

A unified approach for complex problem solving: Method Integration

Özer Ciftcioglu, Sevil Sariyildiz and Peter van der Veer
Delft University of Technology
Faculty of Architecture and Civil Engineering
Department of Computer Science
Berlageweg 1, 2628 CR Delft, The Netherlands
E-mail: Sariyildiz@bk.tudelft.nl

Abstract

The advances in the computer technology have led to significant productivity enhancement in the information processing technology. This is basically due to increasing utilization of computers virtually in every area of applied sciences that urged efficient and effective information processing for solving the complex problems encountered. One can consider that such problems may be encountered in complex engineering systems. Historically, the computer-aided and/or enhanced applications were initially restricted to algorithmic computations. Algorithmic solutions are mostly systematic solution procedures primarily based on first principles. However, especially in the last decade, in addition to increasing complexity of the tasks to be performed with the special features of diverse engineering disciplines, there appeared new demand for support of algorithmic problem solving methodologies. The new support is provided by a new approach for problem solving stemming from emerging methods and associated technologies.. The algorithmic and non-algorithmic problem solving paradigms can commonly and effectively be combined in such a way that they are applicable for complex problems encountered. This new problem-solving approach applied to complex engineering problems is referred to, here, as *method integration*.

Within the above context, the study identifies the guidelines for method integration and concentrates on those methodologies and technologies supporting the paradigm of method integration with associated problem areas.

1. Introduction

The advances in the computer technology have led to significant productivity enhancement in the information processing technology that is briefly referred to as information technology (IT). This is basically due to increasing utilization of computers virtually in every area of applied sciences that urged efficient and effective information processing for solving the problems encountered. With the feedback of the advances in IT, also a number of new technologies have appeared in the last decade. These new technologies are collectively referred to as emerging technologies that are advanced technologies bringing along their problems. Added to that, the existing technologies have also made rapid advancements with the help of computer technology so that the problems of 90s are quite different and comprehensive than those of 80s. The solutions for the new technological problems required new solution methods since the conventional methods obviously remained inadequate. In this respect, these emerging problems are relatively complex since they belong themselves to complex systems. Although, one may think of complex problems in many different ways and disciplines, in the present context, such problems are deemed to stem essentially from engineering systems, although extensions/extrapolations to engineering related disciplines and areas should not be excluded. For the latter two, Architecture, computer graphics can be referred to as example areas that are potentially important. However, unified approach for solution procedures to be considered here can be considered as encompassing the solution procedures of complex problems of other areas, whereas the degree of applicability may vary.

Historically, the computer-aided and/or enhanced engineering applications were initially restricted to algorithmic computations. Algorithmic solutions are mostly systematic solution procedures primarily based on first principles. As such well-known examples, finite element analysis and for dynamic computations Kalman filtering can be referred. The algorithmic solutions are well-established methods that various mathematical methods are integrated into. Therefore method integration in problem solving is a common approach. However, especially in the last decade, in parallel with increasing complexity of the tasks to be performed with the features of many special domains in diverse engineering disciplines, there appeared new demand in support of algorithmic problem solving methodologies. This is due to the complexity of the problems belonging to advanced technologies. This new support can be provided by the possibilities of the emerging technologies that they were not known or mature enough before. Integration of emerging

methods into the known algorithmic solution methods is commonly referred to, here, as *method integration*, as unified approach for complex problem solving. Such a concept was mentioned earlier [1]. The emerging methods are quite in number but probably they are in majority in form of non-algorithmic in contrast with algorithmic methods. As an algorithmic method multiresolution decomposition with its associated technology, in signal analysis can be mentioned as a typical example. The non-algorithmic methods concern the area of artificial intelligence (AI) which is essentially *Heuristic Complex Problem Solving*. In the AI category one can realize *neural networks*, *fuzzy-logic*, *expert System*, *combinatorial optimization*, *cellular automata* as example areas. The AI with its methods, methodologies, paradigms, have established its technology known as AI technology. By the integration of this new technology into engineering systems, many formidable computational tasks can be reduced to an acceptable size and/or many ill-defined problems can be manageable with “intelligent” solutions. This can take various forms in actual implementations. Utilization of advanced emerging methods for complex problem solving involve, in principle, complex systems like, aviation systems, robotics, power plants, where the latter may be categorized as “large” to distinguish it from others.

This study aims to define the problem areas subject to method integration and to identify generic/general guidelines to establish the paradigm of method integration. It also briefly concentrates on those “emerging” methodologies/technologies subject to method integration, for the completeness. The organization of the paper is as follows. Section II concentrates on the nature of the problem subject to method integration. Section III deal briefly with the “emerging” methodologies/technologies subject to method integration. Section IV, points out the method integration as unified complex problem solving method by referring to the application areas which is followed by conclusions.

2. Complex problems and method integration

2.1. Nature of the problem

Complex problems exist in engineering in various forms. Among them, accurate determination of engineering systems, non-linear systems, non-linear dynamics, data/sensor fusion, non-linear control can be referred, for instance. Such problems exist due to associated complex engineering systems where they can be broadly categorized by distinguishing between engineering systems and large-scale engineering systems or distinguishing

between real-time systems and off-line systems. In these categorizations, the majority of the complex problems are due to complex real-time systems and large-scale systems. Real-time systems are well defined. By complex system we mean those systems which typically involve multiple, interacting processes. Such systems may be viewed from different but equally valid points of view, or described at different levels of abstraction. Complex system problems can require the processing of formidable amount of information that should be accomplished complying with the associated constraints. Complex(-integrated) systems are the systems where human-machine interface is (explicitly) included. However for large-scale systems there has been no accepted (universal) definition for what constitutes a “large scale” system. However, a viewpoint is that a system is large-scale when its dimensions are so large that conventional techniques of modeling, analysis and control design and computation fail to give reasonable solution with reasonable computational efforts. In such systems more than one controller or decision-maker is required. It is commonly accepted that such systems possess the following characteristics:

- Large-scale systems are often controlled by more than one controller or decision maker
- The controllers have different correlated “information” available to them
- They are represented by relatively imprecise models
- They may be satisfactorily optimized by means of sub-optimal or near-optimal controls

Although nature of the engineering problems varies, they commonly concern efficient and effective solutions. Efficiency and effectiveness may be interrelated. Effectiveness can be defined as cost effectiveness where an appropriate cost function is defined according the problem at hand. In this case the engineering problem can be related to the optimization (or minimization) of the cost function subject to some additional constraints in a general case. This is to say that, optimization is a common engineering system issue and in most cases the complex problem is non-linear. Therefore non-linear complex problem solution tools have special importance in method integration where the engineering system may be complex, real-time, large or characterized by any combination of these. However, the *real-time* large or complex systems require additional attention for complex problem solving due to time-critical nature of the problem as described below.

Two crucial aspects dominate efforts to apply method integration to large real-time systems. First, such systems may typically operate in a time-critical manner. Second, the information that can be used to manage

such systems will often contain elements of uncertainty. The time-critical nature of a complex process environment forms the fundamental underlying constraint in the development of managing large real-time systems. Such situation can easily be imagined in aviation or power plant systems, for instance. In each of these systems, the critical nature of operations is exacerbated by the nature of continuous human/machine interactions and by the decision-response characteristics of both the human operators and the advisory intelligent controllers. In these situations, both human and machine must operate in parallel.

During time-critical operations, many tasks must be performed in close synchronization. These include accessing appropriate information, interpreting the state of the operation, and making decisions. Frequently, extensive knowledge resources must be accessed. This can include consulting experts and searching knowledge bases in a time dictated by the dynamics of the process involved. The goal of the method integration is to evolve integrated methods to aid during these crucial, time-critical periods.

The second fundamental characteristic of a complex process environment is the uncertainty of evidence available for situation identification and decision-making. This evidence may consist of data, information and knowledge, and may exist at different levels of abstraction both being at the level of individual process parameters and at the level of the process as a whole. There are many different sources of uncertainty. At the process level uncertainty seems to arise from the dynamics and complexity of the process itself; from incomplete and/or conflicting information; or from multiple and contradictory control goals. At a lower physical level, uncertainty arises from measurement errors, drifting instruments, faulty sensors and/or improperly processed data.

2.2. Method Integration for complex problem solving

In the preceding section, it is pointed out that the application of the method integration is especially required for complex systems. This is due to the complexity of the related problems and integration of the "intelligence" to the solution procedures could substantially enhance the effectiveness of the solution and could provide optimality. Although the concept of method integration can be suitable for any problem size in engineering system, the most beneficial outcomes of the concept become obvious for complex systems where the solutions to problems are not adequately efficient, effective or amenable if one restricted to only the traditional algorithmic methods. For complex problems contribution from especially non-algorithmic approach that deemed to have "intelligence"

could improve the situation substantially. Therefore, for the method integration concept, large, complex and real-time systems are the main concern and the problems of such systems are the emerging most problematic areas nowadays. Aviation, power plants can be mentioned as such dynamic systems, for instance. However this is not to exclude other complex engineering problems within the context we deal with. Examples can also be given from such areas as well, referring to an architectural design or civil engineering realization subject to optimization.

Intelligent method integration can be thought of as a set of tools, concepts, advanced programming techniques and methodologies involved in the solution of problems which if done by human would be said to require "intelligence". Such problems typically cannot be specified using purely traditional mathematical (algorithmic) analysis and intelligent methods are especially powerful to deal with the issues of uncertainty of the available information and/or system representation that arise virtually in all engineering systems.

In order to create a solid foundation for the application of the method integration a wide range of approaches that are associated with artificial intelligence must be brought to synergistic interplay. Disciplines which contribute to complex problem solving using method integration approach include neural networks, pattern recognition, robotics, fuzzy systems, genetic algorithms, signal interpretation, real-time learning, adaptive control, decision theory and resource optimization. AI concepts and methodologies, as developed for expert systems and knowledge-based systems, contribute methods for high-level knowledge representation and decision-making. Natural language systems and machine vision systems, coupled with smart sensor technology, provide the basis for autonomous system operation and human/machine interfaces. Fuzzy logic methods enable systems to deal with ambiguity and failures.

In time-critical dynamic complex systems, the crucial problems involve a variety of considerations that follow from system monitoring, state (or fault) diagnosis and control. These include deciding which protection systems to activate and what order to activate them in, what operations or actions the supervisors should perform and determining the appropriate sequence of actions. Certain operations require that intelligent systems emulate even human cognitive capabilities. The ultimate test of the correctness of the decisions will be the efficacy of their actions in maintaining the plant, process, or operation in a desired state. To meet these challenges, method integration tools must be identified with a view of the computer as an information and knowledge instrument, operating as a colleague or intelligent advisor. Here, the goal is to add flexibility and

increased effective control to the human operator, as well as to improve the safety and efficiency of system operation.

3. Advanced technologies for complex problem solving

The solution of complex engineering problems is reliant on access to information and the use of advanced technologies for processing and analysis to a degree unprecedented before. A variety of technologies that emerged in the past quarter century in the physical sciences and engineering seem to have provided the means of bringing complex systems under control. However, it is only in the last decade that the extraordinary advances in computer hardware and software have made possible the implementation of *advanced technologies* that are essential for the solution of the problems of complex (engineering) systems.

In particular, advanced technologies are necessary to perform a number of tasks that are essential to intelligent problem solving. These tasks include surveillance, diagnostics, communication, modeling, information handling (access, processing, analysis, transmission, storage and retrieval), optimization and decision making. Certain advanced technologies are critically important to only one of the above tasks; others assist with many of the tasks. Below, we review several of the more important advanced technologies that are necessary for intelligent problem solving of complex (engineering) systems

3.1. Artificial neural networks

Artificial neural networks ANN are arrays of highly interconnected simple processing elements called artificial neurons, arranged in architecture inspired by the structure of brain. They attempt to emulate the information processing capability of the brain. Neural networks may be designed to classify input patterns in predetermined classes or to create categories that group patterns according to their similarity. Perhaps the most important characteristic of ANN is their ability to model processes and systems from actual data. The neural network is supplied with data and then trained to represent the input-output relationships of the processes or system. Neural networks have the ability to respond in real-time to changing system state descriptions provided by continuous sensor (or other) inputs. ANN may be used both as classifying and clustering systems. To perform classification, it is necessary to attach to each signature a label that

describes the operational state of the system at the time of collecting the data. The input to the ANN is a signature of a particular class of behavior, and the output is a class label associated with this behavior. The network is trained to identify an arbitrary pattern as a member of a state among a set of possible states.

Clustering, on the other hand, involves the grouping of patterns according to their internal similarity and requires no labels. The aim of clustering is to distribute the set of patterns into classes such that the patterns in each class have similar statistical and geometrical properties. Once the classes have been defined, characteristic features can be identified and used in the classification of new (unknown) patterns.

It is clear that the classification and clustering capabilities and the input-output modeling based on real-world data makes ANN a very powerful tool in the processing of data into information that is needed in the solution of complex problems.

3.2. Fuzzy Logic and expert systems

Fuzzy logic offers a convenient mathematical language for describing a system whose behavior can be articulated in “IF-THEN” rules. Since its inception in mid 60s, fuzzy logic has been successfully applied to a variety of challenging problems. In recent years there is an increasing number of applications in process control. Besides the process control, fuzzy logic is applied to diagnostics, many expert system applications and simulation and generally offers tools well suited for the description of non-linear and time-varying systems. The main difference between fuzzy logic approaches and conventional techniques being that the former use qualitative information whereas the latter require rigid analysis relations to describe the system.

A first generation expert system is a shallow expert system that consists of a knowledge-processing unit and a heuristic knowledge base. An expert shell contains no a priori knowledge. It has to be filled with domain knowledge prior to its use. Domain knowledge is captured in production rules. The production rule paradigm is a model for human reasoning. It captures an expert’s experience and casual reasoning strategy. It is a representation paradigm where knowledge can be captured in the form of rules. The rules consist of compiled associations of facts and phenomena with solutions and actions. The knowledge base containing these rules is a large set of precompiled chunks of deep knowledge ready to use rather than a collection of shallow knowledge in the form of if-then rules.

In the first-generation expert systems, two fundamental forms of reasoning process are involved. These are forward chaining and backward chaining. Both strategies work on production rules, but complementarily. Forward chaining works from antecedent to conclusion, while backward chaining works from conclusion to antecedent. There are several ways for the improvements to increase search speed, performance. However first generation expert systems contain shallow knowledge. Expert systems based on purely shallow knowledge cannot give satisfactory explanations about their behavior and show abrupt degradation at the edge of their knowledge domain, since no compiled knowledge about cases that never occurred before is available. Moreover, since knowledge elicitation depends on subjective human experts addressing only a limited number of cases, the expert system's knowledge domain is incomplete and possibly inconsistent of format and meaning.

In second generation expert systems, knowledge is derived from the first principles introducing generality. However this does not imply that the resulting model precisely describe the physical model because the first principles are not detailed enough for the complexity of the real world.

Integrating first and second generation expert systems make it possible to use heuristic knowledge to decide when to carry reasoning back and forth from heuristics to first principles. This is important when experience fails or is lacking or when the domain model is incomplete.

3.3. Genetic algorithms

The power of genetic algorithms derives from their emulation of nature's principle of evolution over generations of species. In searches for optima, a population of candidates is allowed to evolve over several generations, with the fittest individuals having the best chances of survival. The simple operations of reproduction, crossover and mutation of populations are the essence of genetic algorithms.

For many practical problems, the only way to find an optimal solution is to search through the entire set of all possible solutions. Unfortunately, such an exhaustive search of the "total parameter space" is not practical, because this parameter space is too large. The next best solution is to organize the search so that there is a high likelihood of reaching a near-optimal solution. Genetic search algorithms, inspired by biological evolution, crossbreed trial solutions by crossover and allow only the best solutions to survive after several generations. In a genetic algorithm approach to optimization, a problem is formulated in such a way that any solution can be encoded in a string of binary digits. After evolving over a sufficient number of generations, the encoding process is reversed, and

the optimal or near-optimal solution is identified from the binary string that survives. Each string of binary bits can be assigned a fitness value based on how well the corresponding problem solution meets some stated goal. A population of strings is subject to reproduction and mating to produce a new population of the same size. In the reproduction stage, the probability of each individual string being reproduced is proportional to the string's fitness. The best strings are reproduced many times, and the strings with low fitness values die out. Mating is achieved by taking two strings and combining the left portion of one string and the right portion of the second string to form a third string of the same length. A fourth string is generated with the right portion of the first string and the left portion of the second string. The "crossover" point where two original strings are separated is selected randomly. To avoid local optima, mutations are introduced during the mating stage. Each binary digit has a small probability of being reversed during the genetic recombination. Generally, populations of 100 individual strings taken through 20 generations have a high probability of including optimal or near-optimal individual strings. Furthermore, a mutation probability of about 0.001 per bit is adequate to prevent a search from locking onto a local optimum. Such an optimization procedure is extremely useful in complex problem solving where it is simply not possible to identify the nature of the interrelationships or to model the process involved.

3.4. Anticipatory systems

Anticipatory systems are systems that change state by incorporating in their control strategies information pertaining to anticipated, future states. Thus, in anticipatory control an action is taken on the basis of present conditions as well as a reliable estimate of what the system's performance may be in the near future. It appears that there may be a variety of anticipatory control strategies applicable to a given system. They all fundamentally share the need for an internal predictive model of the system and/or its environment used in a time-scale faster than real-time to influence decisions taken at the present. Thus, the fundamental building blocks of an anticipatory system are **S**, the actual system; **M**, a predictive model of the system and/or its environment; **C**, the controller and **E**, the environment.

In conventional control, processes are essentially modeled as reactive, i.e., the control action is taken in response to the past and present behavior of the system. In anticipatory control, the model **M** is used to anticipate the impact of current control decisions on the future performance of the system, i.e., Δt time-step beyond the present. Thus the predicted states

enter into the computation of the control law. Rule-based systems make anticipatory control decisions typically based on performance measures both current and anticipated, with anticipated performance being incorporated in their overall control strategies. Performance, therefore, becomes a quantitative tool for summarizing context specific information about the system's state, so that an intelligent control/observer can make effective decisions within the time frame smaller than Δt .

The anticipatory paradigm may be particularly well suited for the intelligent problem solving of complex problems. It allows us to view complex systems as systems possessing a variety of descriptions or models. The appropriate one at any given time is chosen on the basis of computational parsimony and efficacy. It attempts to overcome one of the most serious problems in large complex systems, that it is the inherent lag due to both large size and complexity.

3.5. Chaotic systems

Chaos occurs in two varieties: random chaos and deterministic chaos. Random chaos has no underlying basis since it arises from a series of independent events or decisions that have no relation to the immediately preceding events or decisions. The only way to relate the state of a random chaotic system over an increment of time is through the use of probabilities. On the other hand, deterministic chaos has an underlying functional relationship between the different quantities involved. Although they are often handled in a statistical manner, this underlying functional relationship allows alternative approaches. Generally, the underlying functional relationship is non-linear. As the range of the fluctuations grow, the behavior of the deterministic chaotic system becomes erratic and less predictable. For small perturbations, such a deterministic chaotic system may be linearized, and behavior of the system, under most conditions, is substantially the same as linear systems. The whole area of analysis using "random noise" techniques is an example of the linearized behavior of such systems when the perturbations are small. There are two types of deterministic (non-linear) chaos: Hamiltonian chaos where there is no friction and the dissipative type chaos, which are the systems of interest in most systems, in which the system loses energy. In dissipative type chaotic systems, the predictability of the behavior decreases exponentially with the time into the future. The initial conditions allow many different variables and many degrees of freedom, but dissipative systems damp out the degrees of freedom, and only a few important components of the behavior remain important. That is, the "motion" of the system, which initially occurs in a

very high-dimensional space, becomes confined after some time to a subspace of low dimensionality called an “attractor”. The behavior is usually determined as much by the initial conditions as it is by the underlying functional relationship, and a special pattern emerges, usually called “limit cycle” behavior. This results in a simplification that assists in the understanding and controlling of the chaotic system.

3.6. Cellular automata

Cellular automata is a form of dynamic simulation that can be used for many physical, engineering, biological phenomena. In a sense, it is analogous to genetic algorithms, where phenomena are enclosed, a biologically based process is imposed and the resultant is decoded to give an ultimate result. Cellular automata consist of “cells” that are in the form of rectangles in 3D space. Cellular automata start with a specific pattern represented by the cells at the corners and generate a future situation by each cell performing changing in accordance with a pre-specified law or pattern. Changing these laws or patterns will give a different result. Starting with a different set of initial conditions will give different final results. Here, the cellular automation is assumed to be

- **parallel**, that is the individual cell updates are performed independently of each other at the same instant
- **local**, that is when a cell is updated, its new value is based solely on the old value of the cell and its nearest neighbors
- **homogeneous**, that is each cell is updated according to the same rules. Typically the cell values and its nearest neighbors are combined according to some rule or look-up table.

In reality interactions are not homogeneous. The introduction of non-homogeneity in the behavior of different cells is like introducing non-linearity into a physical phenomenon. There is an analogy with the situation in chaotic behavior. Results are not very predictable beyond a short time period and the final result often depends on the initial conditions. With updating schemes that have a dissipative type behavior, the pattern could be expected to approach equilibrium similar to that involved in a “limit cycle” in chaotic phenomena.

3.7. Wavelets and multiresolution

The wavelet transform is a fairly simple yet very powerful mathematical tool which divides data or functions into localized components, then studies each with a resolution matched to its scale. Given a function $f(x)$, wavelet transform provides coefficients, called ‘wavelet coefficients’,

which results from the inner products of the signal and a family of ‘wavelets’. With these coefficients and the associated wavelet functions the function $f(x)$ can be expressed in different level of approximations. For continuous wavelet transform, the wavelet, corresponding to a scale $\mathbf{a}$ and the time location $\mathbf{b}$ is

$$\psi_{\mathbf{a},\mathbf{b}}(\mathbf{x}) = \frac{1}{\sqrt{|\mathbf{a}|}} \psi\left(\frac{\mathbf{x}-\mathbf{b}}{\mathbf{a}}\right) \quad (1)$$

where $\psi_{\mathbf{a},\mathbf{b}}(\mathbf{x})$ is the wavelet which is an element of the space $\mathbb{L}^2(\mathbb{R})$.

Hence, the integral wavelet transform $(W_{\Psi} f)_{(\mathbf{a},\mathbf{b})}$ is defined as

$$(W_{\Psi} f)_{(\mathbf{a},\mathbf{b})} = |\mathbf{a}|^{-1/2} \int_{-\infty}^{+\infty} f(x) \psi[(x-\mathbf{b})/\mathbf{a}] dx \quad (2)$$

where $\mathbf{a}, \mathbf{b} \in \mathbb{R}$ and $f \in \mathbb{L}^2$. Since by definition, ψ is localizable in both time and frequency scale, the integral wavelet transform is also localized and gives information in both domains within the bounds of the uncertainty principle. $\psi_{\mathbf{a},\mathbf{b}}$ are the basis functions and can be viewed as contracted and shifted versions of the prototype basis function $\psi(x)$. The parameter $\mathbf{a}$ is a scale parameter, and the argument $\mathbf{b}$ is location variable. If the scale parameter is large, then the basis function becomes a stretched version of the prototype, corresponding to a low frequency function. Similarly, small values of the scale parameter $\mathbf{a}$, correspond to high frequencies. Since the windowing is contained in the basis functions, they handle the time-frequency window. The wavelet transform $(W_{\Psi} f)_{(\mathbf{a},\mathbf{b})}$ decomposes the function $f \in \mathbb{L}^2(\mathbb{R})$ in the base of elementary wavelets. The wavelet is often chosen to be well localized in space and frequency. In the Fourier domain

$$(W_{\Psi} f)_{(\mathbf{a},\mathbf{b})} = |\mathbf{a}|^{-1/2} \int_{-\infty}^{\infty} d\omega f(\omega) \cdot \psi(\mathbf{a}\omega) e^{2\pi i \mathbf{b} \omega} \quad (3)$$

where $f(\omega) = \int_{-\infty}^{\infty} e^{2\pi i \mathbf{b} \omega} f(x) dx$ is the Fourier transform of $f(x)$ and $\psi(\omega)$ is the Fourier transform of $\psi(x)$.

The natural requirement for the wavelet transform would be the reconstruction of the function $f(x)$ from its wavelet transform $W_{\Psi} f$. The reconstruction of the function $f(x)$ from its wavelet transform counterpart is possible that it is given by

$$f(x) = (1/C_{\psi}) \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (W_{\Psi} f)_{(\mathbf{a},\mathbf{b})} \mathbf{a}^{-1/2} \psi\left(\frac{\mathbf{x}-\mathbf{b}}{\mathbf{a}}\right) d\mathbf{a} d\mathbf{b} / \mathbf{a}^2 \quad (4)$$

so that, $f(x)$ can be described as a summation of the basis functions $\psi_{a,b}(x)$ with the weights of wavelet coefficients $(W_\psi f)_{(a,b)}$ where C_ψ is given by [2]:

$$C_\psi = \int_{-\infty}^{\infty} \frac{|\Psi(\omega)|^2}{\omega} d\omega < \infty \quad (5)$$

Above, $\Psi(\omega)$ is the Fourier transform of $\psi(x)$.

4. Method integration applications

In this section some application areas of method integration are briefly presented to show the efficiency and effectiveness provided by the *method integration*. The problem areas are from engineering disciplines where the most implementations take place.

4.1. Data/sensor fusion

Data/sensor sensor fusion is the synergistic combination of information made available by various information sources such as sensors, in order to provide a better understanding of a given scene. Such information fusion is important in many engineering systems as robotics, power plant. Kalman filtering technique is one approach for the estimation of the state vector pertaining to the observed system. However, such approach requires that all information used should have an accurate probabilistic interpretation. The situation is often further exacerbated since the solution can be hard to compute and difficult to implement practically. For this case a trained neural network in the form of multi-input multi-output can serve as a reliable solution. A power plant application is given in [3].

4.2. Neural networks with fuzzy logic

Although fuzzy logic algorithms have been demonstrated to be reliable and superior in performance to conventional systems, a number of issues primarily having to do with the completeness of the rule set and the generation of membership functions remain open. There is encouraging evidence, however, that the latter problem, i.e., generating the difficulty of generating membership functions may be addressed through the employment of artificial neural networks [4]. Fuzzy logic has an enormous tolerance for imprecision and therefore may offer the mathematical language of choice for the description of the systems that

are large, non-linear and by the nature of their enhanced organizational complexity difficult to manage. Various neuro-fuzzy treatments can be found in [5,6,7], for instance, for the effective solutions of complex engineering problems. An application on robotics is given in [8]. Seemingly, in a particular application fuzzy logic and neural network approaches are interchangeable. However, from the actual application viewpoint fuzzy logic and neural network implementation areas are represented in perspective in Fig.1.

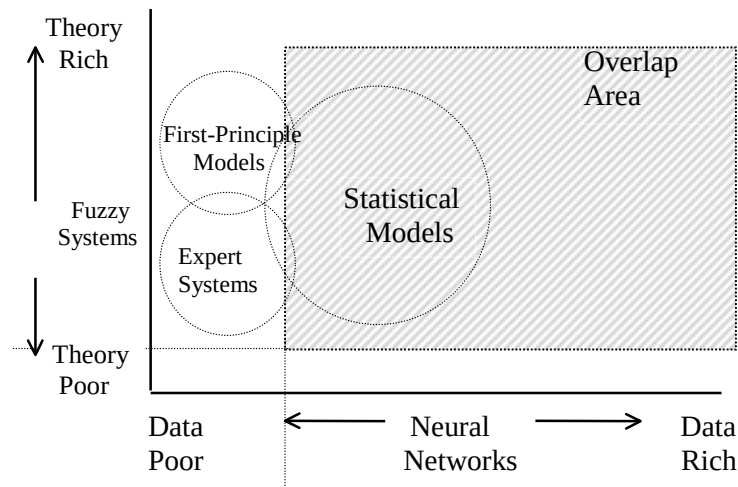


Figure 1. Fuzzy versus neural networks in supervisory monitoring [9]

4.3. Neural networks applied to signal processing

The use of neural networks in signal processing is becoming increasingly widespread, with application virtually in all areas such as filtering, parameter estimation, signal detection, pattern recognition, signal reconstruction, system identification, signal compression and so on. Various treatments can be found in [10], for instance.

4.4. Wavelet functions for finite element analysis

In finite element method application with variational methods, an obvious choice to use for curves and surfaces are the B-splines, since they are already commonly used as a basis in modeling systems. However, in the solution procedure each B-spline basis function represents only a small portion of the solution so that any broad changes in the curve propagate incrementally from one basis function to the next. The implication of this

is that the iterative solution techniques commonly used for large sparse systems are slow to converge to the correct solution. The solution to the slow convergence is to use wavelets constructed from B-splines for the finite elements, rather than the B-splines themselves.

4.5. Real-time systems

In a complex (-integrated) real-time system, at the most basic operating level sequence of actions based on measurements describing the changing conditions in the system environment have to be planned. Because of the large volume of data at a particular time, simple means or human assessments due to timing constraints cannot perform the duly evaluation and optimal decisions. The data from the information sources (e.g., sensors) therefore must be synthesized into global knowledge-structures pertaining to the behavior of various plant systems. This requires the validation of (sensor) data, its preprocessing and fusion into global parameters. The synergetic use of robotics, neural networks, fuzzy expert systems, and related AI technologies to facilitate the computation as well as the display of such information is essential. This may involve model-based validation of signals at a lower-level of plant description or the improved cognitive ergonomics of human-computer interaction that would enable the operator to comprehend the system's state and plan on desirable actions.

On a larger scale, AI may be useful in scheduling the entire work cycle of a system that may include operators, robots, as well as other types of automated machinery. An example of a highly automated system is the so-called "dark factory", where operations such as welding may be done by robots. Transport of part-completed assemblies is automatically scheduled as a set of actions for robotic transporters and cranes. The human operator has a supervisory role at the global plant level aided by monitoring, diagnostic and explanatory computer facilities.

Intuitively, system reliability apparently is a function of the complexity if the complexity is related to the number of components in that system. In such a case complexity is intrinsic property of the system. However, complexity in complex real-time systems can be defined differently. Here, complexity in complex real-time systems is considered to be not an intrinsic property of a system or a system description but rather arises from the number of ways in which an observer/controller is able to interact with the system. In this sense complexity is a function of the interactive relationship between the observer and the system. An important feature of large dynamic complex systems is that their operating environment often changes without any external intervention. Hence it is

crucial that the supervisor has a good model of the system and its environment. AI tools can facilitate and validate the procurement of information about the system so that an updated global model of the process at any given time is available. Considering such dynamic complex engineering systems, they involve three distinct yet related functions: monitoring, diagnostic and control that are briefly described below.

4.5.1. Monitoring

The primary goal of the monitoring function is to maintain a state of readiness or preparation for changes in the status of the system. It requires that supervisor continuously samples displayed data, evaluate it and detect significant system changes. This is a challenging task. The conventional monitoring technology emphasizes data collection, integration, storage and display aspects. Unless this data is synthesized or fused into a contextual structure it is of little and sporadic value. Systems must be developed for complex systems to enhance monitoring technology by taking into account ergonomic and cognitive factors. In this respect, auto-associative neural networks can play excellent role for monitoring as they can be effectively trained for learning the system dynamics in a multi-input multi-output structure [11,12].

4.5.2. Diagnostics

Diagnosis in a general sense is defined as the process of determining the state of a system based on the interpretation of potentially noisy or uncertain data. In most applications this is followed by a recommendation as to what can or should be done, if anything.

Almost irrespective of the diagnostic domain all intelligent diagnostic systems must address the problem associated with the interpretation and abstraction of data. Namely, data may be noisy and uncertain and the fact that measurements that could directly confirm a diagnostic hypothesis are quite expensive and at times may endanger the system. Diagnostic expert systems developed throughout 2 decades relied heavily on heuristic (or "shallow") reasoning i.e., heuristic relationship between findings of the system to be diagnosed and possible diagnoses. They are often called "first generation expert systems". They are presently replaced with "second generation expert systems" which employ deeper forms of reasoning such as causal reasoning or sophisticated models.

One of the major limitations of expert systems in performing on-line diagnostic functions is the fact that they are relatively slow. Large complex systems are highly dynamic and have underlying time constants

associated with their behavior. Therefore it is desirable to develop time-critical approaches to AI approaches. Time critical AI allows one to model the time constants, build a time constraint propagation model, and devise knowledge intensive search strategies. This may call for the development of multiple bifurcating search strategies, one or more geared for arriving at a feasible decision and one or more geared for subsequently arriving at an optimal decision if time permits.

Well-trained neural networks can be used as a means of diagnosing transients or abnormal conditions. Although it is time-consuming to train the network in the first place, a library of trained networks accessible by the system may provide a useful way of addressing the time-critical constraints of diagnosing a complex system for the purpose of managing its total behavior. This includes maintaining it within the desirable region of its state-space.

4.5.3. Control

Applications of intelligent systems to real-time control have been relatively constrained by the tardiness of symbolic computations. There has been a rather successful introduction of AI in the area of auto-tuning adaptive controllers, with commercial products. Several expert systems have been proposed for control system design. It is widely recognized that expert systems for real-time control will be model-intensive, dynamic and time stressed. Important precursors to any control function are the monitoring and diagnostic functions and the execution of all three for the successful control of complex real-time system. The two main bottlenecks associated with the real-time control of large complex systems are as follows.

-Bottleneck at the level of software implementation: Real-time control systems typically use procedure-oriented algorithms. It is not easy to develop closely coupled numeric and symbolic tasks in an integrated environment. Existing knowledge representation structures and inference strategies are not suitable for representing dynamic model reasoning processes in real-time. The use of neural network trained to emulate real world systems from experimental data, offer an alternative that could overcome these problems.

-Bottleneck at the level of hardware implementation: At the hardware level, under a time-stressed environment, a distributed structure is needed for implementation of real-time control, allowing real-time reasoning in parallel.

In the last decade, neural networks and other connectionist approaches have gained enormous gravity in control applications. The trained

network offers considerable promise as fast and robust computational models that implicitly take into account the non-linear behavior of a system and can be used in a predictive manner in model-based control. They can be used as sophisticated filters that can distinguish between control errors that can be adjusted and control errors that should be ignored.

Conclusions

A concept of engineering method called *method integration* is presented. The method is essential especially for the integration of the emerging information processing methods into the conventional methods. By the integration powerful and robust solutions for complex problems can be obtained and thereby complex problems of the modern technologies can effectively and efficiently be treated. Such solutions may include “intelligence” by means of which complex man/machine integrated system problems can also be thus handled. *Method integration* can take various forms in actual implementations as the problems subject to method integration can vary. Since the emerging methods have established their modern technologies, the familiarity with these is required for finding effective integration and thereby solution procedures. In this respect the following two areas are of strategic importance:

A focused theme: The integration of the rapidly advancing distinct, yet disjoint, theories and technologies into coherent framework of paradigms and tools for the intelligent solving the problems of complex systems.

Innovative research: Innovative research related to robotics, fuzzy-expert systems, neural networks, anticipatory systems, genetic algorithms, fuzzy systems theory, multiresolution and so on, have to be undertaken.

In both cases, the goal is to make such complex systems simpler and safer to operate more efficient, flexible and producing on demand.

References

- [1] Veer, P. van der & Beheshti, R. M., The choice of decision support methods for Civil Engineering Design coping with integrated uncertain and incomplete data, *Journal of design Sciences and Technology*, Vol.5 no.1, pp.35-41, 1996
- [2] Chui, C.K., *An Introduction to Wavelets*, Academic Press , Boston , 1992

- [3] Bonfert, J.P. et al., Instrumentation and signal validation under extreme conditions, *Reinforced concerted action on reactor safety FISA-EU research on severe accidents*, eds. G. van Goethem, W. Balz & E. Della Loggia, EUR 16896EN, European Commission ECSC-EC-EAEC, Brussel•Luxembourg, (ISBN 92-827-6980-1) 1996
- [4] Wang, Li-Xin, *Adaptive Fuzzy Systems and Control*, PTR Prentice Hall, Englewood Cliffs, New Jersey 07632, 1994
- [5] Lin, C.T. and Lee, G., *Neural Fuzzy Systems*, Prentice Hall PTR, Upper Saddle River, NJ 07458, 1996
- [6] Jang, J.S.R., Sun, C.T. and Mizutani, E., *Neuro-Fuzzy and Soft Computing*, Prentice Hall Upper Saddle River, NJ 07458, 1997
- [7] Nauck D., Klawonn F. and Kruse R., *Neuronale Netze und Fuzzy-Systeme*, Vieweg Computational Intelligence, ISBN 3-528-15265-6, 1994
- [8] Holweg, E.G.M., *Autonomous Control in Dexterous Gripping*, PhD Dissertation, Delft University of Technology, 1996
- [9] Ciftcioglu, Ö. and E. Türkcan, Supervisory monitoring system in nuclear power plants, *IAEA Technical committee meeting on nuclear power plant diagnostics, safety aspects and licensing*, Portoroz, Slovenia, 23-25 June 1997, IAEA-J4-TC1031, Vienna , Austria.
- [10] Luo, F.I. and Unbehauen, R., *Applied Neural networks for Signal Processing*, Cambridge University Press, England, 1997
- [11] K. Nabeshima, E.Türkcan and Ö. Ciftcioglu, Real-time nuclear power plant monitoring with adaptively trained neural network, *ICANN'93: International Conference on Artificial Neural Networks*, 13-16 September 1993, Amsterdam, eds. S. Gielen and B. Kappen, Springer Verlag, Berlin, p.863, 1993
- [12] Ö. Ciftcioglu and E. Türkcan, *On-line plant-wide monitoring using neural networks*, *8th Power Plant Dynamics, Control and Testing Symposium*, May 27-29, 1992, University of Tennessee, Knoxville, Tennessee, USA