot more than just learn to respond to simple sensory data with simple physical outputs, as will be described in a later section.

2.2: Design features required for the achievement of the intended capabilities of a good AGI system (AGIS) as employed in the Gen-I-Sys machine

The requirements established in the previous section are simple enough on their own, but may be problematic where the exact means of implementation are unclear. The design of the system as discussed here provides a description of the implementation.

The genotype is represented and implemented by a Primary Control Module (**PCM**). The input elements of the basic genotype are represented and implemented by an Input Control Module (**ICM**) while the output elements of the basic genotype are represented and implemented by an Output Control Module (**OCM**). The PCM accepts raw data files from the ICM carries out some preliminary processing on them. This data is sent as Input Blocks (**IB**s) to the Advanced Analysis Module (**AAM**), which represents and implements the elements of the advanced phenotype. The AAM uses preprogrammed guidelines to store categorically/hierarchically the IBs and to derive system outputs as high-level descriptions in the form of Output Blocks (**OB**s) that are then sent back to the PCM. The PCM translates the OBs into raw output data files and sends them to the OCM. The OCM converts these output data files into the appropriate signals and sends them to the concerned software/hardware, thus making the system move, generate sound, etc. All of these modules are intended to operate in parallel - the sequential interaction between them, as just described, identifies the prerequisites for a module before it can carry out any new useful operation.

It has been established that the inputs and outputs all correspond to combinations of the basic phenotype elements. That is, if the system registers a visual image of a red ball then it is seeing it; if the system registers a sonic waveform corresponding to the word "danger" then it is hearing the word; etc. If the system's central control sends electronic signals to say wheels, making them turn clockwise, then it is moving forward; if the system's central control sends electronic signals to a sound synthesizer device then it is (perhaps nonsensically) talking.

The system has several sensory inputs (clearly the more there are the more 'intelligent' the system will be). The processing of those inputs is carried out by an algorithm. The algorithm will run continuously, taking in data from the physical input hardware, assimilating it, and, when necessary, passing data to the physical output hardware. Some assimilated data will be chosen for grouping and storage in memory as objects or as abstracted concepts.

The system has several layers or levels that involve themselves in various orders of data abstraction. The lowest level (Level 0), takes in all available sensory input data directly from its environment through its hardware, as well as directly from its internal sensors (of internal state). Here the data is processed into some form of useful abstracted information which is passed on to the next level, Level 1, as objects (sets of input data of a particular nature). Level 1 then takes this information, processes it further to get information of further abstracted nature, and passes it on to Level 2 as concepts (a concept being a collection of objects connected by relationships that are defined by the characteristics of variation or stability of the external environment). Ideally there would be a significant number of levels, at each of which screened and processed information would enable the formation of more concepts of varying levels of abstraction. Abstraction here would mean the definition of concepts in terms of objects or in terms of other concepts as determined by the system in other (lower) levels. In principle, this type of architecture would allow the system to extract simple information sets from the environment, and then to use those to form more complex or abstracted information sets.

Inputs to each Level:

The model functions with respect to discrete environmental instances. Depending on the hardware sampling rates the system will accept environmental data a certain number of times every second, each set of instantial data (an instance) being somewhat similar to a single frame in a movie. There would however be different sampling rates for different sensory inputs. While video could be sampled at 20 frames per second, audio would have to be sampled at a much higher rate, say 10kHz or 10,000 sample slices per second. This would mean that for each frame of video captured, 500 audio samples would have to be buffered. Thus, where one instance of visual data would correspond to one frame, one instance of audio (corresponding to that visual instance) would comprise 500 audio samples.

Level 0: At the top level, or level 0, the system takes in input data directly from its sensor hardware (both internal and external). The processing carried out here identifies those elements of the input data that correspond to some variance or invariance (from the last set of input data). Sets of the data elements identified in this manner are passed on as “property sets” to the next level (level 1) for storage, with the property sets of all the input types grouped together to represent one specific object (which does not necessarily need to correspond to an object in the real world). Each object will consist of property sets that are simply the categorization and clubbing together of the

data from the full range of different inputs available (e.g. visual, audio, etc.).

If any particular property set (or even a specific object) is encountered more than once, their existing definition will not be overwritten with the new data, but a new record will simply be added on. This should allow the system to have multiple views on any situation, thus increasing its diversity.

Level 1: The inputs to this level are the objects that are passed to it by level 0. Whenever a property set persists consecutively over more than one instance, that particular property set along with all other property sets occurring over those instances will be registered in order of occurrence and sent to the next level (level 2) for storage as concepts.

Level 2: This level will primarily hold the individual concepts. Those concepts that occur simultaneously over one or more instances will be linked with each other. Combinations of linked concepts can be passed to the next level (level 3) to form higher order concepts (hyper-concepts).

The nature of data storage:

In order for the system to maintain maximum flexibility, any input data that is to be stored must be converted into a relative representation form before storage. This means that it is the pattern of the data that is mostly stored rather than the absolute data itself. For visual data, this would be the set of pixel information taken relative (relative position and color) to the first (which may be assigned to be the left topmost pixel), while for auditory data the frequency ratios relative to the first could be taken (as human hearing operates on the basis of logarithmic intervals).

The nature of information processing in the model:

In order to explain how the outputs are expected to result, first must come an explanation of how the information that is to reside at the various levels is determined by the system.

At level 0, elements of the input data in the present instance that are changed from the last instance as well as those that remain the same are determined.

An Attention Focusing Module (**AFM**) would be required as the whole range of input data available to the system would be too much for it to assimilate (although it may need to do something like this at times). The torrent of data would delay its processing significantly and would also probably exhaust its total memory capacity quite rapidly. That is, of course, provided that the system is able to make sense of the data. Given such a huge jumble of data, the system would find it almost impossible to extract any information of sense. The Attention Focusing Module would allow it to identify anything of importance. The module would also ensure the selectivity of data to be processed.

The Attention Focusing Module: This module identifies those elements of specific sensory inputs that the system should concentrate on out of the massive amount of input data.

An explanation of the processing and storage with an example:

Suppose that three images are sequentially presented for assimilation to the system. The first image is completely black, the second has a red square somewhere to the right, and the third has the red square somewhere to the left. Suppose also that the sound "square" is produced by some external entity during the time in which the first image is presented and then substituted for the second, and that the sound "move" is produced during the time in which the second image is presented and then substituted for the third.

Having nothing to compare against, the Attention Focusing Module sets the focus to nothing and the system ignores the first image, simply storing in full the complete set sensory input values corresponding to that instance. In the second image, the Module identifies those pixels belonging to the red square as points of focus because these pixels are changed in value from the previous instance. The system then saves absolute information regarding those pixels as a property set. It also saves another property set - this one just comprising information about the shape that the identified pixels make up (relative position). That is, only the positions (relative to the first red pixel) of the red pixels will be stored.

The Attention Focusing Module also enables the system to track an object that was present in the previous instance by searching for and adding the pixel positions and shape of that object to the set of input data elements to be focused on. This is especially useful when an object persists over several instances but does not move at all. Without this “persistence” feature the object would have been completely ignored because over the static instances there would have been no pixel changes of the color of the object (assuming there are no other unrelated pixel changes of the same color). Here however, there were no objects detected in the first instance and therefore no additional focus is created for the second.

Other property sets arrived at will be with respect to those pixels that are not declared to be the focus of attention, although the processing of these will have a lower priority than the processing of those property sets derived from the focus of attention. Here, these will describe the black background. In fact, in all cases those data from input sensors that are not concentrated upon will be treated as the background (although even the background will have property sets and may contain objects). The system will also arrive at property sets corresponding to the auditory data of the sounds "square" and "move". For as many different sensory inputs there are, the Attention Focusing Module will do its job and direct the system to produce property sets.

When the third image is presented to the system, certain property sets that already exist in the system are encountered. Records that describe any transformation between the two occurrences of the repeated property set are newly created and are saved as "concepts". These are apart from the standard property set determination processes that are described before.

The data manipulation involved in the presented example:

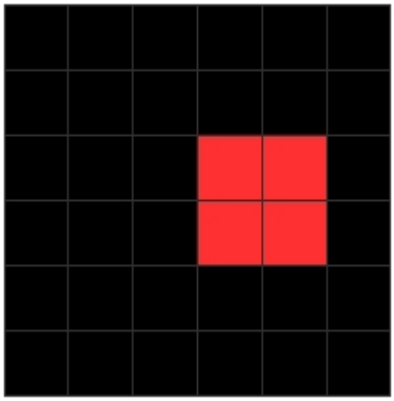
We take the images to be a 6x6 ones, with all pixels in the first image set to the color black.

At Instance 1 >> The system is “born”. The visual is blank and the audio buffer is empty. None of the “video instance 1” records **V1.x** or the “audio instance 1”

records **A1.x** exist (where x is a whole number representing the number of unique property sets registered for instance '1').

At Instance 2 >>

The second image (Figure 2.1) and the sound "square" (Figure 2.2) are presented to the system as shown in Diagram 2 below.



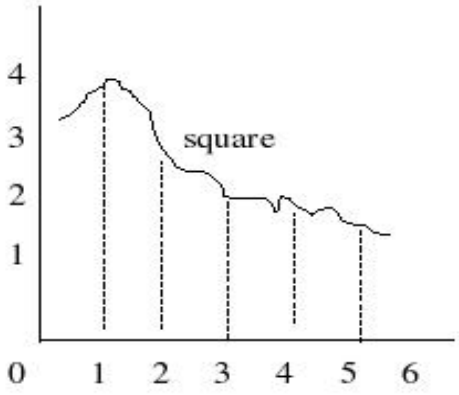


Diagram 2: Figure 2.1 – Visual Instance 2 Figure 2.2 – Audio Buffer for Instance 2

(I have used just numbers, not any specific units, to represent the magnitudes of the frequency on the y scale and the time on the x scale)

The Attention Focusing Module sets the focus on those pixels comprising both the red square as well as the black background, as pixel changes from the last image involve both the colors red and black (some pixels that were black before are red now). Since the entire screen is made up of either black or red pixels, the Module directs that the colors be focused on separately. This allows the square to be distinguished from the background. The visual property sets thus identified by level 0 will be:

V2.1 The exact square: {(16,4), (17,4), (22,4), (23,4)}. If the number '4' represents the color red and if the first pixel, relative to which all other pixels are assimilated, is the left topmost pixel of the video display, then this is the resulting property set that is saved. The first number in every set of round braces indicates the pixel position while the second indicates the pixel color. The pixels are numbered sequentially row-by-row. Thus the first contains the pixels 1 through 6, the second row contains the pixels 7 through 12, and so on. In the diagram above, the first red pixel appears at position 16, followed by the ones at positions 17, 22, and 23.

V2.2 The shape of the square: {(0, 1, 6, 7)}. Here, just the relative positions are considered, color being immaterial to the shape. Relative position = Absolute position - Absolute position of first red pixel. Thus, relative to the first one, the red pixel positions are (16-16=) "0", (17-16=) "1", (22-16=) "6", and (23-16=) "7".

V2.3 The exact background: {(1,0), (2,0), (3,0), (4,0), (5,0), (6,0), (7,0), (8,0), (9,0), (10,0), (11,0), (12,0), (13,0), (14,0), (15,0), (18,0), (19,0), (20,0), (21,0), (24,0), (25,0), (26,0), (27,0), (28,0), (29,0), (30,0), (31,0),

(32,0), (33,0), (34,0), (35,0), (36,0)}. The number 0 represents the color black. The pixels 16,17, 22, and 23 are taken by the red square.

V2.4 The shape of the background: {(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 18, 19, 20, 21, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35)}. The first pixel in the background is pixel number '1' and all the pixels (including the first) are taken relative to it such as: (1-1=) "0", (2-1=) "1" ... (35-1=) "34", and (36-1=) "35".

The audio property sets identified by level 0 will be:

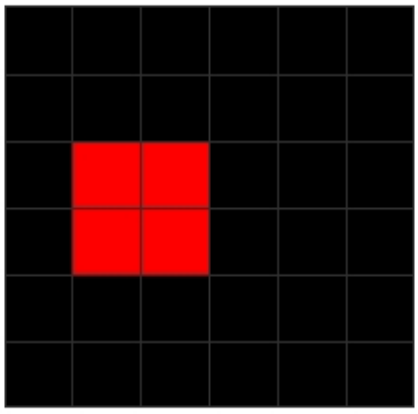
(The check for a sound already encountered would fail as these sounds are being presented to the system for the first time).

A2.1 The exact sound "square": {(3.5, 2.5, 2.1, 2.0, 1.8)}. These are the frequency values corresponding to the end of the time intervals 1, 2, 3, 4, and 5, as shown.

A2.2 The pattern of the sound "square": {(1, 1.4, 1.67, 1.75, 1.94)}. Since humans analyze sounds logarithmically, it seems better to take the relative ratios in the case of audio. Assuming the first frequency value, against which all others are to be compared, is the one occurring first with respect to time, this is the resulting property set that is saved. The absolute frequency values at the end of the time intervals 1, 2, 3, 4, and 5 are 3.5, 2.5, 2.1, 2.0, and 1.8 respectively. Relative to the first, the frequency ratios would thus be (3.5 / 3.5 =) 1, (3.5 / 2.5 =) 1.4, (3.5 / 2.1 =) 1.67, (3.5 / 2.0 =) 1.75, and (3.5 / 1.8 =) 1.94.

At Instance 3 >>

The third image (Figure 3.1) and the sound “move” (Figure 3.2) are presented to the system as shown in the Diagram 3 below.



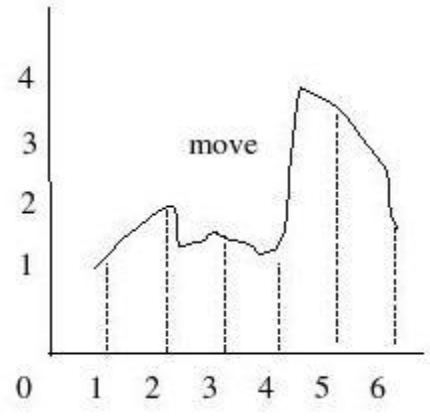


Diagram 3: Figure 3.1 – Visual Instance 3 Figure 3.2 – Audio Buffer for Instance 3

(I have used just numbers, not any specific units, to represent the magnitudes of the frequency on the y scale and the time on the x scale)

Attention is still focused both on the red square as well as on the black background due to the color changes caused by the moving square covering up some black with red and presenting some new red over black. The Attention Focusing Module also specifies separately that the pixels comprising any object with the same shape and color as the square encountered before needs to be focused on, however it finds that these are already in focus. This will give the following visual property sets:

V3.1 The exact square: {(14,4), (15,4), (20,4), (21,4)}.

V3.2 The shape of the square: {(0, 1, 6, 7)}. This is the same as before. Invariance in shape could be used for the identification of static relative orientation in terms of cyclical rotation in unrestricted dimensions (x, y, z, ... planes) as well as distance (moving closer would cause the image to seem enlarged).

V3.3 The exact background: {(1,0), (2,0), (3,0), (4,0), (5,0), (6,0), (7,0), (8,0), (9,0), (10,0), (11,0), (12,0), (13,0), (16,0), (17,0), (18,0), (19,0), (22,0), (23,0), (24,0), (25,0), (26,0), (27,0), (28,0), (29,0), (30,0), (31,0), (32,0), (33,0), (34,0), (35,0), (36,0)}

V3.4 The shape of the background: {(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 16, 17, 18, 19, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35)}.

The audio property sets identified by level 0 will be:

A3.1 The exact sound "move": {(1.5, 2.2, 1.7, 1.5, 3.3, 2.0)}. Here six frequency values have been taken as opposed to five for the last sound. This is because is possible that a sound will not consume all available buffer space corresponding to a particular visual instance, and after the sound ends there is conceivably silence thereafter until the end of that buffered stream. Considering this example, the sound “square” as shown in the diagram is seen to take up 5 out of 6 units of buffer space whereas “move” takes up all 6.

A3.2 The pattern of the sound move: {(1, 0.68, 0.88, 1, 0.45, 0.75)}. Relative to the first, the frequency ratios are (1.5 / 1.5 =) 1, (1.5 / 2.2 =) 0.68, (1.5 / 1.7 =) 0.88, (1.5 / 1.5 =) 1, (1.5 / 3.3 =) 0.45, and (1.5 / 2.0 =) 0.75.

2.3: Objects and Concepts in the Gen-I-Sys machine

Property sets resulting from different sensory inputs are combined to form ‘objects’ that are then passed on to level 1. Level 1 processes the relationships existing between these objects (if any) and sends these ‘concepts’ on to level 2. Similarly, advanced implementations of this AGIS could involve the extraction of relationships between the concepts themselves to produce ‘hyper-concepts’ that are more abstract.

Objects

An object consists of a visual property set separated by a field separator ("||", for instance) from the auditory, olfactory, etc. property sets associated with that visual set. Several objects may result from different combinations of the visual, auditory, etc. property sets discovered in an instance.

With regard to the example that was being discussed, the objects formed here are:

For Instance 2 -

O2.1 = {V2.1 || A2.1, A2.2}
= {(16,4), (17,4), (22,4), (23,4) || (3.5, 2.5, 2.1, 2.0, 1.8), (1, 1.4, 1.67, 1.75, 1.94)}

O2.2 = {V2.2 || A2.1, A2.2}
= {(0, 1, 6, 7) || (3.5, 2.5, 2.1, 2.0, 1.8), (1, 1.4, 1.67, 1.75, 1.94)}

O2.3 = {V2.3 || A2.1, A2.2}
= {(1,0), (2,0), (3,0), (4,0), (5,0), (6,0), (7,0), (8,0), (9,0), (10,0), (11,0), (12,0), (13,0), (14,0), (15,0), (18,0), (19,0), (20,0), (21,0), (24,0), (25,0), (26,0), (27,0), (28,0), (29,0), (30,0), (31,0), (32,0), (33,0), (34,0), (35,0), (36,0) || (3.5, 2.5, 2.1, 2.0, 1.8), (1, 1.4, 1.67, 1.75, 1.94)}

O2.4 = {V2.4 || A2.1, A2.2}
= {(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 18, 19, 20, 21, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35) || (3.5, 2.5, 2.1, 2.0, 1.8), (1, 1.4, 1.67, 1.75, 1.94)}

For Instance 3 -

O3.1 = {V3.1 || A3.1, A3.2}
= {(14,4), (15,4), (20,4), (21,4) || (1.5, 2.2, 1.7, 1.5, 3.3, 2.0), (1, 0.68, 0.88, 1, 0.45, 0.75)}

O3.2 = {V3.2 || A3.1, A3.2}
= {(0, 1, 6, 7) || (1.5, 2.2, 1.7, 1.5, 3.3, 2.0), (1, 0.68, 0.88, 1, 0.45, 0.75)}

Since the visual property set for O3.2 matches that for O2.2, the record for O2.2 is updated to include all other sensory data associated with V2.2 and is renamed to O3.2, with O2.2 itself being deleted.

O3.3 = {V3.3 || A3.1, A3.2}
= {(1,0), (2,0), (3,0), (4,0), (5,0), (6,0), (7,0), (8,0), (9,0), (10,0), (11,0), (12,0), (13,0), (16,0), (17,0), (18,0), (19,0), (22,0), (23,0), (24,0), (25,0), (26,0), (27,0), (28,0), (29,0), (30,0), (31,0), (32,0), (33,0), (34,0), (35,0), (36,0) || (1.5, 2.2, 1.7, 1.5, 3.3, 2.0), (1, 0.68, 0.88, 1, 0.45, 0.75)}

O3.4 = {V3.4 || A3.1, A3.2}
= {(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 16, 17, 18, 19, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35) || (1.5, 2.2, 1.7, 1.5, 3.3, 2.0), (1, 0.68, 0.88, 1, 0.45, 0.75)}

This specific representation scheme places the other sensory data with respect to visual data. That is, the object format becomes:
Visual Object > sounds, sound patterns, other sensory data (all that are ever associated with that visual object).

Visual Shape > sounds, sound patterns, other sensory data (all that are ever associated with that visual shape).

This is because in this example and in most learning situations as we know them, it is the visual data that recurs the most. Should a sound or a sound pattern ever recur, the respective object formats would be drawn up:
Sound > visual objects, visual shapes, other sensory data (all that are ever associated with that sound).
Sound Pattern > visual objects, visual shapes, other sensory data (all that are ever associated with that sound pattern).

At level 1, therefore, the system has a growing collection of objects, each of which is made up of grouped sensory data or property sets. Any specific sensory data or sensory data pattern picked up by the system is thus linked to all other data or patterns encountered at the same instant. From the data collected as per the example being discussed, the system can technically infer the following (after the three instances are presented to it):
1. The sound or sound pattern of "square" could refer to the red square, the shape of the square, the black background, and the shape of the background.
2. The sound or sound pattern of "move" could refer to the red square, the shape of the square, the black background, and the shape of the background.

Only some of these inferences can actually be correct in the real world, as the system will learn from repeated associations. As more and more sounds are associated with the image or shape of the red square, those associations that occur again and again will increase in strength of correlation to the square, and these will probably be sounds like "red", "square", etc. In memory, there must be an index to identify this. Similarly, the system will learn that the black background is not referenced by either the sounds "square" or "move", other sounds such as "black" will hold much stronger correlations to it than any other irrelevant sounds. Additionally, the processing taking place would involve the identification of similarities and differences in property sets that occur consecutively on more than one instance. These similarities and differences form the basis of "concepts".

Concepts

Concepts represent relationships between instances in terms of the introduction, disappearance, persistence, physical translation, growth, shrinkage, etc. of objects. In fact, many types of changes involving objects can be identified with the use of concepts. The identification of some basic changes must be preprogrammed into the AGIS before it is able to interrelate elements of this basic set for the derivation and identification of more complex object transformations or "hyper-concepts".

In the example being discussed, the square is introduced into the system's visual field after which the square is displaced leftwards (and, if more instances had been considered, might have been made to continue leftwards until it disappeared from the screen). Some of the basic changes that need to be specified to the AGIS are 'introduction', 'physical translation' (whether horizontal or vertical, leftwards or rightwards), and 'disappearance'. When the square suddenly appears a new object is registered by the AGIS. The various property sets associated with the new object may be associated with a predefined concept of "introduction". It is initially difficult to see how this could be of any practical use (a characteristic shared by several aspects of the Gen-I-Sys engine). If we consider those images, ideas, and thoughts that are brought to

our own minds upon hearing or reading the word "introduction" then we may see how the categorization of property sets into different concepts may help. Usually this word would bring up in our minds any or all of the following: "new", "starting the use of", "beginning", "first social meeting", "incorporating within", etc. Now consider this: the Gen-I-Sys engine functions primarily by first generating an Action Sequence that takes it from its current state to a goal state and then "executing" everything that can be executed in that Action Sequence (see the later section 'How outputs are produced in the Gen-I-Sys machine'). Thus, if any of these thoughts or ideas were verbally pronounced during the introduction of the square then the property sets for these sounds would be correlated to the concept of introduction. If the introduction of an object were to fall in one of the system's Action Paths (just like the introduction of a feeding bottle would fall in a hungry baby's action path) then upon retracing the path the system would "execute" or produce the waveforms for the associated sounds. Not only does this help the system to understand what the sounds mean (in terms of associated images and transformations in the images) but it also helps the system to take a first step in spoken language.

Similarly, whenever the Attention Focusing Module indicates the continued presence of an object, the associated property sets may be incorporated into a concept of "persistence". If movement of an object were detected then the concept category would be "physical translation" with perhaps sub-concepts existing for the left, right, up and down directions of motion. When the AFM searches for a pattern match with a previously existing object (in order to identify persistence) and finds none then the data associated with that object is appended to the "disappearance" category.

With reference again to the earlier example, when the system detects that the exact square has persisted across instances 2 and 3, it searches within records of the visual property sets of V2.1 and V3.1 to find the absolute position of the square at both instances. The displacement of the square can thus easily be determined by finding the difference between the coordinates of any point of the square before and after the translation:

Displacement, **D** = (14-16 = 15-17 = 20-22 = 21-23 =) -2

Therefore, **C-Translation-3.1** = {D | A3.1, A3.2}
= {(-2) | (1.5, 2.2, 1.7, 1.5, 3.3, 2.0), (1, 0.68, 0.88, 1, 0.45, 0.75)}

Since the word "move" was registered by the system across this displacement, it can technically infer the following:
1. The sound or sound pattern of "move" refers to a displacement of –2 and to the concept of "physical translation".

2.4: How Outputs are produced in the Gen-I-Sys machine

The Gen-I-Sys engine functions primarily by first generating an Action Sequence that takes it from its current state to a goal state and then "executing" everything that can be executed in that Action Sequence.

For outputs such as motion to occur, the system would have to be equipped with sensors telling it what mechanical movements it is going through. Given this, outputs become recollections of pre-encountered movements that the system is

able to trigger off itself. As the intended functioning of the system is best explained through examples, another one will be presented here. Although in the earlier example only audio-video associations are considered, this one will consider more associations (video, audio, motion, and internal state).

Suppose the system at its 'birth' is sitting in the middle of an empty room. The room has a power outlet on one wall. This power outlet is covered so that although the system might 'recharge' itself from it, it would need to call upon the help of a human (who will also be standing in the room) to remove the cover and give it access. The system will initially start out with full power, which will drop steadily with time so that it is necessary for it to recharge in order to keep running. The main rule guiding the system is 'never let the power reach 0, and aim to trigger outputs that will maximize it'.

Sitting in the middle of the room, the system's power continues dropping. After some time, the human approaches, pushes the system towards the power outlet, removes the cover, and plugs the system in. The system has continuously been monitoring all sensory data (exact data as well as patterns), and it now detects a surge in its power level. After it is completely recharged, the human unplugs it, pushes it back to the middle of the room, and leaves it there. The next time the system senses a low power level, it searches its memory for an instance in which its power increased. It finds this instance in the form of a visual image of the power outlet literally 'pressed against its face'. It then traces back until it finds an instance corresponding to its present situation, and simply triggers off those same motions that it sensed it was going through while being pushed by the human.

Just as in the example presented earlier, in which the system noted sensory data corresponding to the transformation of (-)2 of the square (in our terms: leftward movement of the square), the system will here also know which sensory data caused which of the transformations that led from its position in the center of the room to its position at the power outlet. The sensory data that are associated with those transformations are the movements of its, say, wheels. These movements, which were recorded before, are triggered off now, and the system rolls towards the power outlet. The sequence of events is shown below:

Image:	---	A (center of room)	B	C	D (at power outlet)
Power Level:	---	Low	Low	Low	Increasing
Motion:	---	Forward	Forward	Forward	None

Teaching the system to speak would be rather similar. The human would in this case, perhaps, lasso the system and pull it towards him/herself. When the system reaches the human, the human would maybe say the word 'food' and would then pull the system to the power outlet and plug it in. Given this, the next time the system requires recharging it would trace back events from power rise to power low and would execute those outputs corresponding to those instances in reverse temporal order. The system would thus roll towards the human, produce the waveform (or 'say' the word) "food", and would then roll towards the power outlet (if the human does not push it around to the contrary). Speech in the system is programmed as the production (‘execution’) of all waveforms on an instance path leading from a low to a high state.

It is level 2 that tracks and stores those sets of instances leading from low to high states. In the above example, all the instances involved in rolling to the wall (or in first rolling to the human, producing the sound, and then rolling to

the wall), are identified by level 2 and sent to level 3 for storage. Thus, level 3 contains macro or compound instance sets that actually are high order or abstracted concepts that let the system carry out a relatively complex series of activities in order to reach a specific goal and with a specific intent. These abstractions could form the basis of plans, predictions, or the memory of plans and predictions. (The system could be said to 'remember' an instance, an object or a property set whenever one of them is being processed by it.) As it is, following this framework, the AGIS demonstrates reasoning (see the Reasoning section) although at a very rudimentary level and although purely based on retracing the actions that it had earlier observed itself being put through. Through 'associative reasoning' it establishes the fact that being near the power outlet (i.e. analogous to having a large view of the power outlet, implying lesser distance) may cause its power levels to go up. Through 'deliberate reasoning' it establishes the chain or sequence of actions that will take it from its current position to its desired position.

If we ignore for the moment the fact that we are talking here about a computer, then we get:

... an entity that is shown the way to food once, twice, thrice ... and subsequently knows how to go and get the food for itself. Could we be talking about anything other than a human child?

For a fuller description of the Gen-I-Sys design and how it works please contact the author.

2.5: Further Issues

The Gen-I-Sys model may be enhanced by recursively clustering groups of neuronal structures together [9], with each individual structure being responsible for the identification of one specific mini-feature out of the entire input data set. Similar to the Recognition Cones, we could thus have one or more structures solely devoted to the identification of edges, with another set devoted to the identification of displacements of those edges when moving from one instance to the next. If we cluster together color recognizing structures with the edge-detectors and break up the displacement-recognizing category of structures into two further sets that deal with vertical and horizontal displacements respectively, then we have a reasonably complex Visual Module. This module could correlate directly to the visual cortex of the human brain. Further improvements may be made to the existing visual and auditory modules by considering works including and beyond research done on visual object movements [7] and research done on phonetics and word boundary identification [15] respectively as well the research focusing on contextual evaluation and interpretation of sensory input signals [8].

Bibliography:

[1] Battaglia, P. W., Jacobs, R. A. and Aslin, R. N. (July 2003) "Bayesian Integration of Visual and Auditory Signals for Spatial Location" Optical Society of America Vol 20

[2] Blamire, J. (2000) "Genotype and Phenotype"
http://www.brooklyn.cuny.edu/bc/ahp/BioInfo/SD.Geno.HP.html

[3] Bush, P. (June 1996) "How the Brain Works"
http://www.keck.ucsf.edu/~paul/brain.html

[4] Dreyfus, H. L. and Dreyfus, S. E. (2000) "Mind over Machine" Free Press 01 March, 2000 Paperback ISBN: 0743205510

[5] Ernst, M. O. and Banks, M. S. (24 January 2002) "Humans Integrate Visual and Haptic Information in a Statistically Optimal Fashion" Nature Vol 415

[6] Ernst, M. O. and Bülthoff, H. H. (April 2004) "Merging the Senses into a Robust Percept" TRENDS in Cognitive Sciences Vol 8 No 4

[7] Feldman, J. and Tremoulet, P. D. "Individuation of Visual Objects over Time" Technical Report #74, Rutgers University Center for Cognitive Science

[8] Herzog, M. H. and Fahle, M. (24 January 2002) "Effects of Grouping in Contextual Modulation" Nature Vol 415

[9] Honavar, V. and Uhr, L. (1989). "Brain-Structured Connectionist Networks that Perceive and Learn".
Connection Science 1: 139-160.

[10] Honavar, V. and Uhr, L. (September 14, 1993) "Toward Learning Systems That Integrate Different Strategies and Representations" TR93-22
Department of Computer Science, Iowa State University of Science and Technology

[11] Kersten, D. and Yuille, A. (2003) "Bayesian Models of Object Perception" Current Opinion in Neurobiology CONEUR 30

[12] Körding, K. P. and Wolpert, D. M. (15 January 2004) "Bayesian Integration in Sensorimotor Learning" Nature Vol 427

[13] Koushanfar, F., Slijepcevic, S., Potkonjak, M., and Sangiovanni-Vincentelli, A. "Error-Tolerant Multi-Modal Sensor Fusion (Short Paper)"
http://dsp.jpl.nasa.gov/cas/short/slijepcevic.pdf

[14] Malkine, A. "Philosophy of the Artificial"
http://www.rpi.edu/locker/48/000848/yesterday/aron/_thought/ai.html

[15] Mattys, S. L. and Jusczyk, P. W. "Do Infants Segment Words or Recurring Contiguous Patterns?" Journal of Experimental Psychology: Human Perception and Performance
June 2001, Vol. 27, No. 3, 644-655
http://www.apa.org/journals/xhp/xhp273644.html

[16] Sober, S. J. and Sabes, P. N. (August 6 2003) "Multisensory Integration during Motor Planning" The Journal of Neuroscience 23(18):6982-6992