Cesare Alippi, Alessandro Ferrero, and Vincenzo Piuri

Artificial intelligence for instruments & measurement applications

Artificial intelligence technologies are now attractive tools to enhance and improve the efficiency, the capability, and the features of instrumentation in application areas related to measurement, system identification, and control. These techniques exploit the computational capabilities of modern computing systems (in particular, of microprocessor and DSP-based systems) to manipulate the sampled input signals and extract the desired measurements. The aim of the paper is to introduce the basic concepts of artificial intelligence techniques and present a survey of those applications related to instrumentation and measurement for which such paradigms have proven to be effective. In particular, we focus on artificial neural networks, fuzzy logic, and expert systems. The common idea shared by artificial intelligence technologies is to process and analyze available data and *a priori* information intelligently in order to attain low, medium, and high abstraction levels in the measurement process, enhance the quality and the consistency of measurements, and create new and advanced instruments.

Introduction

The introduction of new techniques in measurement science has usually been directed toward enhancing the quality of measurement procedures and instrument features, in order to create advanced solutions for known applications and to provide suitable tools for new ones. Research and industrial activities in the instrumentation and measurement fields have traditionally been concerned with the design of, and experimentation with, new sensors, new materials, new technologies, and new methods to analyze data, process signals and images, and identify and control systems.

1053-5888/98/$10.00©1998IEEE

Recent research and industrial applications have been finalized to study and exploit the use of intelligent and complex manipulations of input signals within the instruments themselves. The main goal is to improve and expand the capabilities of traditional measurement systems and introduce new instruments for high-level measurements (and, consequently, for new applications). Artificial intelligence techniques, such as artificial neural networks, fuzzy logic, and expert systems, are relevant examples of the evolution of data processing in this direction.

In traditional instruments, including those based on complex techniques for digital signal processing, operations performed on input signals usually follow strictly deterministic algorithms. Conversely, artificial-intelligence techniques abandon such approaches, thus allowing one to deal with high-abstraction-level measurement procedures, such as complex nonalgorithmic processing, self-validation of measurement results, and automatic selection of the most suitable measurement procedure for the given environmental conditions.

Information-processing technologies execute an algorithm to identify the desired solution for the specific application case. Unfortunately, in some cases, the complexity of traditional algorithms is too high, or it is too difficult to develop an algorithm itself, or the algorithm is unknown or cannot be completely specified. This is the case with several applications related to approximated reasoning and associative problems, as many times happens in system identification, control, prediction, robotics, signal processing, filtering, image processing, vision, and pattern classification. In several cases, adaptive technologies (e.g., neural networks and fuzzy logic) have been shown to be capable of finding reasonable solutions for such problems by applying strategies resembling those typical of human reasoning, such as similarity, analogy, generalization (e.g., data interpolation and extrapolation), and multiple-goal optimization.

To this end, artificial neural networks realize an intelligent data treatment, which is often able to capture the behavior of a system without necessarily requiring *a priori* knowledge. The desired solution of a problem is learned through examples instead of being defined by means of algorithmic statements. Fuzzy logic is a nondeterministic data treatment which deduces the system behavior by estimating the probability density function associated with each possible variable value and by applying suitable combination rules. In contrast to neural networks, some knowledge about the physical system is needed to identify the relevant variables and generate the combination rules.

The technical literature contains a large number of commercial products and application areas solved by neural and fuzzy technologies. It should be noted that even if some guidelines have been proposed to optimize these paradigms to specific applications, well-assessed general methodologies are not yet available due to the complexity of the paradigms, the number of possible variants, and parameters that must be configured.

Intelligent techniques for data processing have also been studied as auxiliary tools to optimize and analyze the behavior of instruments and systems by exploiting the knowledge and the sensitivity of human experts. This knowledge can be concentrated and realized in the expert systems. Input observation and deduction of the consequences on system behavior are performed according to empirical and statistical rules extracted from such knowledge. The idea is to capture the behavior and the skills of experts so that they can be autonomously reproduced and used to infer actions in new cases. Typical and well-assessed applications are mainly related to analysis of faults and errors, diagnosis of instruments and systems, and configuration and tuning of instrumentation.

This paper presents a simple introduction to the most relevant and widely adopted artificial intelligence technologies with a specific reference to their use in the instrumentation and measurement fields. This provides reference points to the designer in order to properly take into account artificial intelligence technologies when designing and evaluating instruments and systems. More detailed and comprehensive presentations can be found in the general references at the end of the paper as well as in the reference listed on the web site cited in [8]. Due to the variety of aspects and alternatives, our attention is focused only on the approaches that have been shown to be attractive and effective in instrumentation and measurement or strictly related areas. Characteristics of neural and fuzzy adaptive computing paradigms are given in Section II, together with configuration procedures and an overview of the most relevant applications. Section III discusses the use of expert systems for diagnosis of instruments and systems, which is one of the most effective and reliable results of this technology. This provides reference points to the designer in order to properly take into account the most suitable artificial intelligence technologies in designing and evaluating instruments and systems.

Adaptive Technologies

Adaptive computational paradigms are techniques suitable to describe nonalgorithmic input manipulations such as those which reproduce the nondeterminism, the fuzziness, the adaptability, the generalization ability, and the parallelism typical of human reasoning. The most relevant techniques among these are artificial neural networks and fuzzy logic.

Artificial Neural Networks

Neural network paradigms were initially proposed as massively parallel computational models capable of capturing and reproducing the behavior and the activities of the human brain. The brain has, in fact, extraordinary properties and abilities in solving very complex tasks by partitioning and efficiently searching the knowledge space of the solution. This activity can be carried out in parallel as in a distributed information-processing system. On the other hand, the brain is able to infer the solution of a given problem by generalizing the interrelationships among known or partially specified information. Such deductive abilities are based on a "learn from examples" philosophy, carried out by means of analytical inspection, understanding, interpolation, and feature extraction. All of these capabilities find their effectiveness in a wide range of applications involving, for example, pattern matching, recognition, image and signal processing, and speech understanding. In addition, they are needed for the identification of the behavior of

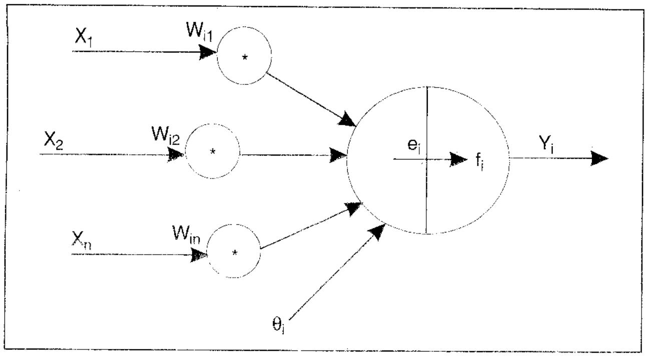

Figure 1 *The general model for a neuron*

dynamic nonlinear systems and related applications such as control, robotics, vision, and forecasting.

An artificial neural network is a computational paradigm whose operation and structure are defined by a directed graph (see Fig. 1). The nodes are the processing units (*neurons*), and the arcs (*synapses*) represent interactions among neurons. For each neuron, an interconnection weight (*synaptic weight*) is associated with each incoming arc in order to model the influence of the other neurons' outputs onto the computation of the envisaged neuron. More formally, a generic *i*-th neuron receives, from a neuron *j* or from the external world (*network inputs*), the *input* X_j and computes an *excitation function* $e_i = e_i\,(W_{ij}, X_j)$ where W_{ij} is the synaptic weight associated with the i_j arc. Usually, the excitation of the neuron is given by the weighted summation of its inputs: if the neuron receives n inputs, then $e_i = \sum_{j=1}^{n} W_{i,j} X_j - \theta_i$ where θ_i is a *threshold* term. A more complex excitation function can be used to model other behaviors. Inhibitory input signals are introduced to prevent neuron activation under given conditions established by other neurons, while pulsed inputs are cumulated to simulate the asynchronous operation of a natural neuron instead of evaluating synchronously the excitation on the current set of input values. Then, a nonlinear transfer function f_i (*activation function*) is applied to the excitation signal to produce the neuron's output. Several activation functions have been reported and tested in the literature, examples are available both for continuous functions (e.g., linear, linear with saturation, and sigmoid) and discretized functions (e.g., sign, step, and multistep). The neuron outputs delivered to the external world are the *network outputs*. Both the excitation and the activation functions may differ within a neural network.

The interconnection topology among neurons defines the order of propagation of the neuron outputs, i.e., the propagation order of the distributed parallel computation, so as to reproduce the learned behavior. In the literature, many approaches have been tested for different classes of applications. Two main families of structures have been proposed, static and dynamic networks.

Static networks are suitable for solving problems which can be logically modelled as static mappings (e.g., pattern classification, filtering, nonlinear regression). Under weak hypotheses, any reasonable function can be approximated by using multilayered feed-forward topologies (see Fig. 2a), in which the network is partitioned into disjoint subsets (or layers) characterized by adjacent connections without creating closed loops.

Extensions of this model incorporate lateral interconnections between neurons within a layer (e.g., to represent inhibitory connections); a special case is the self-organizing map (see Fig. 2b), which groups inputs according to an optimizing function. Other extensions incorporate backward links, which happen in the Hopfield's networks (see Fig. 2c); the final result is obtained when a steady state is reached.

Dynamic networks are able to capture the dynamical behavior of a system by introducing memory elements (in terms of time-lag elements). Such modifications allow network topologies to describe dynamic systems (e.g., for identification of dynamic systems, control, motion, and high-level vision applications).

Interesting results have been achieved by considering neural networks based on *external feedback loops* (see Fig. 2d). The network's inputs are current and past primary inputs and past predicted (or actual) outputs. The number of past inputs and outputs that need to be taken into account is related to the internal dynamics of the system and can often be deduced—at least in a first approximation—by studying a rough physical model of the real system and the related differential equations. This neural topology has been shown to be effective in several practical cases of prediction and identification.

Feedback loops can be introduced *locally* within each neuron (see Fig. 2e). These networks are quite difficult to be configured since they have many degrees of freedom and local states. This may lead to a long training procedure, and it requires many examples to suitably configure the network weights.

Alternative topologies may be considered by introducing memory elements at the layer level. From preliminary experiments, it seems that these more complex structures have limited influence on the practice.

A general network structure can be derived from the *state-output* system model, which is well known in systems theory (see Fig. 2f). In this case, the state variables are explicit, and the system model is partitioned into two blocks generating the states and the outputs, respectively. The first block uses past inputs and states to compute the current state with a dynamic neural network; the second block approximates output from the current state and inputs by adopting a multilayered feed-forward topology.

When a family of models (i.e., a network topology) has been selected, the specific neural model must be defined: the number of layers, the numbers of neurons per layer, and the interconnection weights. Sample general guidelines are available for choosing the number of layers and the number of neurons within each layer.

The next step is to configure the network weights by applying a learning procedure: The weights configuration is obtained by optimizing a discrepancy function (e.g., a mean-squared error function) defined over the available examples. Two main classes of learning may be identified: *supervised* and *unsupervised*. In the first case, the actual outputs of the network are compared with the expected ones produced by the supervisor (i.e., the available example), and their discrepancy constitutes the loss function to

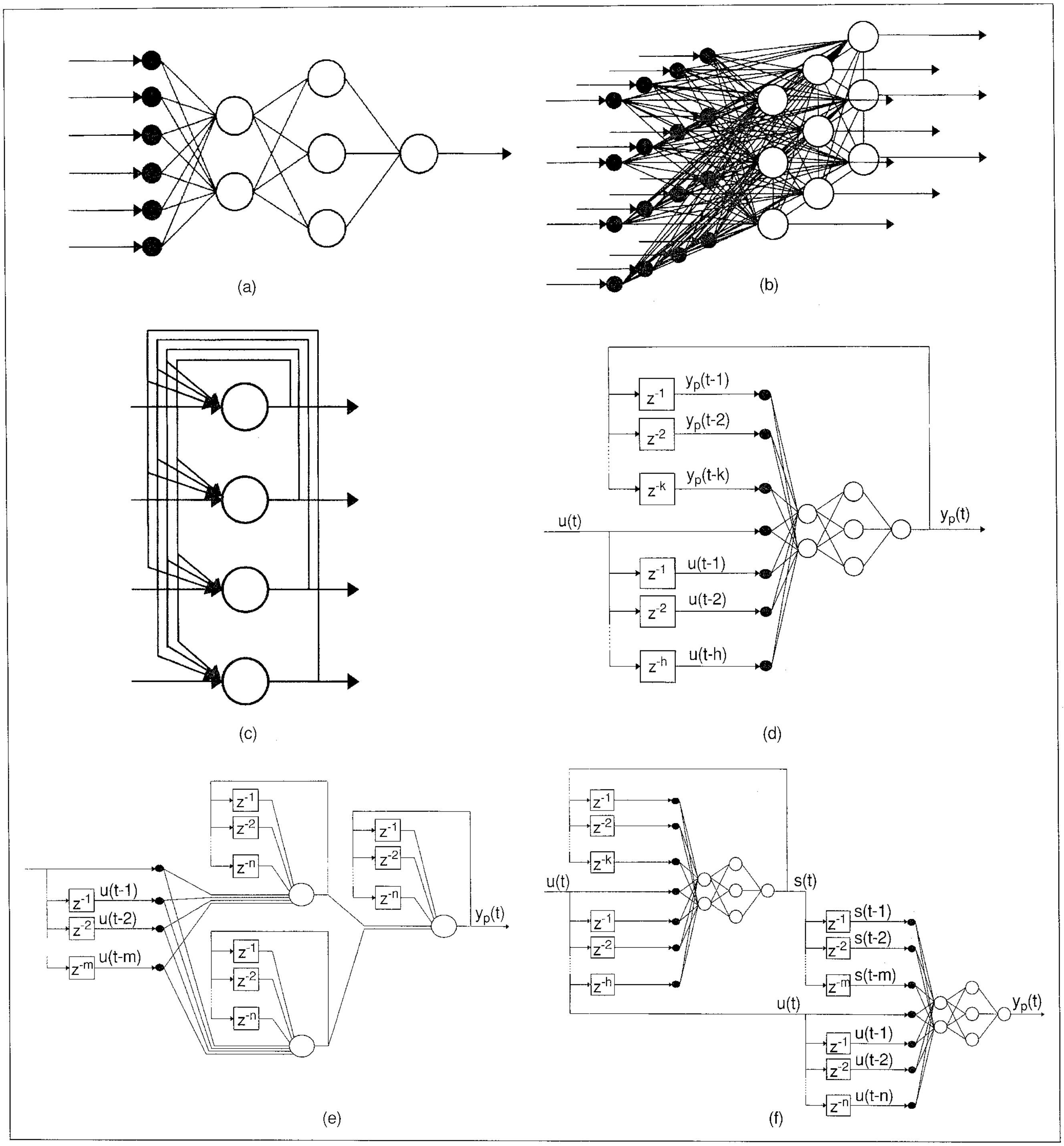

Figure 2 *The neural network topologies: multilayered feed-forward (a), self-organizing maps (b), Hopfield's network (c), external feedback loops (d), local feedback loops (e), state-output network (f).*

be optimized. In unsupervised learning, no supervisor is given, and weights are adjusted according to the optimization of a function depending on current inputs and weights; this is typical of self-organizing maps. During learning the error decreases as long as the network has unexploited degrees of freedom. On the other hand, learning in the long run has a main side effect if the network is overdimensioned (in terms of the number of degrees of freedom) with respect to the application. The learning procedure may extract from examples the possible noise which affects the measurement, and the obtained network features poor generalization ability (i.e., performs badly on new examples). Therefore, the learning procedure needs to be terminated as soon as a reasonable learning error has been achieved, before the network loses its generalization capability.

Fuzzy Logic

Fuzzy logic has been created as an approach for defining the computation in a nondeterministic way by introducing the concept of incompletely determined membership of an object to a set. (This is in contrast to traditional "crisp" logic in which an object either belongs or does not belong to a set in a definite way). The final goal is the effective simulation of human reasoning, which is capable of making good or reasonable decisions even in the presence of uncertainty and lack of precision. When the problem complexity increases, exact formulations of the system model and the desired solution become difficult and impractical, or even impossible, due to the necessity of understanding and abstracting the relevant aspects related to identification of a solution. In this case, a qualitative (and not quantitative) description of the system becomes more effective in capturing and representing the system characteristics, which are difficult to deal with in a deterministic "crisp" logic. However, the use of a fuzzy approach requires knowledge of some physical features of the system. The success of fuzzy logic arises from the ability to "measure" and treat the intrinsic vagueness of a physical system by using a flexible representation.

The key idea which led to the fuzzy logic theory was to extend and generalize the basic concepts of "crisp" set theory by introducing uncertainty and, in particular, the definition of membership grade to a set. An object does not simply belong or not belong to a set: it belongs with a given confidence (*membership grade*), that is, it can be classified as belonging to the set with a given confidence. There are no definite boundaries between sets; one shades gradually into another. An object is not fuzzy (i.e., incompletely determined) by itself: what is fuzzy is the subjective evaluation related to the membership of the object with respect to the envisioned set. Fuzzy sets and fuzzy logic are therefore generalizations of traditional "crisp" set theory and logic, respectively. Fuzzy representation and, specifically, fuzzy measurements, are radically different from probabilistic representation and measurements, respectively. In the second case, the characteristics of an object are not certain: their measurements are uncertain and imprecise. In fuzzy logic, the measurements of the object characteristics, though accurate and precise (at least within the precision of the instruments used for their measurement), cannot give any certain information about the system status since their interpretation in terms of membership to given sets is shaded and incompletely determined. As a consequence, overall system behavior cannot be derived by a deterministic processing of the available measured quantities but needs to be represented in terms of fuzzy membership to given sets.

Formally, a fuzzy set A, defined on a domain U, is described by means of a membership function which maps the domain U into the membership grade (which ranges in the interval [0,1]); the extreme values either indicate nonmembership or complete membership, as in the traditional "crisp" set theory. An example is shown in Fig. 3. In the case of data sequences, the membership function can be defined either on the individual datum or on a subsequence having a given length around the datum. The fuzzy rules allow expressing the processing strategy compactly in the form of approximate reasoning. A fuzzy rule usually involves a group of antecedent clauses, which define conditions and subsequent clauses defining the corresponding action. A fuzzy system is a nonlinear mapping of input fuzzy variables to an output fuzzy variable, defined by a set of fuzzy rules. To solve a problem by means of fuzzy reasoning, all the knowledge about the problem must be incorporated and expressed as a set of fuzzy rules and membership functions. As a consequence, the envisioned application needs to be known in advance and in detail in order to describe and model the system in a fuzzy form, by introducing some uncertainty in the formal description. In some cases, the fuzzy rules can be learned and extracted from the data during a training phase.

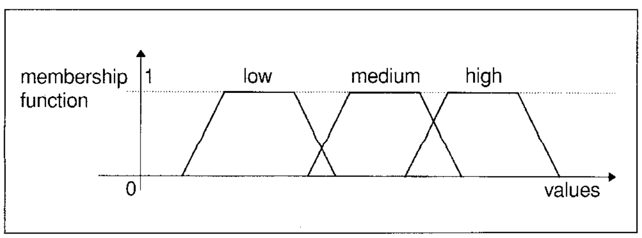

Figure 3 *Fuzzy sets: low, medium and high.*

The inference mechanism is the process that numerically evaluates the information embedded in the fuzzy set of rules (rule base) in order to generate the final result. More than one rule may become active at each step of the fuzzy computation, according to the fuzzy validity of the premises. Several approaches have been proposed in the literature for the inference mechanism, but they can be reduced to three basic strategies. The min-max approach evaluates the membership grade of the output variable in each active rule as the minimum value among the grades of the input fuzzy variables; then the final grade of the output variable is obtained by or-ing the fuzzy distributions of all active rules. In the product-sum method, the membership grade of the output variable for each active rule is scaled down by the minimum value of the grades of the input variables; these grades are then summed up. In the min-sum approach, the first step of each rule is performed as in the min-max method, while the intermediate results are then added as in the product-sum strategy to point out the nonlinearities. A scalar value needs to be generated as a final result of the fuzzy computation: "defuzzyfication" methods are available to extract a scalar value from the final membership grade distribution generated by the inference mechanism. For example, the scalar value corresponding to the center of gravity of the membership grade distribution is assumed as the expected result of the fuzzy computation.

Applications: Sensors, Measurement, and Modeling

The use of adaptive technologies in the area of instrumentation and measurement has several applications with regard to data analysis and manipulation at different abstraction levels, from sensor implementation to sensor enhancement, from sensor fusion to high-level sensors, from system identification to prediction, from calibration to system control, and from complex measurement procedures to intelligent instrumentation. This wide spectrum of feasible and efficient application is due to the capabilities of capturing the behavior of complex nonlinear dynamic systems by means of a limited number of parameters that can be configured adaptively and progressively. In addition, the relative compactness of these techniques in many practical cases, that is, their limited computational complexity with respect to other traditional approaches, is often suitable to support real-time applications and allow realization of analog/digital integrated circuits.

Neural networks and fuzzy logic can be exploited to implement new sensors at a physical level. Detection of physical quantities has been shown to be feasible in the literature. For example, artificial retinas and cochleas have been realized by using integrated circuits, as well as infrared and high-energy detectors based on CCD technologies. Nonintrusive and noninteracting instruments can also be realized by observing physical quantities strictly related to the desired one. For example, magnetic resonance, echography, or reflected light can be used to measure the state of an object (e.g., the roughness of a surface or the position in space).

The characteristics of traditional nonlinear sensors can be enhanced by transforming and smoothing the output generated in response to physical input stimulus. In particular, these technologies are effective in modifying the strong nonlinearities of some sensors so that the output becomes linear (or quasi-linear) to simplify subsequent use. Transformation and, in particular, linearization could obviously be implemented either by means of a look-up table or a specific algorithm. However, in many cases, an accurate table may be too large to be realized, while the algorithm may be time consuming or may need a dedicated computing system.

Noise reduction in the sensor measurements and, in general, signal or image filtering are other features which allow movement of the measurement and application problems related to the noisy data into the sensor itself, without imposing any specific requirement on subsequent operations using the sensor data. Filtering allows an outline of some specific features of the signal or image under observation; in several cases, the adaptive techniques are attractive with respect to traditional DSP-based algorithms since they can be expressed and implemented more easily, even by learning on the actual data.

A particular form of this ability deals with missed data: a loss of input data occurs in real systems when sensors are occasionally or temporarily unable to generate desired measurements from observed physical quantities (e.g., for transient or intermittent faults). In such a case, adaptive techniques are able to interpolate available data to predict the expected measurement. Alternately, when physically related data are available from other sensors, the techniques can deduce the most probable measurement of the missed sensor from the correlated analysis of data coming from the other sensors.

At a higher level, adaptive technologies can be exploited to extract information from the inputs coming from sensors and to create an instrument capable of performing a high-level measurement (possibly an indirect measurement). As for any traditional physical quantity we can, in fact, define a feature in the system under observation and a measurement procedure to evaluate quantitatively such a feature. The simplest example of these high-level approaches consists of measuring a quantity defined as a complex function of physical quantities observed by sensors; the desired quantity is measured by suitably merging the information extracted from the sensors (sensor fusion). Other more abstract examples are the membership of the inputs (e.g., sensor data, signal, and images) to predefined classes and the identification of features (e.g., pulses, spikes, specific waveforms, regularities, segments, and textures) in signals and images (e.g., for classification, pattern recognition, speech recognition, and quality measurements).

A high-level measurement is also the diagnosis of complex systems, which involves analysis and measurement of the operating status from observation of characteristic parameters. This application area may involve both the detection of the fault presence as well as the identification of the fault and/or the faulty component. Examples are available in the literature for sensors, motors, complex systems, and plants.

System identification has a relevant role in measurement and instrumentation since it allows capturing the system behavior (i.e., its characteristic parameters) from the analysis of measurements, even in the presence of noise. This is important whenever a product needs to be assigned and certified as belonging to a specific class, according to a given standard taxonomy. Neural modeling and fuzzy modeling have been shown to be effective and efficient in capturing the behavior of a wide range of systems, encompassing linear and nonlinear systems as well as static and dynamic ones. In particular, it was proved that they are universal approximators of any static system.

A direct application of dynamic system modeling is prediction of expected output after a given time interval. This is often useful in predicting the system behavior well enough in advance to make correct decisions about the system operation and control. Examples are in the control of complex systems and industrial plants, in robotics, and in the stock exchange. A particular case of prediction is "what if" simulation: Quite accurate simulations are performed in a very limited time to predict machine/plant reactions to different possible actions without actually applying them, so that the supervisor is able to select the most suitable one.

Calibration of sensors and systems through adaptive techniques is another interesting area. The optimal (or near optimal) set points of the characteristic parameters are identified and maintained so that the system behavior is adapted to the specific realization, the components actually used, and the surrounding environment.

Automatic control of complex (possibly nonlinear and dynamic) systems is a further field in which measurements and instruments are deeply involved to identify and maintain the working point within the desired accuracy. On the other hand, complex measurement systems often need embedded control systems in order to guarantee the target performance. Several examples are available in the literature covering these aspects. This is the widest application area which is now known to be effective for fuzzy logic techniques, even if neural approaches have been shown to be attractive, in particular, when the nonlinearities are very strong, the dynamics are complex, and the system behavior is not completely known.

Expert Systems

This technology tries to capture the knowledge and the reasoning abilities of people having an outstanding expertise in a specific field. The final goal is to reproduce, with a computer program, their intelligent reasoning, understanding, and skills in solving complex problems, that is, to apply the same rules these people explicitly or unconsciously use.

An expert system therefore has two main components: the knowledge base and the inference engine. The knowledge base is the collection of all information that constitutes the knowledge or expertise about the problem. Public knowledge includes published definitions, facts, and theories. Expertise usually involves more than public knowledge: human experts generally have a private knowledge that has never been published and that consists mainly of approximate rules deduced from practical experience (heuristics). Heuristics enable the human expert to make educated guesses when necessary, to recognize promising approaches to problems, and to deal effectively with wrong or incomplete data.

Abstractly, knowledge consists of descriptions, relationships, and procedures in the domain of interest. The descriptions are sentences that identify and differentiate concepts, objects, and classes. Relationships describe the interactions, dependencies, and associations between items in the knowledge base. Typically, they describe taxonomic, definitional, and empirical associations. Procedures specify operations to perform when attempting to reason or to solve a problem. Elucidating, capturing and reproducing all this knowledge allow creation of the knowledge base.

The inference engine is the tool that applies the reasoning rules and the specific procedures of the experts to the expert knowledge base to find the desired solution. This component has to identify and apply some strategies to look for and derive the solution from the available knowledge, as efficiently and effectively as possible, without exploring blindly the whole space of possible solutions (which is usually too wide). To achieve this goal, both general problem-solving abilities and domain-specific methods are available as searching strategies. A rational and robust ability to organize the sequence of decisions and rule application is essential for efficiency and effectiveness.

A more detailed structure of an ideal expert system is shown in Fig. 4; actual systems may contain only some of these components. The language processor supports problem-oriented communications between the user and the expert system, usually in some textual restricted variant of the natural language or even in structured or graphic ways. It parses and interprets user questions, commands, and volunteered information, and it formats information generated by the system, including answers to questions, explanations, justifications for its behavior, and requests for data as well.

The blackboard is a storage area that is used to record the intermediate results of the search operations, namely, intermediate hypotheses and decisions that the expert system manipulates (plan, agenda, and solution elements). The plan describes the overall or general attack the system will pursue against the current problem, including current plans, goals, problem states, and contexts. The agenda records the potential actions waiting for execution, which generally correspond to knowledge base rules that seem relevant to some decision placed on the blackboard previously. The solution elements represent the candidate hypotheses and decisions the system has generated so far, along with the dependencies that relate decisions to one another.

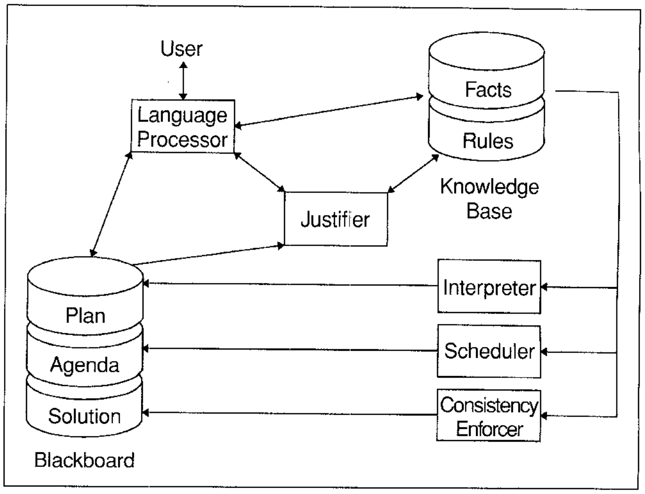

Figure 4 *The ideal structure of an expert system.*

A scheduler controls the order of rule processing, maintains the agenda, evaluates the priorities of the pending actions, and estimates the effects of applying the potential rule. The interpreter executes the chosen agenda item by applying the corresponding knowledge-base rule; it validates the relevance conditions of the rule, binds variables in these conditions to particular solution blackboard elements, and makes those changes to the blackboard that the rule prescribes.

A consistency enforcer adjusts previous conclusions when new data or knowledge alter their bases of support, maintaining a consistent representation of the emerging solution and avoiding inconsistent solutions.

A justifier rationalizes and explains system behavior to the user, by explaining why and how some conclusions were reached or why some alternative was rejected. For this purpose, the justifier traces backward the solution elements from the conclusion in question to the intermediate hypotheses or data that support it.

The knowledge base comprises information and facts as well as heuristic planning and problem-solving rules that may be useful in formulating a solution. It is worth noting that expert systems are able to solve problems efficiently and accurately only as long as there is enough knowledge in the base and the operating rules are effective. Therefore, the configuration of the expert system needs to be as complete and exhaustive as possible in the specific domain envisioned, even if the knowledge of experts may often be difficult to be elucidated, captured, and formalized.

The use of expert systems in instrumentation and measurement may encompass several areas, basically related to all forms of classification and prediction. However, due to the intrinsic complexity of the operations performed by the expert system to identify the solution and due to the large space of solutions to be explored, real-time applications cannot usually be envisioned.

Conversely, the larger is the knowledge base, the more accurate may be the solution identified. Since the quantity of information that can be stored in the knowledge base is usually much larger than the information held by realizable neural networks or fuzzy systems, expert systems may be more effective and accurate than adaptive techniques, even if definitely slower.

The expert systems used to interpret data in a specific domain (interpretation systems) may be viewed as high-level sensors: they explain observed data by associating symbolic meanings describing the situation or system state. These systems are, for example, able to perform surveillance, speech understanding, image analysis, and signal interpretation. Diagnostic systems are another application of high-level system analysis; they are able to infer system malfunctions from observed behavior irregularities and to relate them to the underlying causes. Monitoring expert systems are able to observe the application system and to point out features that seem crucial to correct the application system behavior.

Prediction and system identification imply an ability to deduce consequences from given situations, system inputs, and states. An expert system usually represents knowledge as dynamic nonlinear models whose parameters are tailored to the specific application case. Control systems can also be realized by using expert systems, however, only high-level decisions and overall set points can be taken by such systems, since they are not able to operate in the real-time environments typical for most machinery and plants.

Design, configuration, and calibration of instruments are other areas in which expert systems can be exploited, since they are able to capture expert knowledge in these critical issues. These tools can be used to design optimized measurement strategies, according to the available instrumentation and to the environmental conditions, as well as to predict the final accuracy and uncertainty of the achieved measurement results. Similarly, they can be used to deduce the final uncertainty of a measurement result generated by complex processing (e.g., with DSP or computer-based techniques) of the measurement results, which are likely to be affected by uncertainty.

Conclusions and Methodological Remarks

Artificial intelligence techniques have been reviewed in this paper for their possible and feasible uses in instrumentation and measurement. A large number of practical application cases have been explored, and many outstanding and attractive results have been summarized.

Adaptive technologies have been shown to be effective in several application areas. The relative novelty of neural networks and fuzzy logic and their intrinsic complexity have not yet allowed a truly systematic and exhaustive analysis, a comprehensive presentation, or the definition of general and standard methodologies for their application to real cases. We can find many effective and efficient applications in the literature and on the market, but each application still requires an in-depth analysis of the paradigm characteristics and an extensive experimentation, mainly based on the experience and the expertise of the system designer rather than on a well-assessed design methodology. Some design guidelines are available in specific application areas, with reference to results previously achieved in similar applications. As a consequence, the successful use of these techniques is often nondeterministic. A great deal of research is in progress not only to expand the application areas and to explore new capabilities and new paradigms, but also to identify and assess systematic procedures to afford and solve real application cases with the same high confidence that the designer has in more traditional technologies.

Expert systems have been shown to be effective in reproducing human reasoning in specific areas related to deduction and inference. In particular, they represent attractive techniques capable of capturing the knowledge and reasoning of experts in specific fields. Behavioral analysis and diagnosis of instruments and systems are therefore well suited for these tools. Diagnosis can also be viewed as the high-abstraction-level measurement of a system feature: the correctness of the system behavior. The complexity of the operations and the computation required to achieve the solution are not suitable for satisfying the real-time measurement constraints (in particular, the difficult ones) that are often present in control systems or in embedded applications.

Success in the use of the above technologies and, specifically, of adaptive techniques, does not suggest that they are the best approach to solve any problem. In particular, the designer has to be careful in considering these paradigms as panaceas (the complexity of problems has to be acknowledged), or in working with a problem for which an algorithmic technique is not known. In practice, in fact, some people believe that neural networks, fuzzy logic, and expert systems are able to solve any problem better than any other technique; this is obviously not true. Any approach has its own positive aspects and limits. At the moment, the main drawback for such techniques consists of the a priori unpredictability of the possibility, effectiveness, and accuracy in solving a given problem. On the other hand, designers should not try to solve every problem with these paradigms, in particular when an algorithmic solution is known to perform well and has a reasonable complexity. In addition, the use of these approaches should be confined to solving only the specific tasks of an entire application for which there is no effective algorithmic solution. This leads to creating heterogeneous systems by mixing classic algorithmic solutions, neural, fuzzy, and expert-based components that best exploit the specific capabilities of the individual technologies.

All artificial intelligence techniques try to capture some aspects of a common knowledge-based approach to the solution of complex problems. Neural networks afford the distribution of the information storing and processing, while fuzzy logic introduces nondeterminism in the computation. Expert systems create sets of rules to capture the knowledge and the behavior of the experts. Other artificial intelligence techniques, such as genetic algorithms, probabilistic reasoning, and chaos theory, have been introduced in the literature but are not yet assessed enough to be widely used in our field as autonomous approaches.

Since all of the above techniques try to attac, complex computing problems from different points of view, research has been performed to merge some of these approaches. Examples are, in fact, neuro-fuzzy systems, the inclusion of fuzzy logic and neural networks in expert systems, and the use of genetic algorithms in neural networks, fuzzy logic, and expert systems. In the future, more research should consider integration and fusion of these techniques as well as other emerging approaches into hybrid systems, since remarkable advantages and advancements might be achieved for the design of adaptive systems based on knowledge and learning.

The use of artificial intelligence techniques introduces a new set of problems related to the characterization of the new instrumentation and measurement procedures, incorporating some components based on these information-processing technologies. Research is needed to define and evaluate suitable methodologies and techniques in order to identify and specify the accuracy, the precision, and the confidence of the measurements performed by using these advanced instrumentation and measurement procedures, in particular with respect to the algorithmic choices embedded in the related software and the computing system itself. For the near future, these are the challenges of the research communities in the areas of instrumentation and measurement as well as in computer science and engineering.

References

[1] S. Haykin, *Neural Networks: A Comprehensive Foundation*, New York: McMillan and IEEE Computer Society, 1994.

[2] D.T. Pham and X. Liu, *Neural Networks for Identification, Prediction and Control*, London: Springer-Verlag, 1995.

[3] R. Linggard, D.J. Myers, and C. Nightngale, eds., *Neural Networks for Vision, Speech and Natural Language*, London: Chapman & Hall, 1992.

[4] H. J. Zimmermann, *Fuzzy Set Theory and its Applications*, 2nd ed., Boston: Kluwer Academic Publishers, 1991.

[5] R.J. Marks II, *Fuzzy Logic Technology and Application* IEEE Press, 1994.

[6] B. Kosko, *Neural Networks and Fuzzy Systems*, Englewood Cliffs, NJ: Prentice-Hall, 1992.

[7] D. Nebendahl, *Expert systems: Introduction to the technology and applications*, New York: John Wiley & Sons, 1988.

[8] C. Alippi, V. Piuri, "A selected bibliography on artificial intelligence techniques in instrumentation and measurement applications," http://www.ims.unico.it/tech_com/tc-22/techinfo/ biblio.html, 1997.

Cesare Alippi *obtained the Dr. Ing. degree in Electronic Engineering in 1990 and the Ph.D. in Computer Engineering in 1995, both from Politecnico di Milano, Milano, Italy. His furthur eduction includes research work in computer sciences carried out at the University College London and MIT. Currently, Dr. Alippi is a research fellow within the Italian National Research Council. His research interests include neural networks (learning theories, implementation issues and applications), genetic algorithms and nonlinear signal and image processing, and his research results have been published in more than 60 technical papers in international journals and conference proseedings. Dr. Alippi is a member of the IEEE Circuits and Systems Society. He can be contacted at alippi@elet.polimi.it.*

Alessandro Ferrero *received the M.S. degree in Electrical Engineering from Politecnico di Milano, Milano, in 1978. In 1983, he joined the Dipartimento di Elettrotecnica of the Politecnico di Milano as a Researcher on Electrical Measurement. From 1987 to 1991, he was Associate Professor of Measurement on Electrical Machines and Plants at the University of Catania, Catania, Italy. From 1991 to 1994, Professor Ferrero was Associate Professor of Electrical Measurements at Politecnico di Milano. He is presently Full Professor of Electronic Instrumentation at Politecnico di Milano. His current research interests are concerned with the application of digital methods to electrical measurements. Professor Ferrero is a member of the Italian C.N.R. Group on Electrical and Electronic Measurements, and the North Italy Chapter of the IEEE IM Society. He can be reached at ferrero@bottani.etec.polimi.it.*

Vincenzo Piuri *obtained the Dr. Ing. degree in Electronic Engineering in 1984 and the Ph.D. in Computer Engineering in 1989, at the Politecnico di Milano, Milano, Italy. Since 1992, he is Associate Professor in Operating Systems at Politecnico di Milano. His research interests include distributed and parallel computing systems, application-specific processing architectures, computer arithmetic, fault tolerance, neural network architectures, theory and applications of neural techniques. Original results have been published in more than 120 papers in international journals and conference proceedings. Dr. Piuri is an Associate Editor of the IEEE Transactions on Instrumentation and Measurement, the founding chair of the Technical Committee on Intelligent Measurement Systems of the IEEE Instrumentation and Measurement Society, and a member of IEEE, IMACS, INNS, and AEI. He can be contacted at piuri@elet.polimi.it.*