A Native Intelligence Metric for Artificial Systems

John Albert Horst
100 Bureau Drive, Mail Stop 8230
The National Institute of Standards and Technology (NIST)
Gaithersburg, Maryland, USA 20899-8230
john.horst@nist.gov
voice: (301)975-3430

ABSTRACT

We define native intelligence as the specified complexity inherent in the information content of an artificial system. The artificial system is defined as a system that can be encoded in some general purpose language, expressed minimally as some finite length bit string, and decoded by a finite set of rules defined *a priori*. Using this definition of native intelligence, we employ a chance elimination argument in the literature to form a simple, but promising native intelligence metric. Several anticipated objections to this native intelligence metric are discussed.

1 INTRODUCTION

We define two perspectives on artificial system intelligence: (1) native intelligence, expressed in the specified complexity inherent in the information content of the system, and (2) performance intelligence, expressed in the successful (*i.e.*, goal-achieving) performance of the system in a complicated environment. In this context, complexity is simply Shannon information [5]. Specified complexity is Shannon information matched with meaningful patterns (*e.g.*, orthography, syntax, and semantics). Native intelligence aggregates things like potential intelligence, learning ability, system integration, richness and potential effectiveness of individual behaviors, and intellectual reasoning capabilities. These are fundamental, theoretical, and innate aspects of intelligence. The performance perspective on intelligence aggregates things like the efficiency of design and maintenance, real-time performance, and the ability to effect desired physical changes on the environment. These are external behavior characteristics that can be measured without knowing where the intelligence came from, how it is represented, or what algorithms are used internally to process the data and produce effective output.

These two perspectives have a parallel with the distinction between theoretical and experimental sciences. In this regard, the native and performance perspectives are complimentary and not competitive. They also ought to give the same results when applied to the same quantities in identical systems. Each one ought to inform and contribute insight to the other.

The essence of intelligence is completely non-material. This must be true, of course, if a native intelligence perspective is to have any validity. Intelligence can be embodied, but it is completely independent of embodiment. For example, a finite state machine (FSM) for a particular behavior can be stored on a compact disk, located in the mind of the designer, or spoken out loud to an audience. The FSM is completely independent of the medium of storage, type of representation, or mode of transmission. It exists as information. The quantitative measurement of intelligent systems is becoming and will become an increasingly important thing to be able to do. The discovery of DNA in living things as the living blueprint of life forms an existence proof that the essence of living things is non-material information. DNA is merely the medium; the information is the real intelligence[1].

1.1 The value of a native intelligence metric

A metric for quantifying the intelligence of a system *independent of its performance*, *i.e.*, a native intelligence metric (NIM), is needed. A NIM would bring substantial gains beyond that possible from performance intelligence metrics alone. A measurement of native intelligence can be made prior to the simulation or execution of the actual system, since all that is necessary to apply the metric is the representation of the system (as a string of bits), the rules for interpreting the string, and the meaning matching the system (the semantics). To measure success earlier in the

[1] Actually, the concept that all hereditary characteristics come from the DNA and *only* the DNA (called Crick's Central Dogma) has been shown to be false [1].

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
AUG 2002		**00-00-2002 to 00-00-2002**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
A Native Intelligence Metric for Artificial Systems	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **National Institute of Standards & Technology (NIST),Manufacturing Engineering Lab,100 Bureau Dr,Gaithersburg,MD,20899**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
Proceedings of the 2002 Performance Metrics for Intelligent Systems Workshop (PerMIS -02), Gaithersburg, MD on August 13-15, 2002

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**8**	

design phase is known to be critical to successful engineering design, particularly for large scale, complex systems. This is true even if the NIM is suboptimal, namely, even if some intelligence actually in the system is missed by the NIM. The ability to define and measure intelligence merely from its native intelligence offers promise to improve important quantities such as system time-to-market and quality. A NIM will also allow a more straightforward debug of the system, since the problem and solution will be more localized. For example, if we know that a few bits in the string cause the problem, we may be able to easily fix the problem by correcting just those few bits.

1.2 The relationship to living systems

Our definition of native intelligence of artificial systems is substantially unlike what is known as "IQ intelligence." IQ intelligence is a type of performance intelligence (intellectual mostly) in which the result is used to measure the potential of the human subject. Furthermore, a fundamental difference between living organisms and artificial systems is that the latter are completely malleable. This is not just a difference but a distinct advantage. Therefore, IQ intelligence cannot and should not constitute a model for the native intelligence of artificial systems.

In human children we often want to distinguish between the potential of a child to learn and how much they have actually learned. Being unable to do this is a source of incredible frustration. The problem is that we do not yet understand how to measure native intelligence of humans. I suspect that some day, perhaps soon, we will be able to make some level of a determination simply be interpreting the genetic information in the genome of each individual. This subject is the topic of a film entitled, Gattaca [2][2], in which the protagonist overcame his genetic "destiny" in spite of an oppressive determinism in the surrounding culture. With humans this is a loaded topic, but with machines the situation is much more amenable, since we can design the system for analysis! In fact, we want to design the system for analysis. We want the "intelligence meter" to detect the maximum amount of intelligence, and since we have full control on how we encode the description of the design of the system (just like we can always encode a computer program into a finite sequence of bits), we will choose to encode it in a standard format so that the intelligence is measurable.

With the NIM are we claiming that a system's performance is fully constrained by its initial determining information? Isn't this crass determinism? No, it is not. A system's performance is not fully constrained by the disembodied information describing the system, since performance depends on the nature of the environment, which will affect future performance. For example, if an adaptive filter adjusts parameters for a certain environment, subsequent exposure to another environment may cause instabilities in the filter. High native intelligence does not mean that the intelligence is realized. The NIM is not meant to exclude the system's potential response to some future contact with its environment. There is a multitude of ways that the same intelligent system can by realized. Finally, living systems are still a great mystery and are arguably not fully defined by algorithms (in the way that we are defining artificial systems) [3].

1.3 Specified complexity and computational theory

Measuring native intelligence is much like the problem scientists and mathematicians asked around the turn of the century. What are the natural classes of computing machines and can we measure the power of a machine? Are there limits of computation and if so what are they? Similarly our question becomes the following. How can we measure the intelligence of a system without ever seeing it perform or even simulating its operation?

In a manner consistent with definitions of information in standard computational theory texts, we propose that the minimal information describing a system entails its intelligence, *i.e.*, the intelligence is fully contained within the information describing the system [4]. Of course, by information we do not mean Shannon information, namely, that the information is merely complex [5]. Random noise is complex, but is not specified (*i.e.*, it has no "meaning") and therefore conveys no real (useful) information. Rather, what we mean by specified complexity is that the information is both complex and has discernible meaning, which is to say that it has recognizable patterns that indicate the power of the system to perform complex and useful tasks. This specified complexity can be represented and analyzed without actually realizing the system and seeing it perform. We argue that intelligence can be distilled down into pure information, even a finite amount of information. The description of a computational machine can always be reduced to a finite string of 0's and 1's. To represent a truly intelligent system, the system's information must be both complex (in the Shannon sense) and specified. In computation theory, information content of a machine is defined as the minimum representation of that machine. On the other hand, certain machines may look complex, but they may not do anything useful. Therefore, the system's description needs to be specified. It

2 Certain commercial companies and their equipment, instruments, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identification is not intended to imply any judgement by the National Institute of Standards and Technology concerning the companies or their products, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose

needs to have meaning that, in some way, is grounded in reality or truth. And furthermore, we have some way of measuring the amount of this connection, *i.e.*, the greater the system connects with real meaning, the greater the intelligence. Here then is a method for measuring the native intelligence of systems: Measure the information content, using information theory and computation theory and measure the magnitude with which it can be matched to patterns representing truth-grounded semantics (something like a dictionary would be employed). For example, the DNA string is certainly complex, *i.e.*, there is no known simple representation for the entire string (no simple equation generates the string of nucleotides) and DNA is also certainly associated with known truth-grounded semantics, *i.e.*, it can be used to form proteins that are the machinery and computing engines of the most complex factory yet discovered, the living cell. Clearly, detecting and measuring the connection of a system's information with truth-grounded semantics is a substantial challenge.

1.4 The analogy to linear systems and the mixed success of intelligent systems

To generate an effective NIM, some mathematically sound concept of what fundamentally constitutes intelligence is needed. Why not use some of the existing IS models, such as neural nets, evolutionary programming, hierarchical models, behavior-based control, and fuzzy control? Because all these models share in common that they are primarily models for facilitating IS design and execution and do not in themselves constitute formal models of intelligence that will readily yield a NIM.

Why not use linear systems theory as a model for a NIM? The successes of traditional linear system theory can in many ways be considered the target we would aim for in the development of a useful NIM. In traditional linear system theory we have a most agreeable situation. The dynamic behavior of many systems can be described as differential equations of motion and dynamics and, coupled with state space structures and linear control theory, we can determine absolutely whether the system is controllable, stable, or observable without doing any simulation or building and testing of an actual system. However, it is well known that when we add non-linearity and time variance, for example, we can measure the stability of only a small subset of the total space of all non-linear systems. The mathematics also gets substantially more complicated and sophisticated. A large class of real-world systems (including discrete event systems) do not seem to submit easily to concise and simple mathematical descriptions. So this path (employing differential equation models) for determining the intelligence of systems has hit a fundamental barrier, it seems.

What about using formal methods as a basis for a NIM? The NIM offered in this paper may have much in common with formal methods and further study of this connection should be pursued. Formal methods have had a very limited impact, but this may not be due to any limitations in the theory. The limited impact seems to be due to the fact that many systems do not submit easily to the formal syntax of predicate calculus and that the learning curve is too high for most system designers to comprehend formal methods in order to successfully use them. Tools for formal methods also seem to be lacking or need substantial improvements [6].

The field of intelligent systems research has rightly been criticized by the traditional control systems community as being open to the proliferation of *ad hoc* design techniques, such as neural control, fuzzy control, hierarchical control, etc. These techniques are not grounded in physical and mathematical theories in the way that traditional control systems are grounded in the theory of dynamics and differential equations. There is certainly no commonly accepted equivalent to differential (or difference) equations for large-scale discrete-event systems (large-scale discrete-event systems may be considered to be the domain of IS). Traditional control systems can be measured in terms of their stability, controllability, and observability without realizing (or embodying) the system. These are metrics that allow us to measure the native intelligence of traditional control systems. For example, a system that goes unstable in a certain critical region would intuitively be less intelligent than one that has no such instability but in every other way performs equivalently. We have no such general metrics for measuring the native intelligence of large-scale discrete-event systems nor do we have the capability to compare the native intelligence of systems designed according to different methods. It is important to have quantitative measures for the relative performance of systems designed using neural control versus those designed by fuzzy control, etc.

Certainly *ad hoc* systems are helpful and even necessary at a time when there is no accepted theory with which to build non-*ad hoc* systems. However, without a theory, we often witness a proliferation of inflated claims of system performance that can not be met. This has several time produced a backlash from funding organizations who made financial commitments based on these groundless claims. This is all the more reason why a well grounded NIM would be helpful.

Furthermore, intelligent systems will have to be designed for analysis. But this is exactly what we do when we design linear, time-invariant control systems in a state space form. It's easier to analyze that way. Furthermore, vendors of systems will not want to hide the intelligence of their systems, but will want others to know for sure that their system is truly intelligent. The advantage of having a

NIM should be obvious. That way we can measure intelligence prior to any commitment to simulation in software or testing in hardware. Computation theory can also help with measuring the degree to which the information (describing the system) matches with truth-grounded semantics. For example, a system that can recognize or distinguish between a broader set of input "languages" is more powerful than one that cannot. For example, a finite state automata (FSA) can be designed to recognize strings of the form 0*1*, but no FSA can be built that recognizes strings of the form 0^n1^n, whereas a push-down automata (PDA) can be built to recognize strings of either type. Therefore, the PDA is more powerful (intelligent).

Measuring intelligence is similar to what a cryptanalyst does normally except that when measuring intelligence we don't anticipate the element of planned deceit. The cryptanalyst is looking for signs that the bits have identifiable patterns underlying the randomness. If there is no randomness, there are no patterns, and no intelligent message underneath. With randomness, the bits could actually be truly random and therefore be nonsense. However, it may just appear to be random and the patterns are not obvious to the cryptanalyst. Ostensibly, an IS would be designed not to fool the IS metrics analyst to think the system is nonsense, but rather the IS would be designed so that its intelligence would be easily perceived by all.

1.5 The nature of a NIM

Perhaps the solution is not in an analogy to linear systems theory, as has been the hope, but rather in information theory, probability, and complexity theory. This is the research question we have asked ourselves and hope to provide some direction towards an answer. We argue that the essence of intelligence in living things is information (versus physics or chemistry). If this is so, we need to discover appropriate tools to correctly analyze that information to glean from it the level of intelligence in the system.

We claim that chance, regularity, and intelligence are mutually exclusive. Regularity is indicated by signals of low complexity. Therefore simple Shannon information filters can eliminate regular systems from consideration as intelligent systems (recall our definition of native intelligence as complex and specified information, in which the level of complexity and specification is proportional to the level of intelligence). After first eliminating regularity, if we can measure the probability that a system arose from chance processes alone, then the intelligence in that system must in some simple way be related to that probability. In a manner similar to Shannon information theory, we propose that, if there is a measure of the probability that doesn't just measure the mere complexity of the information (Shannon), but measures the *specified* complexity of the information, then the expression $-\log_2 p$, where p is the probability that the supposed specified complexity (of the system) arose purely through a logical chance hypothesis, is a theoretical measure of the system's native intelligence. Such a measure exists [7]. We will describe and examine this measure and suggest how it applies as a NIM.

2 THE CHANCE ELIMINATION ARGUMENT

Our goal is to find a quantitative intelligence metric for an artificial system that can be applied to the minimal informational description of the system. Another way of looking at this problem is to ask ourselves, what is the probability that this system either arose merely by chance (or merely by regular processes)? Regularity is relatively easy to eliminate from consideration since the information can in this case be generated from a relatively small expression (or a small number of "lines of code"). It may contain many bits, but not have much information. An example of regularity is an infinitely alternating string of four 0's and four 1's. The lines of code are simple: write down four 0's, write down four 1's, repeat steps one and two. Clearly there is little intelligence in this system. So, having eliminated regularity, if the probability that chance did the work is high, the intelligence in the system will be low; if the chance probability is low, the intelligence will be high. Why is this so? There are only three options for the source of artificial system information: chance, regularity, or intelligence. It has to be one of the three. So if chance and regularity have sufficiently low probability, then the source must be intelligence. Say that the system needing analysis is encoded in what appears to be a random sequence. If we happen to know that it is the description of a complex space shuttle control system, we can readily eliminate chance and declare it to be originating from intelligence. Note that false negatives are possible (thinking it is a chance process when it is an intelligent process), but false positives are avoidable.

A theory for chance elimination has been developed by Dembski in [7] and is called the Generic Chance Elimination Argument (GCEA). This theory forms the basis for the NIM development in this paper. Therefore, we will begin with a summary of the aspects of GCEA relevant to IS metrics. In the next section, we will investigate how the theory might be (simply) applied to IS metrics.

We start with a subject, S, that observes an event, E. By analyzing the circumstances surrounding E, S defines the chance hypothesis, H, gotten from a reasonable chance process that might have been responsible for E. S discovers (doesn't matter how) a pattern, D, that matches the event,

E^3. With D* as the event associated with the pattern, D, S calculates the conditional probability, P(D*|H), of the event, D*, assuming the chance hypothesis, H, is true. S tests whether P(D*|H) is less than a universal probability bound, δ. Dembski determined that $\delta = 10^{-150}$ from fundamental physical limits (space, particles, and time) within the known universe[4]. Through knowledge of some side information, I, S computes the conditioned complexity $\varphi(\mathrm{D} \mid \mathrm{I})$ of formulating the pattern, D, given the side information, I. "Complexity" in this context is defined within the domain of "complexity theory," the most common example of which is computational complexity. Complexity theory is a dual with probability theory in the sense that with probabilities we are dealing with "events" conditioned by "background information" and with complexity theory we are dealing with "problems" conditioned by "resources." So formulating D is the "problem" and I are the "resources" for solving the problem. A problem conditioned by resources places $\varphi(\mathrm{D} \mid \mathrm{I})$ in the domain of complexity theory. S computes a tractability bound, λ, which is used to calibrate the complexity values, since complexity, unlike probability, is essentially uncalibrated. λ is the upper bound of complexity, below which the side information, I, is sufficient to form the pattern, D. Finally, if $\mathrm{P(D^*|H)} < \delta$ and $\varphi(\mathrm{D} \mid \mathrm{I}) < \lambda$, S can infer that E did not occur according to H. This is a substantially abbreviated form of Dembski's argument, but is sufficient for our needs. We will now give an example to help clarify the GCEA.

Say we are S and we stumble upon Stonehenge. We don't wonder whether humans carried the stones (some weighing over 5×10^4 kg) to the site. However, we do wonder whether the particular arrangement of the stones is meant to align with certain seasonal, celestial events such as the positions of sun and moon at summer solstice and during eclipses. S declares that E is the particular arrangement of stones S encounters that may be exhibiting alignment with celestial bodies at certain seasonal times. S determines that the designer of Stonehenge had full view of the sky enough days in the year to be able to note seasonal occurrences. S further determines that at summer solstice, for example, the rising sun precisely aligns over the Heel Stone shining rays directly into the center of the monument in the inner horseshoe arrangement of stones. S notes that there are several of these curious alignments of lunar and solar events (including eclipses) with various stones. This alignment of events with stones constitutes D, our pattern. D* then constitutes the particular pattern of stones consistent with the alignments. E would then, at the least, be delimited by D, but for simplicity's sake, say D* = E. Now we calculate the probability, P(D*|H), that D* could happen by chance alone, even though D* was gotten by assuming the matching of the stones with particular celestial events. The various celestial events and our prehistoric Stonehenge designer's awareness of these events constitute the side information, I. We calculate the tractability bound, λ, below which the particularly alignment of stones is possible given I. We then calculate $\varphi(\mathrm{D} \mid \mathrm{I})$, the actual complexity of forming the pattern given I. If $\mathrm{P(D^*|H)} < \delta$ and $\varphi(\mathrm{D} \mid \mathrm{I}) < \lambda$, S can infer that the particular arrangement of aligned stones, E, did not occur according to the chance hypothesis, H.

All these definitions and formulations are completely consistent with probability theory, complexity theory, and statistics. Note that the measurement is heavily dependent on the correctness of I (*i.e.*, that I is statistically "sufficient"). For example, we must have sufficient knowledge whether the prehistoric Englishman could perceive those celestial objects and that the alignments have changed but slightly over these thousands of years. On the other hand, say we now know that the atmosphere of England was even more persistently overcast thousands of years ago than it is today. We might then conclude that $\varphi(\mathrm{D} \mid \mathrm{I}) > \lambda$, *i.e.*, our prehistoric Englishman caught sight of the celestial bodies so infrequently that the complexity of forming the match with the alignment pattern would be virtually impossible. Now a possible criticism here is that our knowledge of the prehistoric Englishman's weather conditions may be imprecise. In this case we may only be able to determine a range for $\varphi(\mathrm{D} \mid \mathrm{I})$. All that would be necessary then is to ensure that the upper end of the range is less than λ, the complexity tractability upper bound.

Furthermore, even if the complexity happens to be tractable, *i.e.*, $\varphi(\mathrm{D} \mid \mathrm{I}) < \lambda$, let us say that only one stone is involved in the alignment pattern. This would mean that P(D*|H) would be quite high, since the random spatial arrangements of only one stone are relatively few compared to the random spatial arrangements of N stones with $N >> 1$. Therefore, it is conceivable that $\mathrm{P(D^*|H)} > \delta$, particularly since δ is such a conservatively small number, even though $\varphi(\mathrm{D} \mid \mathrm{I}) < \lambda$.

3 CHANCE ELIMINATION ARGUMENT TRANSFORMED INTO A NIM

3 Actually if D* is the event associated with the pattern, D, and the occurrence of E implies that D* also occurred, we say that D *delimits* E rather than D matches E.

4 There are less than 10^{80} elementary particles in the known physical universe, no more than 10^{45} physical state transitions per second, and (assuming "big bang" cosmology) no more than 10^{25} seconds of time will ever be available in the known physical universe. So $10^{80} \times 10^{45} \times 10^{45} = 10^{150}$ is a liberal measure for the maximum number of possible state transitions of all the particles in the known universe throughout the total lifetime of the universe.

Our goal is to apply the GCEA to the problem of developing a NIM. The GCEA is intended simply to *eliminate* the chance hypothesis from consideration, whereas we wish to discover a metric. However, in the process of eliminating chance, the GCEA has to compute a measurable quantity, namely, the probability of the pattern (transformed into event space) conditioned by the chance hypothesis, P(D*|H). The GCEA is also intended merely to compare P(D*|H) against a threshold δ, in order to (possibly) eliminate the chance hypothesis and infer intelligent agency. However, the artificial systems we normally wish to measure are those that we know intelligent agents have designed *a priori*, so no threshold comparison is needed. Probabilities are already calibrated, having a target range in the interval, [0,1]. The GCEA computes a complexity measure, $\varphi(\mathrm{D}\,|\,\mathrm{I})$, comparing it with λ, in order to ensure that the pattern is tractably constructible from the side information. Even though we know that our system is designed, it is certainly possible that it cannot be executed. Therefore, the role of the complexity measure should remain the same for the IS measure. Finally, the GCEA computes a *probability* and we are interested in a complexity measure. Since regularity can be easily eliminated from consideration, the only other options are chance or intelligent agency. Therefore, to the degree our candidate IS is not attributable to chance, the IS displays intelligence. Furthermore, the information theoretic method for converting a probability into an information measure is to take the negative of the logarithm base 2 of the probability. Since we've defined information as the specified complexity of the minimal representation of the IS, $-\log_2 \mathrm{P}(\mathrm{D}^*|\,\mathrm{H})$ is the NIM we seek. So a larger NIM means higher system intelligence and *vice versa*.

Therefore, the NIM consists the following steps: 1) identify E as the design of the IS of interest, 2) gather all the relevant side information, I, in order to identify the pattern, D, 3) identify the event D* that delimits E, 4) define the chance hypothesis, H, for the event D*, 5) compute the probability, P(D*|H), that D* might occur according to H, 6) compute the tractability bound, λ, by quantifying the complexity of forming the pattern with the minimal amount of resources, 7) compute the complexity of generating the pattern, D, from I, 8) test if $\varphi(\mathrm{D}\,|\,\mathrm{I}) < \lambda$, if true, then continue to next step, or if false, stop (it has no intelligence if it is impossible to construct and/or execute), 9) compute $-\log_2 \mathrm{P}(\mathrm{D}^*|\,\mathrm{H})$.

How do we identify E? This is simply the particular system representation we wish to analyze. However, the system design needs to be in an analyzable format. This begins with a need for some unambiguous syntax that can be represented in something like the extended Backus-Naur form (EBNF) [8]. How do we identify I and D? The relevant semantics will be captured in the side information, I, and the gathering and identification of side information is expected to be a challenging task. The challenge should lie mostly in the difficulty in assembling all the applicable semantic information. In fact, we can easily underestimate the intelligence in the system if we miss or overlook key information. Going back to the Stonehenge example, if we mistakenly conclude that the prehistoric man is inobservant and therefore is completely ignorant of eclipse events, but the eclipse alignments actually match with several alignment of stones, we will conclude that the stones display less intelligence in their particular arrangement than they actually do. This is a false negative, which will be very common. However, because complex specified information is typically highly differentiated to other complex specified information (*e.g.*, Schubert's music is very different from J.S. Bach's), false positives will be highly unlikely. The use of formal and standard system specifications [9] should also be used in order to define what is meant by the syntax. Therefore, intelligence in the system will sometimes go unnoticed. Different than the Stonehenge example, system designers will want to ensure that the systems they design achieve the maximum (intelligence) value under the NIM. This is called design for analysis and is done all the time now, though more for performance metrics, since they are most common. However, the current practice of formal testing can be generally considered in the same category as the NIM [10].

4 OBJECTIONS TO THE NIM

A possible objection to the NIM is that a system such as a neural net or an evolutionary algorithm starts out without a lot of intelligence, but as it learns, it grows in intelligence. Recall that we specifically do not define intelligence as the ability to realize certain tasks, but rather as the specified complexity inherent in the system at design time, prior to execution. The mere *ability* to learn should be considered intelligent, independent of, or prior to, having learned anything. The simple acquisition of knowledge, even behavioristic knowledge, should not be counted as an increase in intelligence. For example, a person with advanced Multiple Sclerosis may be intelligent even though he or she may neither see, speak, nor move.

Another objection to the NIM is that certain intelligent systems, particularly learning and optimization approaches such as neural nets or evolutionary algorithms actually effect an increase in specified complexity as the system executes, because it solves complex optimization problems without front-loaded intelligence. However, it can be shown that through the designer's choice of the particular fitness functions, system structure, and initial populations, specified complexity is entered surreptitiously [11]. The No Free Lunch theorems are proof of this, since, for any particular optimization algorithm, if the fitness functions

are not constrained within a certain domain, but are allowed to freely range throughout the entire domain of possible fitness functions, the optimization algorithm will not perform on the average any better than blind search [12]. So specified complexity can only be added by intelligent agents [13].

Another response to this objection is largely experiential. The author has now some fifteen years experience in the practical design of large-scale control systems for mining vehicles, inspection machines, and on-road autonomous vehicles. The consistently strong impression is how much sophisticated, up-front design intelligence (from the intelligent agent) is needed to generate a successful system. Furthermore, the maintenance and improvement of the system also requires enormous input of human intelligence, very far from the easily and powerfully evolving systems that are promised. Perhaps though, we have been using the wrong design paradigm (we have been using a hierarchical paradigm [14]). Because of the No Free Lunch Theorems, we suspect that switching to another design paradigm will make little difference in this. Furthermore, as a working intelligent system designer, we have, of course, had substantial interaction with other designers using other paradigms. Based on this informal testimony, we have no reason to believe that the other paradigms require *substantially* less input of external intelligent agency, both at design time and after.

Yet another objection is that performance intelligence metrics are easier to obtain and furthermore are all that is needed. Besides the NIM is too hard to compute and obtain, *i.e.*, too much work is required to make it worthwhile. This could be argued against linear system theory, which certainly requires a substantial mathematical and physical sophistication, and very few would argue that it has not been of substantial value to control system design. How is linear system theory like the NIM we propose? Linear system theory supplies an analytical theory and allows for metrics for measuring native capability, such as stability, controllability, and observability. Similarly, NIM is based on probability, complexity, and statistical theory and provides its own system capability (or intelligence) metric. As to the objection that the NIM is too hard to compute and obtain, just in the same way that the effort of converting homogeneous differential equations into state space form allows us to apply metrics, converting our IS into an appropriate format, should also allow ease of analysis.

One might argue that elegant and simple designs performing some function should be considered more intelligent than more complex designs that perform exactly the same function. In this case we can consider as side information some metric like "design efficiency" that will increase $-\log_2 P(D^*|H)$ for the more elegant design.

Will the NIM perform well given a complex but useful system as well as an equally complex system but one that simply thrashes, performing no useful task? The key here is that $P(D^*|H)$ will be higher in the latter system, since the thrashing system is not specified even though it is complex. The transformed pattern, D*, will not be supported by side information, I, and so the chance hypothesis, H, will have increased support. Additionally, how will the NIM perform given two other systems, one buggy and one bug-free? Again the system with the bugs will fail to support the side information (that is, the side information affected by the bugs) that indicates increased intelligent agency, giving the chance hypothesis increased support.

Finally, it will be argued that chance, regularity, and intelligence do not cover the entire space of possibilities and particularly that intelligence is not the only conclusion when there is specified complexity. Both Kaufmann [15] and Davies [16] argue that there must be an (as yet) undiscovered regularity that can generate specified complexity. Since this is not yet discovered and since there is little concrete evidence for its existence, it certainly is not unreasonable to maintain our claim.

5 CONCLUSIONS AND FUTURE DIRECTION

We make no claim that the determination of the actual value of the NIM will be easily gotten, particularly as the systems under analysis become more complex. The precise determination of side information, the matching pattern, and the event probability under the chance hypothesis will be challenging to quantify with reasonable hope of accuracy. Clearly, many examples, starting with known and relatively simple systems (such as linear systems), should be attempted in order to exercise this metric.

We may also discover that such a metric (or any intelligence metric, for that matter) cannot measure certain broad classes of systems. This would be disappointing, but should not stop us from pursuing a metric or even surprise us. The twentieth century has been known for the discovery of a wide variety of limitations. Special relativity posited a limiting speed for matter. Heisenberg discovered a fundamental uncertainty in measurement capability (infinite measurement accuracy in position and momentum simultaneously of a particle is impossible). Chaos theory realized that the trajectories of certain deterministic systems cannot be accurately predicted without infinite precision in the initial conditions. A broad class of problems is either not computable (like the general tiling problem) or unsolvable given our computing resources (like NP-hard problems). All these are extremely helpful discoveries, even if somewhat disappointing. At a minimum, they lessen inflated claims of performance by quantifying performance limitations. Perhaps some similar fundamental limitation for any intelligence metric is also

inherent in some broad class of artificial systems. To find such a limitation and to parameterize it would be a worthwhile discovery.

REFERENCES

[1] Barry Commoner, "Unraveling the DNA myth: The spurious foundation of genetic engineering," Harpers, February, 2002.

[2] Columbia Tristar Pictures

[3] Roger Penrose, "The Emperor's New Mind: Concerning Computers, Minds, and the Laws of Physics," Oxford University Press, 1989.

[4] Michael Sipser, Introduction to the Theory of Computation, PWS Publishing, 1997.

[5] Claude E. Shannon, "A mathematical theory of communication," Bell System Technical Journal, Vol. 27, July and October, 1948.

[6] John C. Knight, Colleen L. DeJong, Matthew S. Gibble, and Luis G. Nakano, "Why are formal methods not used more widely?" Fourth NASA Formal Methods Workshop, vol. 27, September, 1997.

[7] William A. Dembski, The Design Inference: Eliminating Chance Through Small Probabilities, Cambridge University Press, 1998.

[8] ISO/IEC 14977:1996(E), Extended BNF, ISO, Geneva, Switzerland, 1996.

[9] John A. Horst, Elena Messina, Thomas Kramer, and Hui-Min Huang, "Precise definition of software component specifications," Ghent, Belgium, 1997.

[10] Paul E. Black, Kelly M. Hall, Michael D. Jones, Trent N. Larson, and Phillip J. Windley, "A brief introduction to formal methods," IEEE 1996 Custom Integrated Circuits Conference, 1997.

[11] William A. Dembski, No Free Lunch: Why Specified Complexity Cannot Be Purchased without Intelligence, Rowman & Littlefield, 2002.

[12] David H. Wolpert and William G. MacReady, "No free lunch theorems for optimization," IEEE Transactions on Evolutionary Computation, vol. 1, no. 1, April, 1997.

[13] Douglas Robertson, "Algorithmic information theory, free will, and the Turing test," Complexity, vol. 4, no. 3, 1999.

[14] J. S. Albus and A. M. Meystel, Engineering of Mind: An Introduction to the Science of Intelligent Systems, John Wiley and Sons, 2001.

[15] Stuart Kaufmann, At Home In The Universe: The Search for Laws of Self-Organization and Complexity, New York: Oxford University Press, 1995.

[16] Paul Davies, The Fifth Miracle, New York: Simon and Schuster, 1999.