Proceedings of the 2003 Performance Metrics for Intelligent Systems (PerMIS) Workshop,
Gaithersburg, MD, August 16 - 18, 2003.

Performance Measures for Intelligent Systems: Measures of Technology Readiness[1]

PERMIS'03 White Paper

1. From Theory to Practice

Previous iterations of PERMIS Workshop Papers [1-7] contain the outlines of the theory of performance measures. PERMIS'03 will begin to examine applications of performance measures to practical problems in commercial, industrial, and military applications. In particular, we will begin to address the issue of technology readiness for practical applications. We believe that it is possible to leverage existing practices and measures from more mature methodologies in other disciplines.

2. Technology Readiness as a Performance Measure

Technology Readiness Levels (TRLs) were initially proposed by NASA in 1995. Sporadic use within the US Science and Technology community followed. They were adopted by the US DoD in June 2001 where they are now mandated for all major Acquisition programs [8, 9].

The stages of technology readiness appear initially to be a check-list for the stages of system's research, development, design and manufacturing. The TRL-6 scale, however, can be used as a framework for validating the overall quality of the system. Indeed, each consecutive Technology Readiness Level demonstrates at a higher resolution a more concrete scenario of operation. The system of multiresolutional scenarios is a leading approach that can express performance under realistic conditions of uncertainty. Table 1 defines the TRL levels and includes some added commentary in the context of PerMIS and resolution levels.

The realistic conditions of operation depend heavily on the nature of the environment in which a system is required to function. For example, a robot vehicle that is capable of navigating successfully within the confines of a building may be completely unable to function outdoors. An intelligent vehicle that is able to drive on well marked roads may be unable to drive successfully through the woods, or in tall grass or weeds. An intelligent vehicle that is capable of driving on the freeway may be unable to drive on two lane roads with on-coming traffic, or on city streets with intersections and traffic signals. This suggests that Performance Measures for Intelligent Systems must not only measure the behavior of the system, but the characteristics of the environment in which the system must perform.

The U.S. Army has launched a major initiative to field a Future Combat System that will consist of light-weight, air-transportable vehicles that include both manned and unmanned vehicles. A major effort is directed at measuring the technology readiness of technology for unmanned driving, both on-road and off-road in all kinds of weather, on all types of roads, day and night, in the presence of pedestrians, animals, and on-coming traffic.

The military services have developed testing methods for measuring the performance of systems under realistic scenarios. These methods are described in a book entitled Code of Best Practice Experimentation. [10] This code is intended to increase awareness and understanding of DOD needs for experimentation, articulate a useful set of principles for the design and conducting experiments, and provide a scientific foundation for testing. The authors of this book will prepare a workshop on the philosophy and methodology of testing new technologies and measuring their readiness for applications in the real world.

A major series of tests have been conducted to assess the technology readiness of unmanned ground vehicles for off-road driving. A report on this series of tests will be presented at PerMIS'03.

Test courses have been developed for measuring the capabilities of teleoperated and robotic devices for searching buildings for human victims of earthquakes and terrorist bombings [11]. A report will be presented on this activity at PerMIS'03.

Other tests need to be developed to measure the readiness of component technologies. For example, improvements are needed for technologies to measure the capabilities of robots and teleoperated devices for searching buildings, and detecting survivors of earthquakes or terrorist bombings. Methods for measuring the performance of components and subsystems are also needed.

[1] Alexander Meystel, Jim Albus, Elena Messina, and Dennis Leedom contributed to this White Paper

Table 1:Technology Readiness Levels in the Context of Intelligent Systems Performance

Technology Readiness Level	Description
1. Basic principles and broad vision of the system observed and reported	The most general discussion of the system, i.e. the lowest level of resolution in system analysis. It corresponds to the lowest level of technology readiness. The results of this level of analysis are usually presented as paper studies of a system's basic properties. Correspondingly, it is also the lowest level of software readiness. Basic research begins to be translated into applied research and development.
2. Conceptual design of a system and/or technology and its application formulated	Beginning of the system's refinement: resolution grows. Key engineering solutions are proposed, innovations are introduced, key resource limits are chosen. Practical applications are invented and tested. Applications are partially tested, partially hypothesized, and there may be no exhaustive proof or reliable analysis to support the assumptions and visions of the developing team.
3. Thorough theoretical and experimental critical analysis of system's function; detailed characteristic proof of concept	More detail is addressed. Active research and development are initiated. Theoretical studies are conducted in the laboratory targeting physical and/or computational (simulation) validation of analytical predictions for separate sub-systems of the system. Those sub-systems are being scrutinized that are innovative and have not been integrated. Similar active research and development is initiated for the software subsystems. The number of resolution levels must be properly chosen. The programs are written that can validate theoretical predictions for separate software subsystems. Algorithms are tested in laboratory environment or in simulation.
4. Component and/or breadboard validation is conducted in the laboratory environment	All basic subsystems and components are integrated to establish that they will work together. This usually includes ad hoc sub-systems integration. This includes integration of software components are integrated to determine how they will work together. They are relatively primitive with regard to efficiency and reliability compared to the eventual system. System Software architecture development initiated to include interoperability, reliability, maintainability, extensibility, scalability, and security issues. At this point, we are able to check the matching between computational parameters of the algorithms and programs on one hand and the parameters of other components (sensors, actuators) on the other.
5. Component and/or breadboard validation in more realistic relevant environment	Fidelity of breadboard technology increases significantly. The basic technological components are integrated with reasonably realistic supporting elements: it includes "high fidelity" ("high resolution") laboratory integration of software components. Configuration control is initiated. Verification, Validation, and Accreditation (VV&A) initiated. At this point, we have an opportunity to check whether the state-space is tessellated properly, whether the parameters of sampling, or parameters of randomization are proper ones.
6. System/subsystem model or prototype demonstration in a relevant environment	Representative model or prototype system, which is well beyond that of TRL 5, is tested in a relevant environment. Represents a major step up in a technology's demonstrated readiness. Examples include testing a prototype in a high-fidelity laboratory environment or in a simulated operational environment. This stage represents a major step up in software demonstrated readiness. Software support structure is in development. VV&A is in process. At this stage we check the value of parameters such as carrying frequencies, bandwidths, etc.
7. System prototype demonstration in an operational environment	Prototype near, or at, planned operational system. Represents a major growth in resolution comparatively with TRL 6, requires demonstration of an actual system prototype in an operational environment such. Examples include testing the prototype in a test bed aircraft. Software support structure is in place. Software releases are in distinct

	versions. Frequency and severity of Software deficiency reports do not significantly degrade functionality or performance. VV&A completed.
8. Actual system completed and qualified through test and demonstration	The system has been proven to work in its final form and under expected conditions. In almost all cases, this TRL represents the end of the system development. Examples include developmental test and evaluation of the system in its intended application to determine if it meets design specifications. Software has been demonstrated to work in its final form and under expected conditions. In most cases, this TRL represents the end of system development. Examples include test and evaluation of the Software in its intended system to determine if it meets design specifications. Software deficiencies are rapidly resolved through support infrastructure.
9. Actual system proven through successful mission operations	Actual application of the technology in its final form and under mission conditions, such as those encountered in operational test and evaluation. Examples include using the system under operational mission conditions. Actual application of the Software in its final form and under mission conditions, such as those encountered in operational test and evaluation. In almost all cases, this is the end of the last debugging aspects of the system development. The system is used under operational mission conditions. Software releases are production versions and configuration controlled.

3. Uncertainty and Complexity

One of the problems encountered by any intelligent system is dealing with uncertainty and complexity in the environment. This may include the geometric and dynamic uncertainty and complexity. It might involve the number and type of moving objects. It might involve the nature of other agents within the environment. How intelligent are the other agents? Are they friendly or hostile? What are their physical capabilities? What are their intentions? How can these parameters be measured and quantified? Measuring and characterizing the uncertainty and complexity of system's representation of the environment and its elements is another important aspect of performance evaluation

There are a number of ways to approach this problem. For example, control theory addresses questions such as, "How much information about a system's input-output behavior is needed to control it to a specified accuracy? How much identification is required if only rough bounds on time and frequency responses are available a priori." G. Zames links the cost of adequate control for imprecisely modeled systems with the value of the complexity of information processing required to achieve a prespecified accuracy [12]. Zames suggests that Kolmogorov's ε-entropy is a better measure of complexity than anything else. In that paper, as well as [13], an attempt is made to characterize problem complexity in terms of Kolmogorov's definition of ε-entropy.

Kolmogorov was inclined to view entropy as a measure of complexity rather than information, as Shannon did. Uspensky reflects on ε−entropy as follows: "complexity of things (as opposed to the complexity of processes, e.g. of computational processes) took the name descriptional complexity, or Kolmogorov complexity ... in appropriate cases one may say 'entropy' instead of complexity [14]."

Kolmogorov was taking into account objects and encodings of objects. The complexity of an object is the minimal size of its encoding. But the encoding of the object is related to how it will be used; the complexity is therefore application-dependent. Thus, there is a set of objects, Y, with elements y, and a set, X, of descriptions (encodings) x. A mode of description is an arbitrary set $\boldsymbol{E} \subseteq \mathbf{X} \times \mathbf{Y}$. If $\langle \mathbf{x},\mathbf{y} \rangle \in \boldsymbol{E}$, then x is called an encoding of y with respect to E. Thus, an object y may have many encodings and a description may serve as a description for many objects.

Kolmogorov complexity has proven to be useful for evaluation of encodings (approximations) of functions specified to a particular precision, ε. The approximation of functions using lower dimensional subspaces has been explored extensively, and developments are reported in, for instance, [15,16]. Some ideas from Kolmogorov and Tihomirov's 1959 paper on ε-entropy [17] demonstrate their applicability to the measuring of performance of the intelligent systems.

Kolmogorov presents his version of entropy as being "of interest in the non-probabilistic theory of information in the study of the necessary size of memory and the number of operations in computational algorithms." Using ε-entropy for complexity evaluation was demonstrated for a multiresolutional intelligent system [18]. We can anticipate that by using computational complexity as one of the performance measures we can improve the existing system of performance evaluators (metrics).

5. Psychophysical Approaches

Many psychophysical experiments are designed to measure the ability of biological intelligence to analyze information in the presence of uncertainty and complexity. It is this emphasis on measurement within a realistic and complex setting that can provide understanding of how to approach measurements of artificially intelligent systems.

6. Biometric Approaches

Two categories of biometric techniques should be taken into consideration: physiology based and behavior based. These can provide guidance in terms of how to deal with error tolerances and manage large knowledge bases in systems that have tremendous richness and variability, as intelligent systems are anticipated to exhibit. Physiology based techniques measure physiological characteristics such as patterns in fingerprints, the iris, facial characteristics, geometry of the hand, vein patterns, the shape of the ear, body odor, DNA analysis, and sweat pore analysis. The behavioral based techniques measure the parameters such as: handwritten signature analysis, keystroke analysis, and speech analysis.

There are two basic concerns in these technologies: the error tolerance and the storage of the templates. It is these factors in particular that may provide insight into performance evaluation of intelligent systems. The error tolerance of these systems is critical to their performance. Both errors (False Rejection and False Acceptance) should be low, and they should both be determined together with the manufacturers of sub-systems (components). Both errors may not incur the same costs, so often they need to be weighted in measures. There is often a tradeoff between the two, which makes the weighting factors very relevant.

The recorded biometric measurements of a user (templates) can be stored in various places depending on the application and the security requirements of this application. The templates can be stored in the biometric device, in a central knowledge base or in portable carriers.

Reliability and acceptance of a security system depends on how the system is protected against threats and its effectiveness to identify system's abuses. There are various sources of threats that the biometric technologies face. They can fall into three main categories: physical, human, and technical.

6. Linguistic approaches

One of the characteristics of intelligent systems is the ability to communicate. Therefore, the question of how to measure the performance of communication between intelligent systems arises. What is communicated? How is the information encoded? What is the bandwidth required? How effective is the communication? How useful is the information to the sender and the receiver? How secure is the channel (can communications be prevented from being intercepted by unwanted listeners?)

7. Future Directions

As a result of Advisory Board discussion and interchanges at the workshop, we will enhance the list of topics for performance evaluation. We look forward to additional ideas from the Advisory Board. We can anticipate an interest in using competitions and test courses for evaluation. There are different views on this matter: that competition is an integrator of multiple performance measures, and that competition gives necessarily skewed results. At least a part of competitions is linked with using subjective evaluations; the latter are broadly used beyond the domain of competitions. How should we deal with this subjectivity?

8. References

1. A. Meystel, J. Albus, E. Messina, J. Evans, D. Fogel, W. Hargrove, "Measuring Performance and Intelligence of Systems with Autonomy: Metrics of Intelligence of Constructed Systems," Eds. A. Meystel, E. Messina, *Measuring Performance and Intelligence of Systems: Proceedings from the 2000 PerMIS Workshop*, Gaithersburg, MD, 2000, pp. 3-34
2. E. Messina, A. Meystel, L. Reeker, "PERMIS 2001,White Paper: Measuring Performance and Intelligence of Intelligent Systems," Eds. E. R, Messina, A. M. Meystel, *Measuring the Performance and Intelligence of Systems: Proceedings of the 2001 PerMIS Workshop*, September 4, 2001, NIST Special Publication 982, Gaithersburg, MD, 2001, pp. 3-15
3. E. Messina and A. Meystel, Eds., *Proceedings of the 2002 Performance Metrics for Intelligent Systems Workshop*, NIST Special Publication 990, Gaithersburg, MD, 2002.
4. J. Albus, "Metrics and Performance Measures for Intelligent Unmanned Ground Vehicles," Eds. E. R, Messina, A. M. Meystel, *Measuring the Performance and Intelligence of Systems: Proceedings of the 2002 PerMIS Workshop*, August 13-15, 2002, NIST Special Publication 990, Gaithersburg, MD, 2002, pp. 61-68.
5. A. Meystel, "Performance of Planning Systems," Eds. E. R, Messina, A. M. Meystel, *Measuring the Performance and Intelligence of Systems: Proceedings of the 2002 PerMIS Workshop*, August 13-15, 2002, NIST Special Publication 990, Gaithersburg, MD, 2002, pp. 99-104
6. L. Zadeh, "In Quest of Performance Metrics for Intelligent Systems – A Challenge That Cannot Be Met With Existing Methods," Eds. E. R, Messina, A. M.

Meystel, *Measuring the Performance and Intelligence of Systems: Proceedings of the 2002 PerMIS Workshop*, August 13-15, 2002, NIST Special Publication 990, Gaithersburg, MD, 2002, pp. 303-306

7. L. Reeker, A. Jones,"Measuring the Impact of Information on Complex Systems", Eds. E. R, Messina, A. M. Meystel, *Measuring the Performance and Intelligence of Systems: Proceedings of the 2001 PerMIS Workshop*, September 4, 2001, NIST Special Publication 982, Gaithersburg, MD, 2001, pp. 127-136
8. AMS Guidance on Technology Readiness Levels (TRLs), see URL http://www.ams.mod.uk/ams/content/docs/trlguide.doc
9. J. C. Mankins, Technology Readiness Levels: A White Paper. Office of Space Access and Technology, NASA, http://www.hq.nasa.gov/office/codeq/trl/trl.pdf, April 6, 1995.
10. D. S. Alberts, R. E. Hayes, J. E. Kirzl, D. K. Leedom, and D. T. Maxwell, *Code of Best Practice Experimentation*, Command and Control Research Program, Washington, D.C., 2002.
11. A. Jacoff, , E. Messina, J. Evans, "Performance Evaluation of Autonomous Mobile Robots," Industrial Robot 29:3, May 2002.
12. G. Zames, "On the Metric Complexity of Causal Linear Systems: ε-Entropy and ε-Dimension for Continuous Time," *IEEE Trans. Automatic. Control*, vol. AC-24, No. 2, April 1979, pp. 222-230.
13. A. Tannenbaum and Y. Yomdin, "Robotic Manipulators and the Geometry of Real Semialgebraic Sets," *IEEE J. Robotic. Automat.*, vol. RA-3, no. 4, August 1987, pp 301-307.
14. V. A. Uspensky, "Complexity and Entropy: An Introduction to the Theory of Kolmogorov Complexity," in "*Kolmogorov Complexity and Computational Complexity*", O. Watanabe (ed.), Springer-Verlag, 1992.
15. G. G. Lorentz, "*Approximations of Functions*," Holt, Rinehart and Winston, 1966.
16. A. Pinkus, "*n-Widths in Approximation Theory*," Springer-Verlag, 1985.
17. A. N. Kolmogorov and V. M. Tihomirov, "_-Entropy and _-Capacity of Sets in Function Spaces," *Uspehi Math. Nauk*, vol. 14, no. 2(86), pp 3-86, 1959. (in English it is published in *American Math Society Translations*, Series 2, vol 17, pp 277-364).
18. Y. Maximov, A. Meystel, "Optimum Design of Multiresolutional Hierarchical Control Systems", Proc. of the IEEE Int'l Symposium on Intelligent Control, 11-13 August, 1992, Glasgow, Scotland, UK, 1992,