Encyclopedia of Library and Information Science (ELIS), vol. 68, pp. 346-366, 2001.

# AN UPDATED APPROACH TO COMPLEXITY FROM AN AGENT-CENTERED ARTIFICIAL INTELLIGENCE PERSPECTIVE

Oscar N. Garcia
Dept. of Computer Science and Engineering
Wright State University
Dayton, OH 45435
Ph.: (937) 775-5134
Fax: (937) 775-5133
ogarcia@cs.wright.edu

Ricardo Gutierrez-Osuna
Dept. of Computer Science and Engineering
Wright State University
Dayton, OH 45435
Ph.: (937) 775-5120
Fax: (937) 775-5133
rgutier@cs.wright.edu

## ABSTRACT

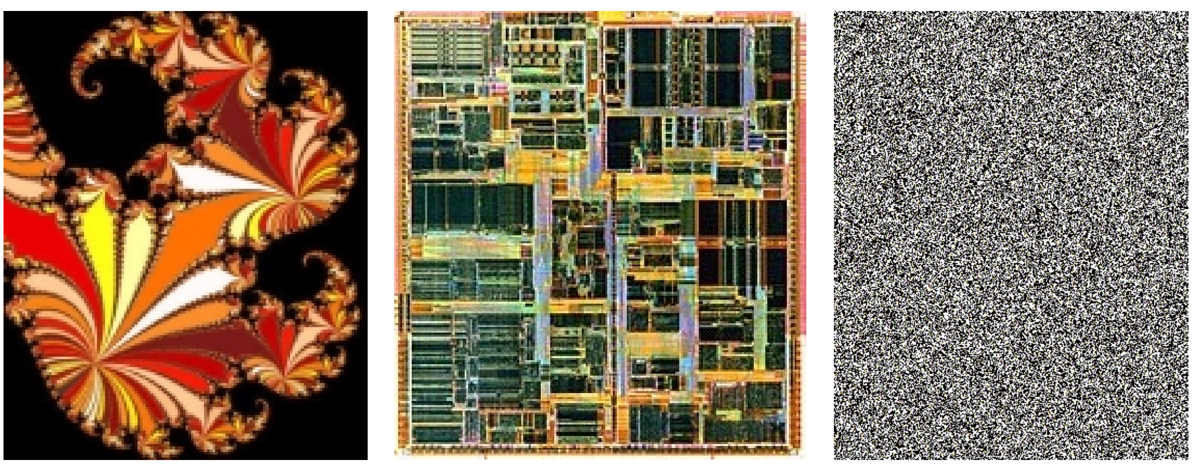
Here we update and expand our previous work to find commonality in a variety of systems, situations, and organisms with regard to a generic concept of complexity (1). Such attempts have drawn significant attention from outstanding researchers of diverse backgrounds (2). Despite years of research with many papers written, there is yet to appear a convergence toward a unified methodology in the multi-disciplinary approaches. In particular, we consider how the synthesis of complex systems, as is practiced in the software engineering of large systems, sheds light on the analysis of complexity. We consider how levels of abstraction at different granularities relate to complexity from the point of view of the description of a model of the system. We consider also the relationship of an intelligent agent, symbolic or biological, with the estimation of complexity. We use the term *agent* to include both intelligent programs and humans, which learn systems (models) from observations and adapt to their environment by learning. It is suggested here that an agent-centered approach, factoring complexity modulo a context (called in Artificial Intelligence a *perspective*), may provide a more general approach to determining the complexity of an object or system. The thesis presented here is that the object must not be considered independently from its user agent and its context and that in the duality of the consideration, one may learn more about both the general characteristics of complexity and of the specific reasons for our interest in it. The proposal is to consider user, interaction, and the related interfaces simultaneously when studying the complexity of a system. This approach is more integrative and less reductionist than studying any of the system aspects individually and bridges the difficulties of a large range of diverse settings. An overview of the field is briefly presented in light of the proposed approach.

## 1. INTRODUCTION

On the surface, what we call complexity may be simply related to the number of parts, elementary actions, or sequences of actions when we apply the commonly held reductionist view to objects, systems, or events. It would seem that anything with a million components should be complex. However, more detailed considerations indicate that the functional connectivity of the parts, their similarity or dissimilarity and even the point of view from which they are considered by a potential user or participant must be considered.

The field of artificial intelligence has classically coped with complexity by invoking hierarchical abstraction. Using this method, the same object, action, or event is considered at different levels of detail, with higher levels of abstraction having fewer directly observable features. In engineering design, this idea is applied more functionally to breaking down a complex system into simpler subsystems, which are assembled together. These have been very useful approaches, particularly in production, because testing and finding errors in the subsystems is much less expensive than doing the same in the completed module or final assembly.

However, when we consider natural systems that change and evolve in space and time, such as ecological systems, planetary systems, or even biological organisms, our complexity paradigm needs to be expanded further. Ideally, we would like a measure that indicates the complexity of the system in the abstract, without regard to its environment or the point of view of a user or an observer. We claim, however, that complexity needs to be determined relative to an *agent* (a system with sensors, actuators, and an internal structure of variable complexity, in many cases capable of adaptability and learning) from a given perspective, or potentially averaged over a class of *agents* with the same perspective. Planetary motion, gravitational pull, and the high acceleration of particles were considered very complex before the discoveries of Kepler, Newton, and Einstein. Science attempts to find general explanations that are not observer dependent. Its purpose is to reduce the complexity of these systems in the eyes and minds of those trained in that kind of scientific observation. However, the same specific system shows different complexities at different levels of detail and depending on what aspects are sensed, as well as the bias, interest, knowledge, and understanding of the observer or user of the system. As we build more sophisticated artificial systems and move toward a science of the artificial in the computer field, it is imperative that the issues of complexity be considered agent-centered, based on the model of the system the interacting agent has managed to acquire.



(a) (b) (c)

Figure 1. Intrinsic Vs. subjective complexity:
(a) Mandelbrot set, (b) CPU layout and (c) white noise

Consider the images in Figure 1, which show a segment of the Mandelbrot set, a VLSI layout for a microprocessor and an image filled with salt-and-pepper noise. If we were to measure the *intrinsic* complexity of these objects by the length of the shortest program that could generate them (the Kolmogorov complexity), we would obtain the counter-intuitive result that the noisy image is most complex. Whereas the Mandelbrot set can be generated from the compact recursion $z_{N+1} = z_N^{\ 2} + c$ (with initial condition $z_0 = c$) and the CPU layout can be hierarchically described in terms of computer-architectural building blocks, the shortest program that *exactly* describes the noisy image is the image itself. From our standpoint, the fundamental flaw of such a measure of complexity (other than the association of complexity with randomness), is that it attempts to quantify the *intrinsic* complexity of an object, instead of the *subjective* or user-centered complexity.

In this update article we propose an approach for measuring complexity —and selecting an appropriate level of abstraction for a given context and modality— that is based on the Minimum Description Length (MDL) Principle and the related Stochastic Complexity of Rissanen (3). The proposed metric quantifies the complexity of a system by the length of the shortest model that describes the regularities of the system, as perceived by an agent operating from a given perspective. This measure of complexity is related to what Gell-Mann (12) calls *effective complexity*: the length of a schema (or approximate model) used to describe the regularities of a system, relative to an agent that is observing it. Hence, unlike the incomputable Kolmogorov complexity, which focuses on an *exact* description of an object, the proposed metric is aimed at an *approximate* (read subjective) model of an object.

## 2. OVERVIEW

In this section we will introduce background material that will assist us in deriving a user-centered approach to measuring complexity. These are: (1) the concept of perspectives in Artificial Intelligence, (2) essential and accidental attributes of complex system, (3) a summary of the multiple views and measures of complexity and (4) the software engineering approach to measuring complexity.

### 2.1. PERSPECTIVES AND CONTEXT IN ARTIFICIAL INTELLIGENCE

We first describe the origins of the concept of *perspectives* within artificial intelligence (AI). The field of AI has struggled for many years (4) with the central issue of how to represent knowledge. In particular, the oft-referenced concept of frames (5) has been the basis of a variety of formalisms for representation. It was the basis for the ambitious *Knowledge Representation Language* (KRL) proposed by Bobrow and Winograd (6) of failed implementation but very fruitful in providing a solid theoretical framework. We borrow the concept of *perspective* from KRL. The representational frame approaches have recently coalesced in models of natural language and cognition. See, for example, the work of Rasmussen (7), pp. 28-30 on *perspectives* and on functional and structural modeling of adaptive systems. The KRL approach may have been overtaken by the improved inferencing of other logic models in AI, but it had higher expressive power. The KRL representations are based on the comparison of prototypical entities (*perspectives*) rather than through a single definition or set of constraints on the object. We propose here that the *perspectives* on the complexity of a system/object/event be based on the agent's model and the interaction among the complex object, its context, and the (possibly human) agent. As an example of how a *perspective* may change the value of the properties of an object, think of an everyday automobile that may be considered fast in the context of a means of transportation but slow in the context of a race of high-performance vehicles. Clearly, the semantics of the *perspective* influences the relative complexity of the frame.

The concept of *view* or *subschema*, present in many but not all database management systems, is an abstract model of the conceptual database or conceptual scheme by means of subscheme data definition and data manipulation languages. The definition of a *view* is obtained by manipulations on part or all of one or more base tables and is, therefore, a derived virtual definition or a manipulated projection of base tables. Views may be involved in searches using languages like the Structured Query Language (SQL), which may serve particular groups of users (8). This is a more concrete implementation than the representation by *perspectives* in KRL. Another explicit use of context occurs when help-menus interfaces, sensitive to current settings, offer applicable assistance to the user.

Methodologies modeling user interactions as finite-state machines or Petri nets of subtasks in different contexts have been used to express them in abstract hierarchies, including both physical and cognitive actions. The "operator functional model" of Jones and Mitchell (9) is one such example. Other approaches, such as SOAR, originally oriented to providing an architecture for general problem-solving methods (10), have also been used to unify diverse cognitive representations.

## 2.2. ACCIDENTAL AND ESSENTIAL ASPECTS OF COMPLEXITY

In a frequently quoted article related to software engineering, Brooks (11) referenced the fact that the state of the art had not advanced significantly in that field through the discovery of *essential* truths, but mostly through tackling *accidental* issues that have yielded improvements on productivity, reliability, and so forth, without providing fundamental guidelines. He refers in that article to the lack of a "silver bullet" to slay the monsters that stand in the way of efficiently producing better software. In a sense, this is also reflected in the search of Nobelist Murray Gell-Mann (12) to find the connectivity between laws general to all entities in the universe (the quantum mechanics of quarks, for example) and accidentally conditioned laws such as those that produce the evolution of entities in complex systems (like jaguars in our ecosystem). How the aggregation of simple parts (the quarks in the jaguar!) subject to universal laws combine in such precise layered constructs in space and time to produce the complex adaptive systems of nature is, indeed, a basic and, so far, inconclusive scientific quest.

In dealing with individual variations between specific members of a set of observations, we often categorize them by deciding on a given level of granularity and cluster them around a centroid in some externally conceived metric space. This centroid is an idealized averaged observation, which we choose as representative of the others included in the cluster. In this manner, we deal with *accidental* variations of features while preserving the *essential* personality of the group. This is the manner in which we cope with the variants of a given gene and the phones of a given phoneme. It is interesting that in the biological world, the preexistence of alleles in populations of bacteria, which signifies genetic diversity, allows adaptive resistance of external attack by antibiotics through the survival of those individuals most fitted for the changed environment. The closest analogy in the software production environment is the reliability improvement offered by the creation in isolation of *N* program versions that follow the same program specifications and whose results are compared in a majority-voting discipline as done in the pioneering experiments of Brilliant, Knight and Leveson (17). Also, the stochastic approach that we suggest, following Rissanen, to choose a level of model abstraction for optimal use is based on the combination of the "regular" attributes of an object (essential?) plus the deviation or "error" of our data for modeling the object (accidental factors?).

## 2.3. VIEWS AND MEASURES OF COMPLEXITY

For as long ago as when eighteenth-century theologian William Paley used the argument that just as a watch was too complex to have occurred naturally—and therefore all living things must have been deliberately and purposely designed – the relationship between life and complexity has been discussed. More recently, Dawkins (22), an ardent defender of Darwinism, has characterized complexity as having the fundamentally necessary condition that we call extreme heterogeneity: something that has many diverse parts. But Dawkins immediately admits to the insufficiency of that condition and restricts his definition to cases where the assemblage of the diverse parts of the complex entity is unlikely to have arisen merely by chance. More is indeed needed, because while each batch of a large number of diverse parts thrown together is unique and possibly unrepeatable, we do not consider it complex, as we would consider complex a constructed operational airliner. The argument that Dawkins presents then is that the large number of diverse parts put together must have some rational quality, specifiable in advance, that is highly unlikely to occur by change alone. In other words, there are interactions between the parts assembled that can be understood by what Dawkins calls "hierarchical reductionism". This is not reductionism down to the smallest components, but rather one that explains behavior in terms of the upper or lower corresponding levels in the hierarchy. This article owes much to this concept of complexity: Both theses assert that complexity must take into account the constitution and function of the complex entity as perceived by an agent (human or whatever) judging complexity in context.

Another view of complexity is held by Gell-Mann, one of the founders of the Santa Fe Institute, which is dedicated to understanding the behavior of complex systems. Gell-Mann (12) considers two types of complex system: those with the ability to evolve and learn, and those without it. The airliner of Dawkin's example will satisfy the latter but the former is capable of adaptive behavior. The term "Complex Adaptive Systems" (abbreviated CAS) describes this most interesting class of complex systems with biological systems as obvious representatives. However, lest anyone thinks that non-biological systems do not qualify, survey the abundant literature on artificial life (23) and even a recent book by Dyson (24), which tackles the complexity of dealing with non-biological evolution using machines as evolving systems in the peculiar ecosystem context of human designers and users.

We are indebted to Lloyd (25) for an accounting of some 31 definitions of complexity. Among those 31 definitions of complexity isolated by Lloyd, the Finmeccanica Assistant Professor of Mechanical Engineering at MIT, we will elaborate briefly on some mostly selected by Horgan, a past senior writer at *Scientific American* (26). These complexity measures are based on the following:

- *Time and Space Complexity:* These are perhaps the complexity measures best known among computer scientists and constituted an early basis to study the complexity of algorithms. Fundamentally, these measures tell us the rate of growth of an algorithm in either execution time or memory requirement, with the size or dimensions of the problem tackled. They constitute a significant part of the theory of algorithms. The measure is given by the most significant term of the order of growth, called *O(n)* for a problem of size *n*, as *n* increases.

- *Algorithmic:* A different view, also from a computer science perspective, is based not on the complexity of the object we are dealing with but with the algorithm which, when interpreted, describes that object. Ideally, this measure (also known as algorithmic randomness) is invariant with an abstract interpretation process. This approach was advanced independently in the sixties by young Chaitin and also by Kolmogorov and Solomonoff. It has evolved into a field called algorithmic information theory (27). We will refer to this approach in the following section.
- *Entropy:* Thermodynamically inspired measures are based in principle on the energy unavailable to be transformed into work due to the motion of molecules. It is directly proportional to the quantity of heat in the body and vanishes at absolute zero. In an isolated system, entropy cannot decrease with change, but increases during irreversible changes and remains constant in reversible ones. Consequently, the universal entropy continuously increases. Entropy also measures the internal level of disorder of the atoms of a substance; for example, the atoms of a diamond are more orderly than those of helium in a balloon at room temperature; the diamond has lower entropy. The field of information theory utilizes the concept of entropy to describe the regularity of information sources, as we see in the next definition. The formula that represents entropy in information theory, which is the average information of all messages in the set of possible messages, is of identical form to that representing entropy in statistical mechanics.
- *Information:* As described by Shannon, the probabilistic nature of information sources is related to the entropy or irregularity of a system. An information source that is very unpredictable has higher entropy and generates more information than one that is more predictable. The most information may be obtained when all possible outcomes are equiprobable. A measure of information is a measure of unexpectedness. Clearly, one would expect high information sources to be more complex.
- *Fractal Dimension:* Systems that may be described by measures of self-similarity or fractals have varying degrees of relatively invariant (complexity) detail at smaller and smaller scales. In this case, the "fuzziness" of the system is an indication of its complexity.
- *Effective Complexity:* This measure of complexity is related to the regularity, rather than to the randomness, of a system. Complex adaptive systems have the ability of distinguish what is random from what is not by discerning patterns and can, therefore, identify effective complexity. Effective complexity relates to the description of the regularities of a system by the CAS (12) observing it. This is very similar to what we are considering here, as it involves defining the complexity of a system relative to an external agent observing it. *Hierarchical Complexity:* This complexity measure is related to the diversity that is found at the different levels of the hierarchy of a system. Yoon and Garcia (29) have considered experimentally the cognitive complexity of searching the abstraction hierarchies of a program in debugging it.
- *Grammatical Complexity:* Grammars are syntactic descriptions of occurrences of well-formed sentences. The degree of generality of the grammar that describes a system is an indicator of its complexity.
- *Thermodynamic Depth:* Proposed by Lloyd and Pagels, thermodynamic depth is a measure not of the complexity of the object but that of the process that created or

calculated it, obtained by measuring the work done or increase in entropy required as the states change to its current state.

- *Mutual Information:* This information-theoretic measure of complexity is related to the information (or similarities) that one part of the system has with regard to the other parts.

As we can see, there are commonalities among the different measures, but there is not a single commonly accepted measure or unit, as also pointed out by Maddox (30), other than those based on information or statistical thermodynamics. Perhaps, some day, the relations among the different measures will become clearer and then, hopefully, more explicitly correlated. In the meantime, we offer a view of how a stochastic approach to complexity based on the probabilities of a hypothesis and a set of observations sensed by an agent.

## 2.4. SOFTWARE ENGINEERING MODEL OF HANDLING COMPLEXITY

The contemporary approach taken by software engineers in handling complexity is an interesting and important one, which has not yet been extended in methodology to other fields. Although it may not have been considered because it is oriented to engineering synthesis and not to scientific analysis, it is an excellent model from which we can learn about complexity because so much past and current attention has been devoted to it. Brooks has revisited these issues in an anniversary extension (13) of his original essay on software engineering.

A basic tenet of Brooks' report of more than 20 years on his OS/360 design experience is that due to the interrelated nature of a programming effort, adding manpower to a late software effort is likely to be counterproductive. He points out that architectural conceptual design integrity is fundamental, as many hands are needed in the implementation of the software within a schedule. Along with our theme here, Brooks defines the architecture of a system as the complete and detailed specification of the *user* interface, specifically the manual of operation.

We mentioned earlier that the number of parts is a very rough first-order approximation to complexity. In software engineering, the equivalent is the number of lines of code or "locs." In the first edition of *The Mythical Man-Month* included in (13), Brooks reported a hyper-linear relationship between effort in man-months and the total number of locs. Classical programming estimates of average coding productivity are about 10 locs/day. However, averages vary widely. There have been measured variations between 1 and 5-10 in programmer productivity, of about up to a factor of 3 slowdown in coding control code versus translator code, and a noticeable speedup when using high-level languages. Estimates of coding effort are full of uncertainties due to programming environment, the programmer, and the problem domain. However, approximating those factors, the relation

$$Programming\ \ effort = Constant \times \left(total\ \ locs\right)^{1.5}$$

has been established in at least two studies. Interestingly, this correlates well with the power laws suggested by Gell-Mann (12) for information complexity relating to word length (Zipf's law), and so on. From a complexity point of view, perhaps the gains in modular or object-oriented (O-O) programming arise from the shorter sequential length of each module, given in locs, which would yield

$$O-O \ \ programming \ \ effort = Constant \times \sum_{objects} (locs \ per \ object)^{1.5}$$

where the summation is over all objects of the program, although possibly having a slightly larger total number of locs, produces a sum of fractional powers of those numbers smaller than *(total-locs)*$^{1.5}$. Although this must have been certainly anticipated by proponents of O-O programming, it has not been argued in this simple form before to our knowledge. An extensive treatment of complexity in O-O systems from a theoretical and practical point of view may be found in (18).

Along the same lines, Boehm (19) proposed a "constructive cost model" or CO-COMO which estimates the effort PM in person-months as

$$PM = aS^bF$$

where $S$ is measured in thousands of delivered source instruction lines (not comments), $F$ is an adjustment factor depending on process, product, and development attributes, but equal to 1 in the basic initial model, and $a$ and $b$ are dependent on the type of software being developed per the following table – although the values are different from those referenced by Brooks, the functional form is the same.

| Complexity of software project | Value of parameter a | Value of parameter b |
|---|---|---|
| Simple | 2.4 | 1.05 |
| Moderate | 3.0 | 1.12 |
| Embedded | 3.6 | 1.20 |

A related observation in the management of complex software projects is that the amount of effort required exacerbates the management process difficulties as to make the process qualitatively different from managing simpler projects. The process scales up in a different manner and with different requirements. One such requirement is increased testing and debugging of the product, both at the component and at the system level, which typically accounts for more than half of the total cost. Other requirements are the harder-to-satisfy needs for preserving the integrity of the architecture, of separating the architecture from its implementation, and of evolving the projects starting with a pilot.

It has been found that between 50% and 90% of all lines of code in industrial and military software are related to human-computer interfaces (20). With the popularization of the World Wide Web and the personal computer, their software must necessarily be designed for interaction with a large and varied population of users and platforms in an international marketplace whose characteristics must be learned, guessed, or estimated and taken into account in the design. One software solution has been the use of a standard interfaces across applications, driven by the appeal of the desktop metaphor, the WIMP (windows, icons, menus, pointing) interface, graphical user interfaces (GUIs) and other interfaces (HTML) for text. The use of multi-modal interfaces, including speech with software actions, is anticipated. The amount of interactivity required makes the design definitely user centered.

Hierarchical and incremental growth approaches in the design of software are combined with its reuse in "families" of applications, as suggested by Parnas (21). Perhaps the most significant trend in modern software development is that of information encapsulation in units called objects, communicating via messages, which results in object-oriented programming (OOP). This approach, which uses classes and subclasses with instantiations and inheritances, is really a more formalized outgrowth of early artificial intelligence knowledge representations using frames with procedures. Designing software in this manner puts the functional dependencies of the objects explicitly at the message-passing interfaces. The OOP approaches and the new related methodologies of OOP system analysis and design of software systems are the modern way of handling complexity in software engineering.

## 3. COMPLEXITY FROM AN AGENT-CENTERED, MULTI-MODAL VIEWPOINT

The concept of system is a powerful one. By referring to a "system" we can determine those elements that belong or are part of it and those that are not: It is as if we could define a mental envelope across which the system interfaces with its surroundings and with the other systems in that habitat. It is a simplifying assumption that often forgets, unfortunately, to reference the *agent's* observations that provide the basis for the system's model. The use of *agents* is a new conceptual tool in artificial intelligence and distributed systems (14) that shows promise and is being increasingly accepted (15). The use of *agents* in immersive virtual reality that includes human interaction is a more realistic approach to experimentation with complex simulated environments. Some of the opportunities and limitations of this model are current topics of promising research (16). Indeed, the synthetic environments in which humans and their avatars interact would seem a fertile ground for research in the complexity of these artificial worlds, and, potentially, a frame of reference for the more constrained real world.

Just as much of the research in complexity is done via computer-based simulation, much of the human interaction with complex systems is done by interfacing with computers and networks. Again, just as complexity manifests itself at the phase transitions in physical and biological systems, it is at the interface between humans, agents, and computers and networks that most of the layered interfaces will need to be "intelligent" to facilitate interaction. Those interfaces are becoming accordingly sophisticated, involving modalities that go beyond the keyboard and point-and-click and drag-and-drop widgets into multiple modalities adapting themselves to the audio, speech, video, haptic, and user-and-gesture-recognition required by the users of the newer digital libraries and repositories. The fusion of these modalities will present new challenges (31). New metaphors are needed that include user interactions in the considerations of the system complexity with full recognition of human capabilities and weakness and the corresponding alignment of the complementary technology. The complexity of the user-system entity is now increased by the variability of capabilities (or degree of physical impairments) and variances (speech, response, time, etc.) among users. Reinforcement in the fusion of intelligent interfaces in multiple modalities is one positive way of complementing the robustness of the system via redundant inputs.

Learning capabilities of adaptable intelligent agents, including humans, can be approximated by their ability to build models of their experiences based on observations. In a sense, we humans are learning machines and that is part of adaptability. How viable and efficient is this model when put to use, is one of the important aspects in selecting the complexity appropriate to the use. We can abstract a model as a *hypothesis* that is supported by observations. The centroid of a set of observations is a simple such hypothesis. Given a set of possible models we will consider how to choose an optimal one for the amount of data or observations available. We will also consider aspects of how to build the model set.

### 3.1. CONTEXT AND FUSION OF SENSED MODALITIES IN COMPLEX SYSTEMS

With the background presented, we are ready to incorporate the user-interface-system interaction in multiple input/output modalities by approaching complexity from the different *perspectives*, as suggested by the representation of knowledge in KRL-like systems of frames. However, a serious research effort is necessary to optimize the method of fusion of the various modal features for best recognition and communication between the human and the machine. In addition, consideration of the context of the communication can be represented by additional *perspectives*. A block diagram for such an interactive system is given in Figure 2. Each of the components—modal and context—of the perspective is represented in a KRL-line frame as suggested above. A control mechanism is necessary to both select and correlate the optimal modal perspectives that are appropriate to the context of the task undertaken by the system and its interface. These types of modal perspectives are shown and briefly described in the alternative fusion methodologies considered next. For example, one approach would provide the representation of *perspective* (say, speech) with the *perspective* of lip-reading (also called speech-reading). State-of-the-art speech recognizers exist with error rates below 10% in conditioned environments. Their robustness in noisy environments could be enhanced with sensing the synchronized lip movements which occur in the articulation of the utterances, as has been described (29) for alternative fusion models.

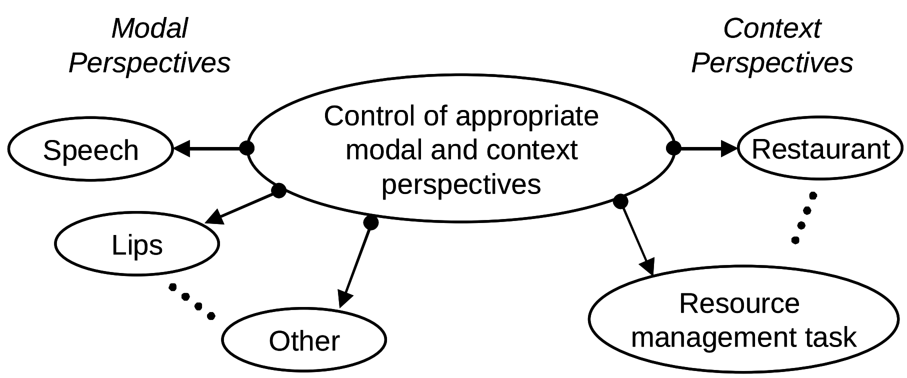



Figure 2. Modal and context components of perspectives and selective control of their frame structure

In deciding at what level(s) fusion should take place in a hierarchy of decisions, one should certainly consider the functional relations between the observations being fused. It seems that the more loosely related they are, the later they can be fused in the hierarchy to make a final determination. There are examples from the fusion of lip-reading and speech that contribute of robust speech recognition. Figure 3a illustrates the case where the decisions with regard to the recognized utterance are made with probabilistic measures before the two are combined. In Figure 3b, a higher-dimensional vector is formed and the decision is made based on the composite data with regard to the recognized utterance. There is yet an intermediate model in which several decisions are made sequentially at different levels of an abstraction hierarchy.

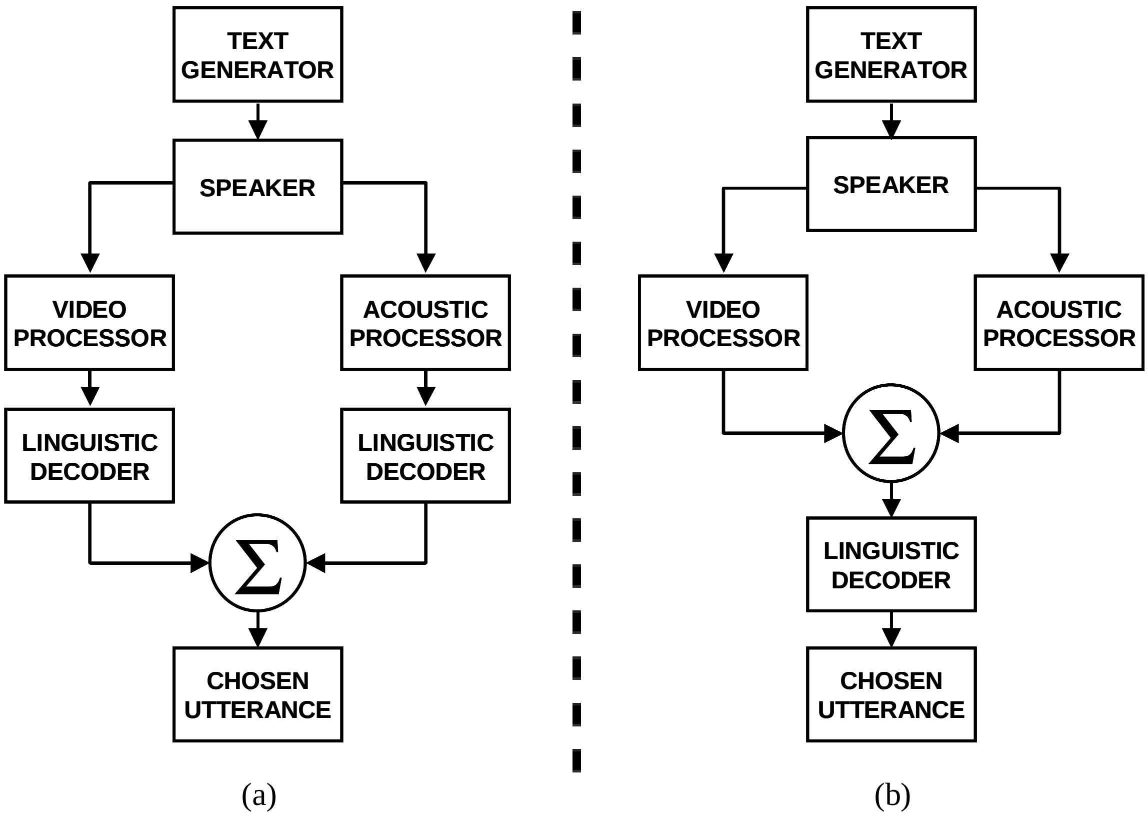



Figure 3. (a) "Separate identification" or "late integration" method,
(b) "Direct identification" or "early integration" model

## 4. COMPLEXITY AND ABSTRACTION

The use of recognizers or classifiers offers a lesson about complexity. We will use the example of speech recognition to illustrate the point, but it holds for many other recognizers and classifiers systems. When a Hidden Markov Model is trained to recognize sentences out of a bigram grammar, for example, we have a case of analysis in which the parameters of the system and its probabilities are averaged over a large number of different speakers to make the operation as independent of the speaker as possible.

It is interesting to note that if one were to use this recognizer in its inverse function, that is, a generator of speech, the results would be terrible even if functioned rather well as a recognizer. Why? The answer is rather obvious. When training as a recognizer, we average the parameter values over as wide a range of samples from different speakers as possible. But those speakers possess correlated parameters that are independently averaged as if they functional dependencies did not exist. Male and female speakers are averaged even though their pitch is quite different. When the recognizer is used as a generator of speech, the outcome is that of neither a male nor a female, but the unnatural average between the two. To have a better recognizer that could be used as a generator, we would need to isolate the functionally related parameters and train with samples that included specifically those cases that can be identified rapidly in the case of real-time speaker-group classification at a low level of granularity. Then, we could do a better job of using the recognizer as a generator for those types for which enough samples existed for previous training. This would require a much larger training population than that usually available.

A plausible argument along the same lines comes from a modern result due to Valiant (32) that has significantly advanced learning theory. It is called the theory of the "probably approximately correct" or PAC learning. A basic formula in that theory relates the number of hypotheses [cardinality of set H or (card H)] to be learned to the probability of making a correct learned decision (independently of the method used to learn!) and the number of samples needed as a minimum for learning within that probability of error $\delta$ not exceeding the value $\varepsilon$. This lower bound on *m* is given by

$$m > \frac{1}{\varepsilon} \ln\left(\frac{card(H)}{\delta}\right)$$

where *"ln"* stands for the natural logarithms and *"card"* stands for the cardinality of the set *H*. For example, if we want to distinguished between some 50 different hypotheses for phonemes and we want to learn under PAC assumptions with probability $\delta$=0.9 of error $\varepsilon$<0.1, then we need some 40 samples (phones) per phoneme from which to learn. As we increase the number of hypotheses (male, female, any one of the dialects for the language, grouping of speakers according to age, etc.), the lower bound on the number of samples needed per hypothesis increases additively by the logarithm of the cardinality of the new groups.

Clearly, there is some compromise where we must use a certain granularity of detail in the number of features (which determine the number of hypotheses) for each level of learning and generation. If the same recognizer is to be used as a generator, it will be necessary to know the functional relations among the parameters involved in the particular generation of a given voice, as they cannot all be independently chosen by arbitrary means. Matters are further complicated when different modalities are considered, because their parameters are also functionally related and the observations must be synchronized for recognition, as in the case of lip-reading.

Something similar happens when we try to represent a probabilistic phenomenon with a Markov transition network. The *order* of the Markov network is the number of previous states on which the next state depends. In general, we use first-order Markov networks for simplicity, but there are two ways in which we can solve actual previous state dependencies beyond the most recent state. One is to use compression by which we code the previous subsequences of states into a new theoretical state (as we do when we use vector quantization) or we go to a lower level of granularity and transform the higher-order Markov network into a more complex first-order network. In the case of vector quantization, we decrease complexity with a loss of information (we generalize to higher levels of abstraction). By reducing the order of the Markov net with a new larger alphabet for labeling the transitions, we increase the complexity of the Markov network (number of states) without loss of information, which is equivalent to a lossless transformation. Most frequently, we assume that only the most recent phenomena affect the next state as a simplifying approximating assumption to manage the increase in complexity. This is exactly what we do when we select the frame duration in the analysis of speech and assume that the duration is long enough to keep the state steady for the time of the frame. This is however usually insufficient to model coarticulatory effects in speech production and multiples of phones (triphones) are used to characterize coarticulation.

These examples illustrate the constraints that we deal with when we try to represent complex systems. The questions are related to the kinds of losses that we must endure when we go from one level of granularity or coarseness in the representation of our model to another, and which are the levels of detail that might invalidate the results for a given agent's goals. In the case of speech recognition, the transformations go from phones to context-dependent phones (triphones), to words, and to sentences, and at each level, there is a transition where losses occur. In a complex system in which it is possible to distinguish hierarchical levels to achieve the user's goal, we can conceive lesser and lesser levels of detail as we reach an apex as we generalize. At each level, we must consider the agent perspective so that we include only features relevant to its needs. Figure 4 attempts to represent this idea. Clearly, within this "cone" of hierarchies, there are other more specialized sub-cones with their own sublevels in topical areas. Something similar has occurred in the professions and in the publishing world with greater specialization to cope with the growth in knowledge and with more focused reader interest.

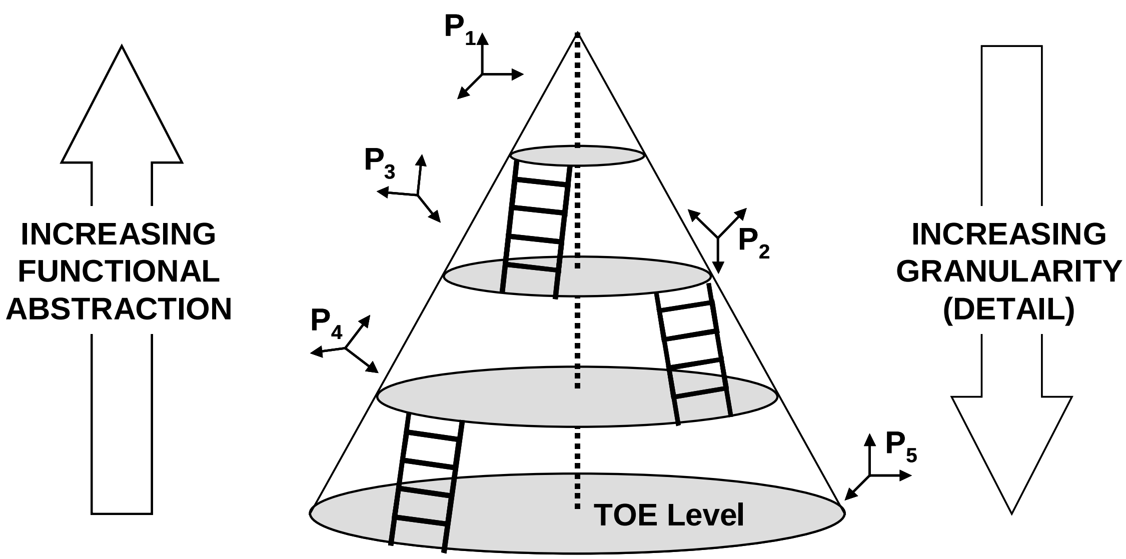



Figure 4. $P_i$ denotes different user perspectives. (TOE is the "Theory of Everything.")

In Figure 4, we consider multiple levels of representation (given that a representation or model is all we can talk about) and the interconnections between the levels which we depict as the "staircases" that show a reductionist explanation when we move from a "theory of everything" or TOE to the upper levels. The point is that we do not require —or want—a complete knowledge from the bottom up —even if we could have it— to operate at a given level. Rather, we need to know explicitly what perspective we take when operating at a particular level. There are enough uncertainties and margins of error at any level that we cannot afford to work satisfactorily if we carry the baggage of all the uncertainties or complexities that come in from the other levels. Once we understand how to move around a given level, we can afford to move up, by generalizing, or down, by specializing and explaining on the basis of more fundamental principles, and advance the scientific state of the art. This relative independence of the behavior of objects at a coarser level of granularity from the influence of less significant factors has been called "decoherence" by complexologists such as in Chap. 11 of (12). Empiricists construct models which have these decoherencies built in.

The point is that we need not have all the facts at a given level to be able to have useful theories. One could conceive at a very high level that in Figure 4 we had the phenomena associated with particle physics (quantum electrodynamics and chromodynamics, superstring theory, relativity effects, etc.) at the lowest level, physics and chemistry in the classical Newtonian sense at the following level, and biology, genetics, anthropology, computer and cognitive sciences, and so forth at the next level, and then social sciences and humanities, and so forth at the uppermost level, possibly a habitat for the culturally evolving “memes” of Dawkins (cultural concepts equivalent to the genes of biology). It should be clear that we use our cognitive primitives at all levels to explain to each other all the theories and hypotheses in terms of basic knowledge originating from our shared experiences and our schooling (witness the technical explanations using common words like flavor and color in talking about quarks to illustrate differences). It is in the use of those basic cognates that our modalities and contextual settings play roles, not necessarily equal at all levels. Within a given level, modality, and context, we also have analogous sublevels of abstraction.

Reducing our view to a single topic, as in Figure 5, and looking at a given perspective under a given modality or modalities, we can observe different levels of difficulty in manipulating or using our representation. There should be a certain level of abstraction at which operating with that representation (say, making inferences) is easiest. This is an indication that we have different levels of complexity for different levels of abstraction if we correlate the difficulty of manipulation with complexity.

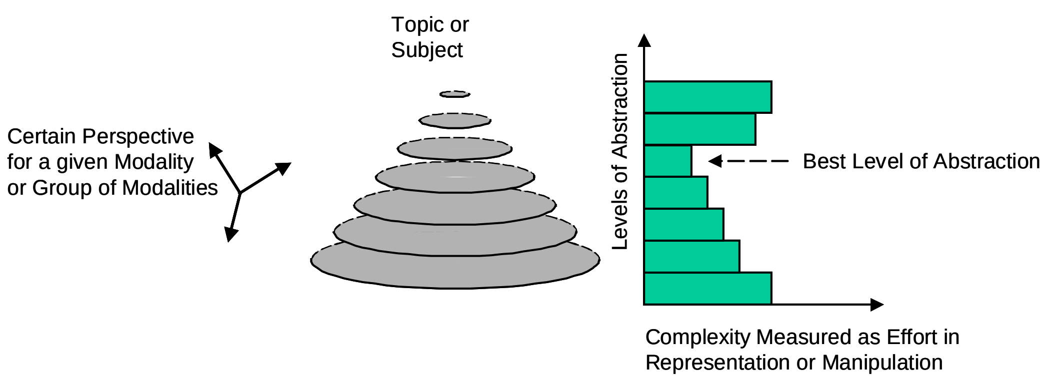



Figure 5. Different levels of difficulty on levels of abstraction in a given topic

One can think of the example of the “modified checkerboard problem in which the perspective and level of representation facilitate reaching certain conclusion. Consider a checkerboard with 64 squares of size $1 \times 1$ colored in the usual fashion. It is clear that this checkerboard could be covered totally with 32 tiles of dimension $1 \times 2$. Then let us remove the two end tiles of a diagonal of the checkerboard as shown in Figure 6. The question asked is: Can we now cover the checkerboard with 31 tiles of size $1 \times 2$? We can easily see that it is impossible if we consider that each of the 31 tiles of size $1 \times 2$ has to cover, of necessity, one black and one white square of checkerboard. Our representation made an otherwise more difficult inference relatively simple.

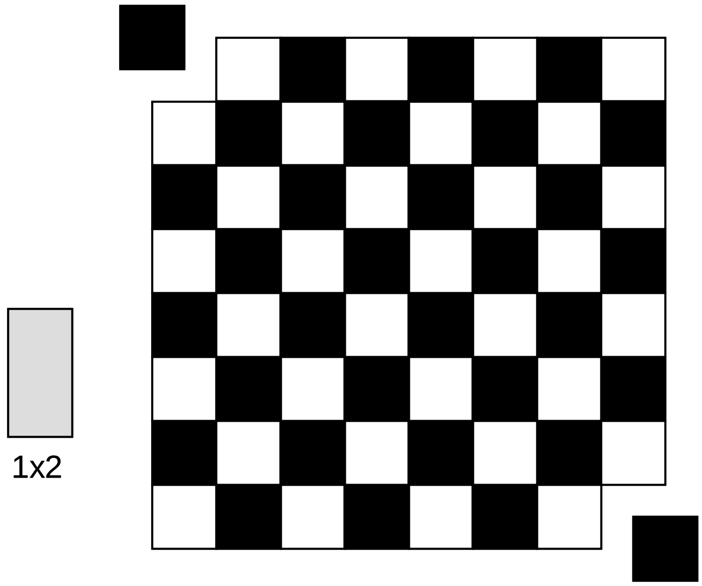



Figure 6. The modified checkerboard problem

### 4.1. AN APPROACH TO SELECTING AN ABSTRACTION LEVEL

Given a user's perspective we have hypothesized here that, there is a level of abstraction of the object's description that best facilitates the agent's interaction with the complex object or system in the sense of that perspective. Here we will advance this hypothesis by utilizing a measure of complexity and its application to an illustration of the problem. A user or intelligent agent interacts with a given world through observations and forms "learned" models or hypotheses at various levels of detail. Those models are organized and may be interfaced at different levels of abstraction (see cone on the right of Figure 7 below, analogous to the one on the left of Figure 5) one of which is chosen by the user or agent. We show a measure that yields an optimal choice for a set of such models.

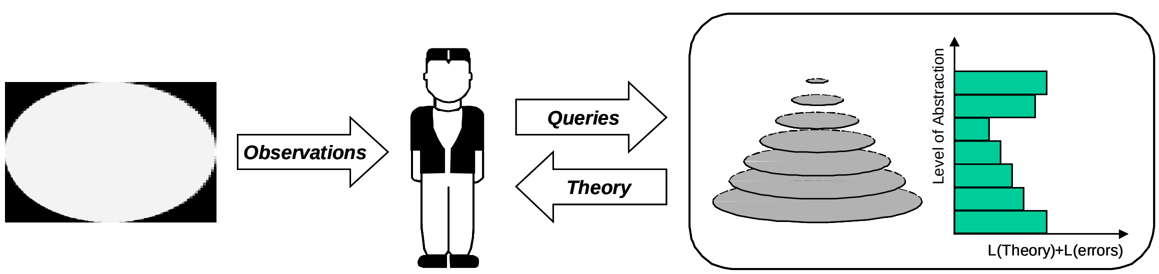



Figure 7. A user-centered perspective of the hierarchical abstract model with different MDL complexities.

#### 4.1.1. MINIMUM DESCRIPTION LENGTH (MDL)

Corresponding to each of the hierarchical levels of abstraction in Figure 7 (as in Figure 5) there is a corresponding (bar) level of complexity for instantiations of the model at different granularities. We expect an instantiation where interactions with the model by the user or agent are easier. An intuitive approach to measuring complexity is the Minimum Description Length (MDL) principle (33). Imagine a communications game where a sender is to transmit a sequence of observations to a receiver. MDL states that the (stochastic) complexity of the problem can be measured by the number of bits used to encode a theory, plus the number of bits used to encode the observations with the aid of the theory. Thus, the optimal hypothesis *h* (out of a set H) for a given observation sequence D is:

$$h_{MDL} = \arg\min_{h \in H} \left\{ \underbrace{L(h)}_{regularity} + \underbrace{L(D \mid h)}_{error} \right\}$$

where *L(h)* is the length of the hypothesis and *L(D|h)* is the length of the data given the theory. *L(h)* captures regularity (the inverse of complexity) in the observations, whereas *L(D|h)* measures those aspects of the observations not predicted by the hypothesis. When the number of observations is small, the term *L(h)* dominates, biasing the selection criterion toward more regular, less complex hypotheses. As the number of observations increases, the term *L(D|h)* becomes increasingly important, allowing more complex theories to be selected. The MDL principle is, thus, a data-driven instantiation of Occam's Razor: choose the simplest theory that explains a phenomenon for a given amount of data. MDL controls over-fitting by limiting the size (number of parameters) of the acceptable hypotheses based on the amount of data available for the reliable estimation of their parameters. In this regard, MDL is related to PAC learning, which provides lower bounds for the minimum amount of data needed for a given learning problem.

To illustrate the MDL principle let us assume the problem shown in Figure 8, where the goal is to transmit the observation sequence $\{y_1, y_2, \ldots, y_N\}$, shown as raw data, across a channel with a fixed quantization error. The naïve alternative would be to transmit the binary codes of the raw data after it has been digitized with the required quantization level. Three other alternatives are shown in this figure. Theory A (least complex theory) encodes the series with a linear model $y = a_1 t + a_0$, Theory B performs a cubic fit $y = a_3t^3 + a_2t^2 + a_1t + a_0$, and Theory C (most complex theory) uses an N-th order polynomial $y = \sum_{k=0}^{N} a_k t^k$ and, therefore, fits the data without error. The relative cost of transmitting each theory is, in a first-order approximation, given by the number of parameters of the model (the order of the polynomials) times the number of bits used to encode each coefficient. The cost of transmitting the time series is proportional to the error bars (difference between the data and the model) divided by the required level of quantization, which remains fixed across models. There obviously exists a trade-off between the naïve approach of transmitting N raw data points and the overkill of transmitting N coefficients (Theory C). Depending on the number of data points, Theory A or Theory B will become the optimal method for transmitting the time series.

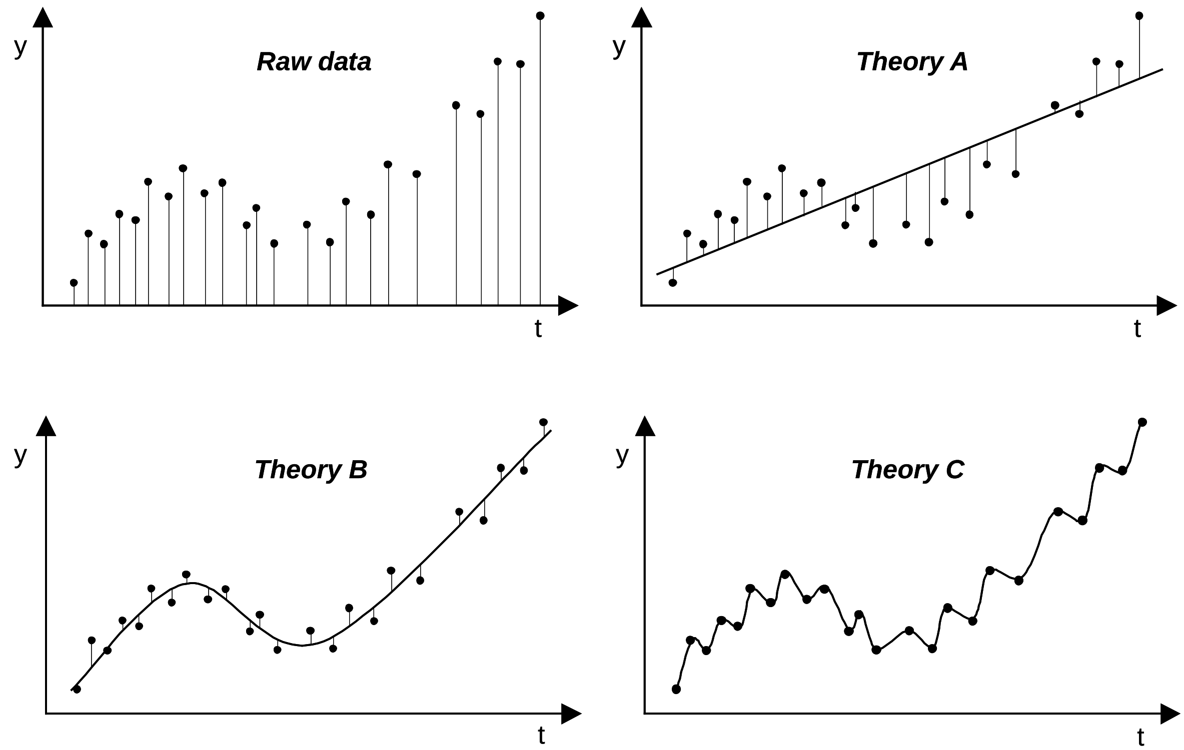



Figure 8. Different theories for transmitting a set of data across a fixed quantization error channel

Interestingly, the MDL principle has strong connections with the Bayesian formalism for model selection (34). The Bayes-optimal hypothesis $h_{MAP}$ is the one that maximizes the posterior $P(h|D)$: the probability that a given hypothesis $h$ is true after $D$ has been observed. This Maximum A Posteriori (MAP) principle can be stated as:

$$h_{MAP} = \arg\max_{h \in H} \{P(h \mid D)\}$$

Applying Bayes's theorem and then taking logarithms:

$$h_{MAP} = \arg\max_{h \in H} \left\{ \frac{P(D \mid h) P(h)}{P(D)} \right\} = \arg\max_{h \in H} \{P(D \mid h) P(h)\} = \\ = \arg\max_{h \in H} \{\log_2 P(D \mid h) + \log_2 P(h)\}$$

From Information Theory we know that the optimal coding of a grammarless string of messages $\{m_1, m_2, m_3, ...\}$ is achieved by assigning codes length $L(m_i) = -\log_2 P(m_i)$, where $P(m)$ is the probability mass function governing the generation of messages. In other words, events that are more likely to occur are encoded with fewer bits so that the overall message string length is minimized as entropy is maximized. If such an encoding is used to transmit hypothesis and data, MAP and MDL are equivalent since maximizing the log posterior equates to minimizing description length.

### 4.1.2. MDL AND THE SELECTION OF AN ABSTRACTION LEVEL

The same MDL model selection principle can be adopted to determine an appropriate level of abstraction to explain a given phenomenon, relative to the perspective of a user or intelligent agent. In this case, hypotheses or theories "explaining" the model become different levels of abstractions. The user/agent makes observations and tries to interpret/understand them by asking questions to an oracle. The oracle's task is then to find an appropriate level of abstraction that yields a simple mental model for the agent, yet accurate enough to explain the phenomena. A very abstract model will not be able to accurately explain the observations, resulting in a large number of special cases to handle the intricacies of the observations. On the other hand, a low level (of abstraction) model will accurately explain the observations, but will be too complex to be of any use to the agent.

## 5. CONCLUSIONS

The proposed representation scheme, coupled with its context for the agent or agents (whether dependent on its use or not), which includes aspects of the training of the system if it is a learning system, must result in a more pragmatic (albeit circumstantial) complexity-evaluation process. This is because a certain degree of specialization is obtained by including the control structure selecting the favored modalities and perspectives for a given user and context. This approach does not consider the complexity of an object, action, or event in its own right, and this may be criticized because it is not a measure of complexity specific to the object. Given the lack of a consensus with regard to what represents inherent complexity, one might argue that such a measure is an elusive as that of absolute beauty or similar subjective concepts. However, the proposed approach is more pragmatic because it generalizes the determination of complexity via selective specialization, as required by its modal and contextual use. The *perspective*, in a sense, determines both (1) and the nature of the question (the sensing modality or modalities of the perspective chosen by the control in Figure 2 to the left) and (2) the context (to the right below the control in Figure 2) to be used in obtaining the answer, whereas the knowledge at each section or level of the cone of abstractions in Figure 4 may provide an answer at a given level of detail.

We have suggested an approach based on *perspectives* in the representation of the system interfaces that involve not only the modalities that optimize the interactions for a specific task but also the fusion of such interactions in different ways, or even possibly at different sequential layers of an abstraction hierarchy. How multiple modalities are best fused is an open question that must be addressed. Also, needing attention are the context and task integration within the interface. This would make the approach to the complexity of a system integrate its user and the context selection as well as its automatic choice and fusion of the modalities of the interface.

We have also shown an approach that allows an optimized choice of abstraction level given a set of model hierarchies and a perspective for using a complex system. A central point is that *the modalities employed by the agent using the complex system being evaluated are, of necessity, part of the system's evaluation in a specified context, and that both modal and contextual aspects (perspectives), are factors in facilitating the use at a given level of detail in the hierarchy of complexities.*

One additional aspect is worth mentioning in dealing with complexity at a specific level from the human-centered perspective. Motivation and interest are undeniable drivers of a successful investigation. Investigating an abstract phenomenon (across levels) may be of interest to a seasoned scholar, but for a new researcher, it is more interesting to deal with specific instances and later generalize, or, at best, generalize along the way of investigation. Human dedication is observably sharpened with concentrated by a focusing perspective.

## 6. ACKNOWLEDGEMENTS

Some of the equipment used to investigate fusion of visual and auditory speech was acquired with the grant NSF-CDA 9601670, “ARI: Development of Specialized Communications and Terminal Equipment for Research in Information Technology and Education Technology,” which is gratefully acknowledged. More recently work on NSF-EIA 9906340, "Speech Driven Facial Animation," has also resulted in contributions to this article. Also, comments on database “views” and other topics from Jeff Ellis, a WSU graduate student who read a draft of this article, were incorporated above and are acknowledged. Flora J. Garcia, of Time, Inc., edited an early version of the article, which resulted in significant improvements.

## REFERENCES

1. “An Approach to Complexity from a Human-Centered Artificial Intelligence Perspective” in *Encyclopedia of Computer Science and Technology*, Vol. 40 (A. Kent and J. G. Williams, eds.), Marcel Dekker, New York, pp.1-16, 1999.
2. L. Peliti, A. Vulpiani (Editors), *Measures of Complexity : Proceedings of the Conference Held in Rome (Lecture Notes in Physics, 314)*, Springer-Verlag, Berlin, 1988.
3. Rissanen, Jorma, *Stochastic Complexity in Statistical Inquiry* (Series in Computer Science, Vol. 15), World Scientific, 1989.
4. R.J. Brachman and H.J. Leveque (eds.), *Readings in Knowledge Representation,* Morgan Kaufmann, Los Altos, CA, 1985.
5. M. Minsky, “A Framework for the Representation of Knowledge,” One of the most recent versions is found in *Mind Design*, MIT Press, Cambridge, MA, pp. 95-128; and in Ref 1, pp. 246-262.
6. D.N. Bobrow and T. Winograd, “An Overview of KRL, a Knowledge Representation Language,” *Cogn. Sci.*, *1* (1), 3-46, (1977).
7. J. Rasmussen, A.M. Pejtersen, and L.P. Goodstein, *Cognitive Systems Engineering,* Wiley-Interscience, New York, 1994.
8. J. Ullman and J. Widom, *A First Course in Database Systems,* Prentice-Hall, Englewood Cliffs, NJ, 1997.
9. P.M. Jones and C.M. Mitchell, “Operator Models for Supervisory Control Systems,” in *Perspectives: Proceedings of the 33rd Annual Meeting of the Human Factors Society,* October 16-20, 1989, Vol 1, pp. 291-295.
10. J.E. Laird, A. Newell, and P.S. Rosenbloom, “SOAR: An Architecture for General Intelligence,” *Artificial Intell., 33*(3) (1987).
11. F. Brooks, Jr., “No Silver Bullet: Essence and Accidents of Software Engineering,” *Information Processing 1985, Proceedings of the IFIP Tenth World Computing Conference,* H.L. Kugler (ed.), Elsevier, Amsterdam, 1986. Reprinted in *IEEE Computer Magazine, 20*(4), 10-19, (1987).

12. M. Gell-Mann, *The Quark and the Jaguar, Adventures in the Simple and the Complex,* W. H. Freeman, New York, 1994.
13. F. Brooks, Jr., *The Mythical Man-Month: Essays on Software Engineering—Anniversary Edition,* Addison-Wesley, Reading, MA, 1995.
14. P. Maes (ed.), *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back,* MIT Press, Cambridge, MA, 1991. Also published as a special issue of *Robotics and Autonomous Systems,* June 1990.
15. S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach,* Prentice-Hall, Englewood Cliffs, NJ, 1995.
16. N.I. Durlach and A.S. Mavor, *Virtual Reality: Scientific and Technology Challenges,* National Academy Press, Washington, DC 1995.
17. S.S. Brilliant, J.C. Knight, and N.G. Leveson, "Analysis of Faults in a *N*-Version Software Experiment," *Trans. Software Eng., 16*(2), 238-247 (1990).
18. B. Henderson-Sellers, *Object-Oriented Metrics – Measures of Complexity*, Prentice-Hall, 1996.
19. B.W. Boehm, *Software Engineering Economics,* Prentice-Hall, Englewood Cliffs, NJ, 1981.
20. B.A. Myers and M.B. Rosson, "Survey on User Interface Programming," in *Proceedings SIGCHI'92: Human Factors in Computing Systems*, May 3-7, 1992, pp. 195-202.
21. D.L. Parnas, "On the Design and Development of Program Families," *IEEE Trans. Software Eng., 2*(1), 1-9 (1976).
22. R. Dawkins, *The Blind Watchmaker,* Norton and Co., New York, 1987; (see pp. 6-14 for complexity views).
23. S. Turkle, *Life on the Screen,* Simon & Schuster, New York, 1995.
24. G.B. Dyson, *Darwin Among the Machines,* Addison Wesley, Reading MA, 1997.
25. S. Lloyd, "Physical Measures of Complexity," in 1989 *Lectures in Complex Systems,* E. Jen (ed.), Addison-Wesley, Redwood City, CA 1990, pp. 67-73.
26. J. Horgan, "From Complexity to Perplexity, *"Scientific American,* 104-109, (June 1995).
27. G.J. Chaitin, *Algorithmic Information Theory,* revised third printing, Cambridge University Press, New York, 1990.
28. P. Grassberger, "Towards a Quantitative Theory of Self-generated Complexity", Int. J. Theor. Physics, vol. 25, no. 9, p. 907-938, 1986.
29. B.D. Yoon and O.N. Garcia, "Cognitive Activities and Support in Debugging," in *1998 Proceedings of Symposium on Human Interaction with Complex Systems*, 1998.
30. J. Maddox, "Complicated Measures of Complexity," in News and Views, *Nature*, vol. 344, April 19, 1990.
31. J.S. Chen and O.N. Garcia, "Challenges in the Fusion of Video and Audio for Robust Speech Recognition," in *1997 AAAI Spring Symposium on Intelligent Integration and Use of Text, Image, Video and Audio Corpora*, March 24-26, 1997.
32. L. Valiant, "A Theory of the Learnable," *Commun. ACM, 27,* 1134-1142.
33. J. Rissanen, *Stochastic Complexity in Statistical Inquiry* (Series in Computer Science, Vol. 15), World Scientific, 1989.
34. T. Mitchell, *Machine Learning*, McGraw-Hill, 1997.