# System of Systems Architecture Evaluation Using Evolutionary Computation

Joseph J Simpson
Missouri University of Science and Technology
6400 32nd Avenue N.W., #9
Seattle, WA 98107
(206) 789-7089
jjs-sbw@eskimo.com

Dr. Cihan H. Dagli
Missouri University of Science and Technology
229 Engineering Management and System Engineering Department
Rolla, MO 65409-0370
(573) 341-4374
dagli@umr.edu



**Abstract.** Evolutionary computation and evolutionary algorithms represent a developing science and technology that can be effectively applied to the generation and evaluation of system of systems architectures. A general technique used by systems engineering professionals is a binary matrix representation of a system or system of systems. The specific meaning and semantics of the binary relationship depends of the type of representation used. Typical representations are, "N squared", design structure matrix, dependency structure matrix, and implication matrix. A key feature of these typical representations is their direct relationship to the structure required in an evolutionary computational approach. Evolutionary algorithms can be applied to the evaluation and optimization of these matrix structures. A new evolutionary algorithm has been developed that applies specifically to the generation and evaluation of systems and system of systems. This new evolutionary algorithm incorporates a fuzzy inference system in the calculation of the best fit evaluation. The current industrial and social environment is populated with a vast array of existing and developing systems. Any new system must take this context into account. Evolutionary computation is applied to assist the system architect and engineer in the evaluation of these complex configurations and interface sets.

The new evolutionary computing techniques are applied to system of systems architecting tasks using a well defined set of measures of effectiveness (MOE). The systems architecting task is divided into three general areas organized around the roles and responsibilities associated with the system architect, the system customer and the system engineer. The system architect is responsible for the complete system operation and MOE balance, focused on life-cycle cost and risk. The customer is responsible for the mission profile and mission functions. Operational effectiveness and operational suitability areas are the responsibility of the systems engineers. Affordability, risk, operational effectiveness and operational suitability are the four MOE used to evaluate the candidate system of systems architectures.

**Introduction:** Systems engineering classically uses measures of effectiveness to evaluate total system performances (Goode and Machol, 1956). The system architecture performance measures in this approach are defined and used in the context of classical systems engineering measures of effectiveness. System effectiveness is defined as a quantifiable measure of the degree to which the candidate system under evaluation is expected to meet and/or perform the stated mission need and objectives using the provided mission profile. It is important to select and match the measure of effectiveness with the given customer problem. The system customer provides the mission context and the mission function architecture which represents the decomposition of the mission function that needs to be performed. During the architecture generation and evaluation process, each candidate system architecture is evaluated to determine how well the system architecture will perform the required mission function as well as to consider other system architecture factors, such as suitability, cost and risk. The primary measure of effectiveness that can be used is system effectiveness versus life cycle cost (Dahlberg, 2004). This classical system MOE is composed of four sub-components: operational effectiveness, operational suitability, life cycle cost and risk. These four sub-components are selected to provide logical and semantic consistency in the development and use of this specific MOE. The operational effectiveness sub-component is focused on how well the candidate system under evaluation meets the complete mission function. The operational suitability sub-component considers the specific physical and design features associated with a specific physical system solution. The life cycle cost component is focused on the system cost evaluation while the risk component considers all sources of risk associated with a specific candidate system. The classical system MOE approach has been modified to accommodate the unique attributes of fuzzy numbers, fuzzy inference systems and evolutionary algorithms.

## System Evaluation Using Fuzzy Measures of Effectiveness

Every evolutionary algorithm must incorporate a fitness function that evaluates the current solution population and selects the best solutions for use in the production of the next generation of solutions. A fuzzy inference system is used as the fitness function for the evolutionary algorithm discussed in this paper. A diagram of the evolutionary computation process is shown in Figure 1, which outlines the basic steps of evolutionary computation.

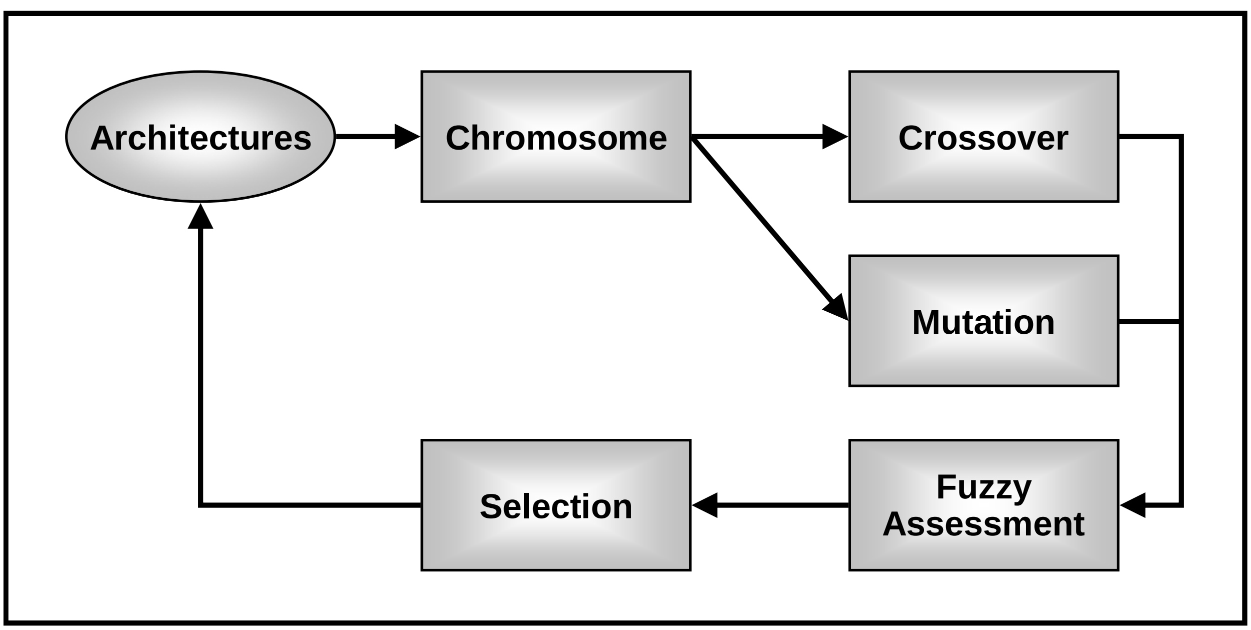


Figure 1. Evolutionary Computation Process

The fuzzy measures of effectiveness are used in the fuzzy assessment as well as the selection steps of the evolutionary computation process. All four subcomponents of the system measure of effectiveness have been modified to integrate with the application of fuzzy logic. The primary modifications are associated with replacing “percentage measures” with “degree of membership” measures as well as the inclusion of a weighting mechanism to vary the weights and impacts of each subcomponent on the final system measure of effectiveness. The context for the application of this set of modified measures is shown in Figure 2, ‘General Architecture Development Context.’

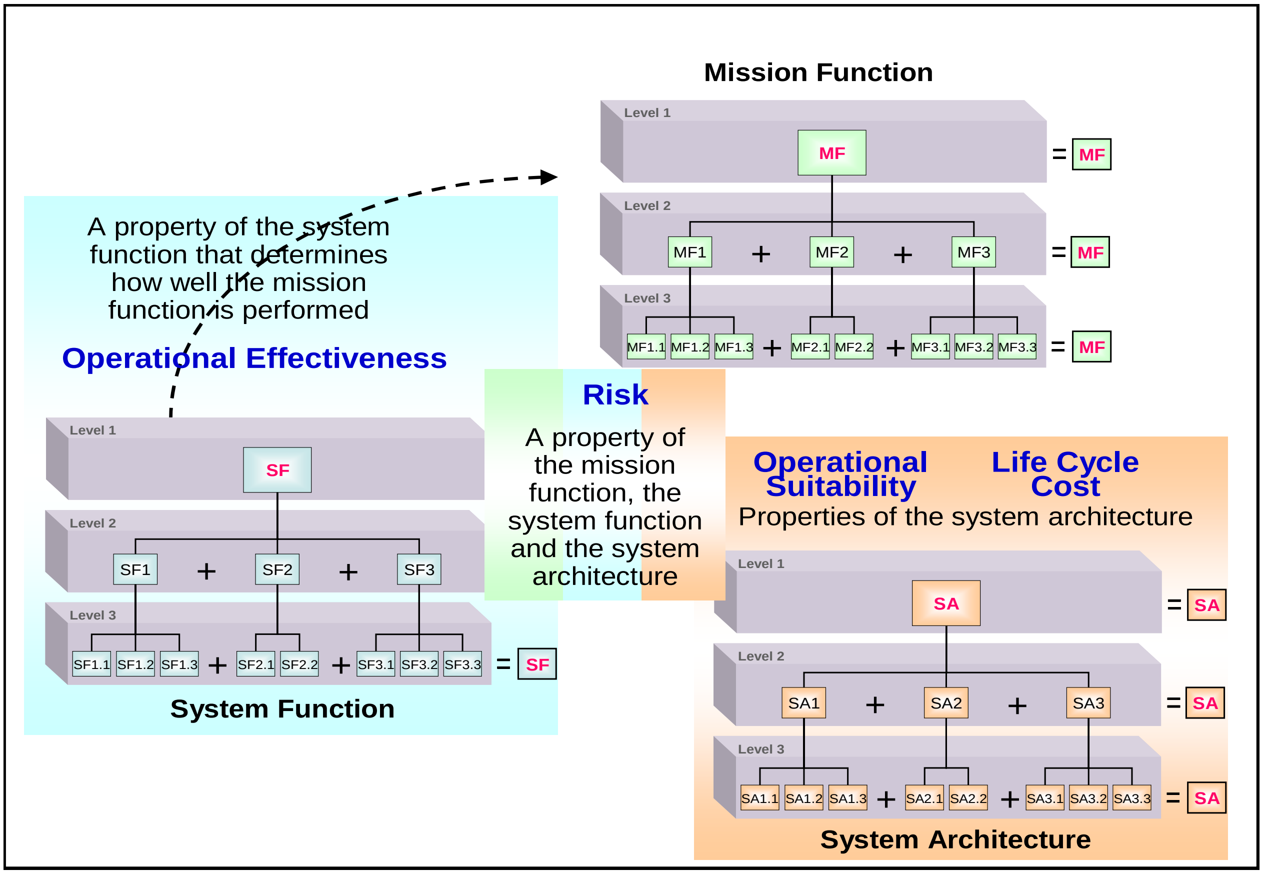


Figure 2. General Architecture Development Context

The four primary fuzzy measures are operational effectiveness, operational suitably, risk and affordability. While the customer and the system architect may apply weighing factors to these measures, operational effectiveness usually has the highest impact on the final metric. The operational suitability measures are associated with the physical architecture that performs the system functions. The operational effectiveness measure is associated with the degree to which system functions provided by the physical system architecture, fulfill the stated customer mission functions while operating in an environment specified by the given mission profile. The affordability measure records how well the total system life-cycle cost meets the customer production and operational cost budgets. Risk is the final effectiveness measure that is used to quantify the technical maturity, cost, schedule, political and operational risks associated with the current architecture under evaluation. The system physical architecture is evaluated at a segment and/or component level to determine how well the group of selected physical components and/or segments will fulfill the stated customer need. In the case of system of system architecture evaluation, the segments and/or components are independent systems in their own right. The fuzzy measures of effectiveness are shown in Figure 3, Fuzzy Measures of Effectiveness.

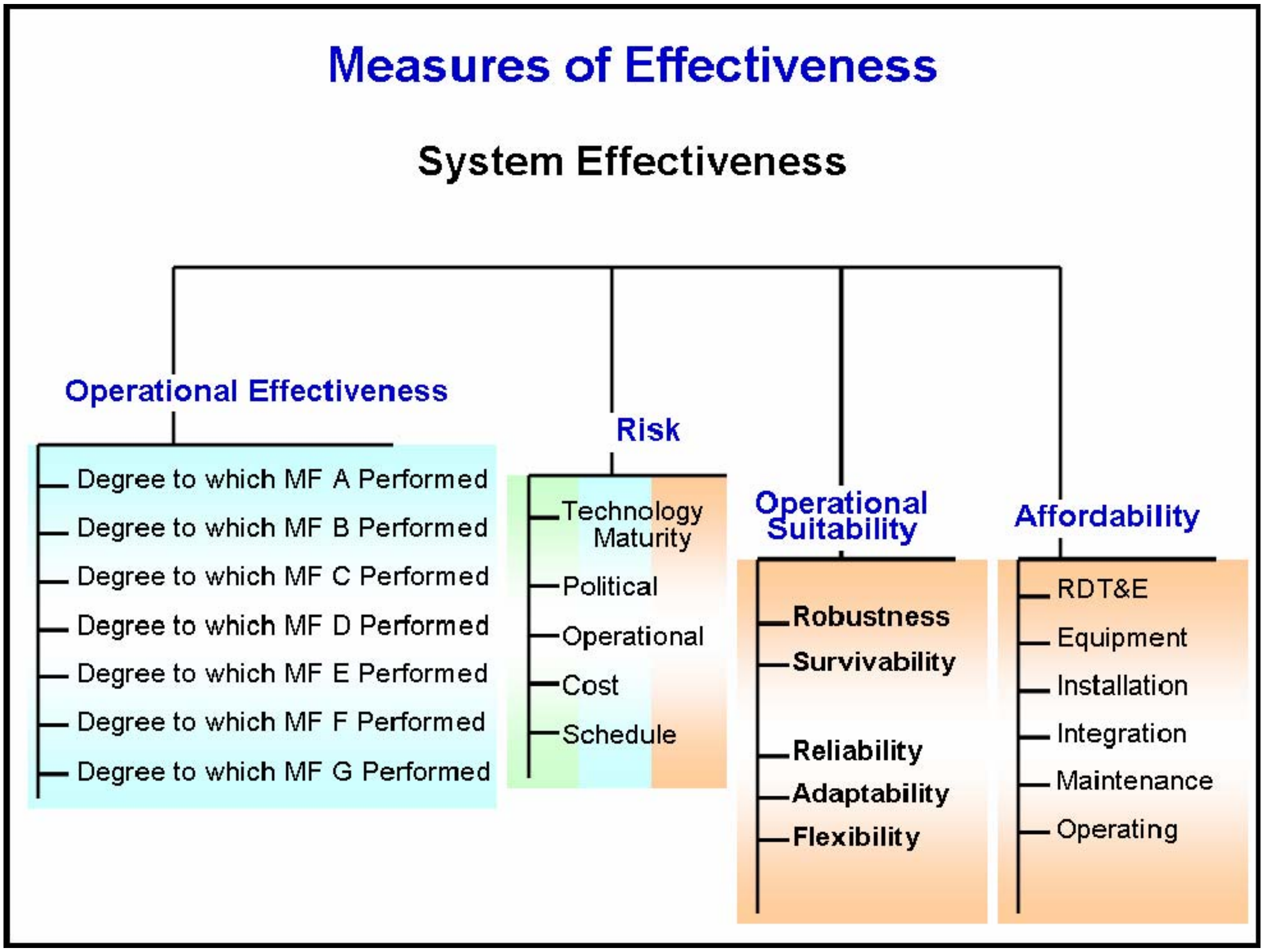


Figure 3. Fuzzy Measures of Effectiveness

## System Description Using Formal Concept Analysis

A fundamental set of design decisions associated with the development of evolutionary algorithms and the application of evolutionary computational techniques is the selection of the system phenotype representation as well as the system genotype. The phenotype (phenomenon) is how the system structure is represented in the real context. The genotype (gene or chromosome) is the structure of how the system is represented as chromosomes in the evolutionary algorithm. These design decisions are supported by the structured form of classical system representations frequently used by systems engineers. Many of these structured representations of systems use a vector and/or matrix format that is populated with a series of ones and zeros. The more general design question facing systems engineers that wish to use evolutionary computing and evolutionary algorithms relates to the semantic value associated with each phenotype and genotype representation. While the phenotype is associated with systems engineering artifacts and the genotype is associated with the computing process and artifacts, there is no structured approach to transition between these two artifact types. Formal concept analysis is a branch of mathematics that has been adopted for use in this transition role.

Systems engineering, systems analysis, software evaluation and knowledge development domain areas use a binary, relation-based approach called Formal Concept Analysis (Ganter, Wille, 1999). A formal context is constructed from two sets, A and B, and a relation I between A and B. The elements of set A are called objects and the elements of set B are called attributes of the context. The relation I is called the incidence relation of the context. The formal context structure is used to create a formal concept definition. These binary, relation-based structures have been used in software and knowledge engineering applications. Active research is continuing in the application of this technology in many engineering domains. In this application of evolutionary computing, Formal Concept Analysis is used to organize the data given by the genotype system form into a structured formal context system form. The formal context system form is then used by the fuzzy inference system fitness function to evaluate the total system architecture given in terms of the phenotype representation. Figure 4 shows an example of a system genotype representation mapping to a formal context system representation.

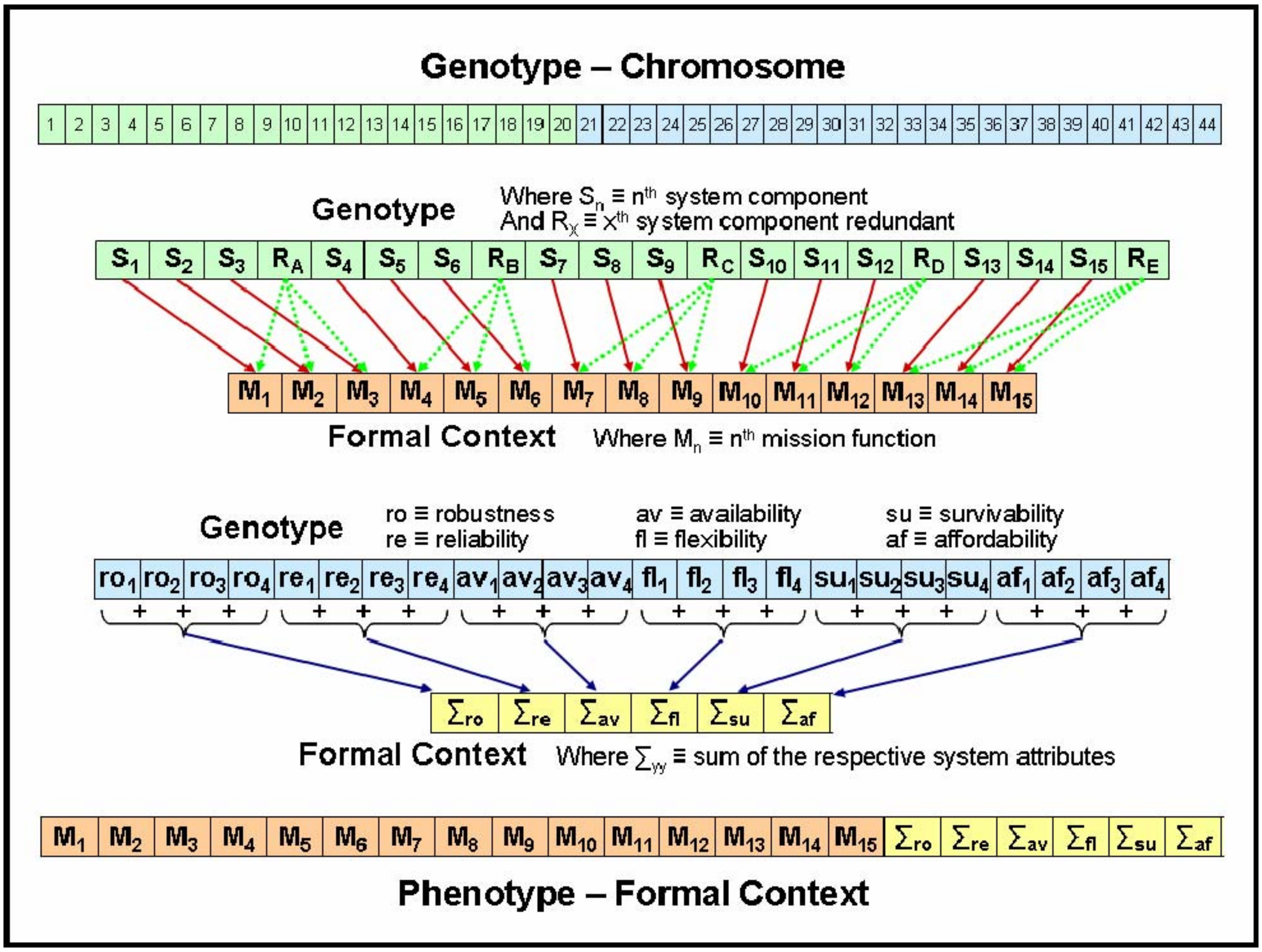


Figure 4. System Genotype to System Formal Context Mapping

In the current system analysis and formulation process, the primary ordering structure for the genotype is the set of physical components that perform the system functions. As shown at the top of Figure 4:

- System component 1 (S1) performs mission function 1 (M1),
- System component 2 (S2) performs mission function 2 (M2), and
- System component 3 (S3) performs mission function 3 (M3), and
- System component redundant A (RA) performs missions functions 1, 2 and 3 (M1, M2, M3).

The first twenty (20) values in the genotype are mapped to the first 15 values in the formal context. The bottom of Figure 4 shows the mapping of the last twenty four (24) entries in the system genotype to the last six entries in the formal system context. The first part of the genotype and formal context encode the system capability to perform the requested mission functions, and to support the evaluation of the system operational effectiveness. The second part of the genotype and formal context encode the values of the following general system attributes: robustness, reliability, availability, flexibility, survivability and affordability. In this section of the genotype, groups of four genes are evaluated to determine the values associated with the last six entries in the formal context.

The affordability value will be discussed first, as it is a value that is determined by summing the life cycle cost of all of the physical system architecture components. In general, if life cycle costs for sub-systems performing system functions used to execute mission functions are low, then the affordability will be high. If life cycle costs for sub-systems performing system functions used to execute mission functions are high, then the affordability will be low. This example of the affordability performance measure shows that there is a logical value relationship that flows from the concept of physical composition, to the evaluation of the physical system architecture. This logical relationship will be used by the best-fit function implemented by the fuzzy inference system.

The other five architectural suitability measures have the same structural connection from the physical system architecture through the system function hierarchy to the mission function hierarchy. However, the robustness, reliability, adaptability, flexibility, and survivability fuzzy performance measures are not independent of each other. Therefore, these fuzzy performance measures may depend on each other in some manner that may be application domain dependent. Figure 2 shows the use of fuzzy measures of performance used in the computation of the system measure of effectiveness. Fuzzy performance measures are designed to achieve the same quantifiable measures of system effectiveness as classical system effectiveness measures, though the computations are significantly different.

The remaining five fuzzy performance measures will be evaluated in two groups. Group one contains robustness and survivability, while group two contains reliability, adaptability and flexibility. Robustness and survivability depend a great deal on the given design mission profile. The design mission profile defines all environmental, threat, and operational parameter values and performance expectations. So, survivability values are given in the context of environmental, operational and other active threats detailed in the mission profile. The robustness fuzzy performance measure indicates the ability of the selected physical system to operate at the margins of, or outside, the operation margins given in the design mission profile. Group one fuzzy performance measures are not necessarily additive or linear like the affordability metric. Each sub-system could be very robust and highly survivable, but the integrated system segment, or total system, could be fragile and highly vulnerable. Therefore, the physical system architecture must be evaluated at each level of physical integration to assure that the robustness and survivability fuzzy performance measures are being properly addressed.

Group two fuzzy performance measures – reliability, adaptability and flexibility – are not necessarily additive or linear, similar to the group one metrics. Unlike the group one metrics, the group two metrics are associated directly with the configuration of the physical system architecture, and how this architecture is used to perform the system functions that support the execution of the mission function.

As an example, a physical system could be architected in a manner that assigns one physical system segment to provide system functions that support the execution of mission functions M1, M2, M3 and M4. Another physical system segment could be assigned the task of providing the system functions that support mission functions M4, M5, M6 and M7. All mission functions, except mission function M4, are single points of mission functional failure. This type of physical system architecture may have adequate system reliability, but would have low adaptability and flexibility values. However, given the same two physical system segments *and the ability of each physical system segment to adapt concurrently to provide support for each mission function in two different ways*, the physical system architecture becomes much more reliable, flexible and adaptable. Further, if the physical system has segments that are flexible and adaptable to support a general class of mission system functions no matter what the specific mission functions are, then the physical system architecture becomes even more flexible and adaptive.

The physical system architectural representation is the foundational construct used to support the development of the system genotype used in the evolutionary computing process. The formal context is the basic mechanism used to transform the genotype representation into a system phenotype representation for evaluation by the fuzzy inference system.

## Fuzzy Inference System Best-Fit Evaluation

A fuzzy inference system is used as the fitness function for the evolutionary algorithm. The six selected fuzzy key performance attributes are mapped directly to two of the measures of effectiveness sub-components: operational suitability and life cycle cost. This mapping was shown in Figure 3. The set of membership functions for each of the six fuzzy key performance parameters are similar, each has a membership value axis ranging from zero (0) to one (1) as well as a variable value range that runs from zero (0) to one hundred (100). The following five specific membership functions are included in each membership function set: very large, large, average, small and very small. Expert judgment and best professional judgment is used to develop and assign specific values associated with the physical system architecture under evaluation and the variable value range component of each fuzzy number. Figure 5 presents the basic fuzzy membership function mapping.

These membership functions are designed to be independent of the specific domain of system development and application. The specific real world values associated with the variable value range (or scale) can change in each real world situation, but this set of membership functions is designed to provide a common "ranking scale" across a large range of system implementations. The general nature of provided system measures make them applicable to almost every application domain associated with engineered systems development and deployment. Other system measures could be utilized that were specific to one system domain or an individual system development project. As stated previously, the membership functions for each of the six fuzzy performance measures all have the same basic form. The assignment of specific values to each fuzzy membership function is based on the evaluation of the candidate architecture by engineering and architectural experts using professional judgment. The fuzzy measures of effectiveness are aggregated at each level of system decomposition to provide a mechanism that communicates the general effectiveness of the total system.

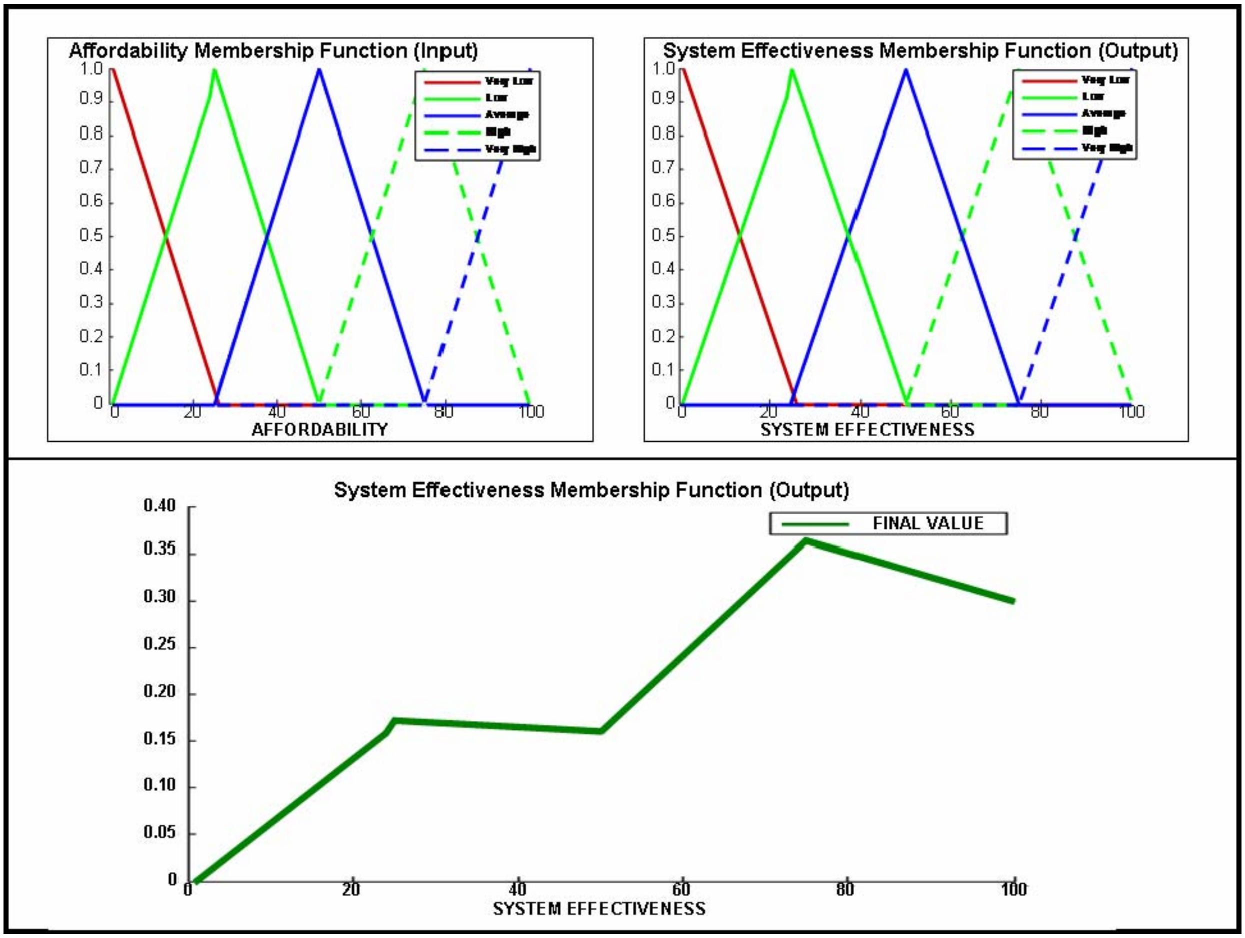


Figure 5. Fuzzy Membership Functions And Final Value

## Summary and Conclusions

Evolutionary computation and evolutionary algorithms represent a developing science and technology that can be applied to the evaluation of systems and system of systems architectures as outlined in this paper. Careful evaluation of classical systems engineering techniques reveals a clear connection between the system phenotype representation and the system genotype representation. Fuzzy logic further expands the application of evolutionary computing techniques by embedding a well defined knowledge management technique into the system and system of systems architecture evaluation process. More research is required to further define specific techniques that can be standardized and documented by a well defined architectural evaluation process. However, any such standardization must be based on the time-tested practices of system measures of effectiveness combined with structured, system decomposition.

## References

Dahlberg, Sten, "Functional Analysis.," Tutorial presented to the Seattle Metropolitan Chapter of the International Council on Systems Engineering (INCOSE, SMC), Seattle, WA, 2005

Ganter, B., Wille, R., *Formal Concept Analysis, Mathematical Foundations*. Springer-Verlag,

New York, 1999.
Goode, H.H. and Machol, R.E., *System Engineering, An Introduction to the Design of Large-scale Systems.* McGraw-Hill Book Company, New York, 1957.

## Biography

Joseph J. Simpson's interests are centered in the area of complex systems including system description, design, control and management. Joseph has professional experience in several domain areas including environmental restoration, commercial aerospace and information systems In the aerospace domain, Joseph has participated in a number of system development activities including; satellite based IP network design and deployment, real-time synchronous computing network test and evaluation, as well as future combat systems communications network design. Joseph Simpson has a BSCE and MSCE from the University of Washington, an MSSE from the Missouri University of Science and Technology, is a member of INCOSE, IEEE, and ACM. Currently Joseph is enrolled in a system engineering doctorate program at the Missouri University of Science and Technology.

Dr. Cihan H Dagli is a Professor of Engineering Management and Systems Engineering and director of the System Engineering graduate program at the Missouri University of Science and Technology. He received BS and MS degrees in Industrial Engineering from Middle East Technical University and a Ph.D. from the School of Manufacturing and Mechanical Engineering at the University of Birmingham, United Kingdom, where from 1976 to 1979 he was a British Council Fellow. His research interests are in the areas of Systems Architecting, Systems Engineering, and Smart Engineering Systems Design through the use of Artificial Neural Networks, Fuzzy Logic, and Evolutionary Programming. He is the founder of the Artificial Neural Networks in Engineering (ANNIE) conference being held in St. Louis, Missouri since 1991. He provided the conduit to the dissemination of neural networks applications in engineering and decision making through these conferences for the last fourteen years. He is the Area editor for Intelligent Systems of the International Journal of General Systems, published by Taylor and Francis, and Informa Inc.