Cesare Alippi, Stefano Ferrari,
Vincenzo Piuri, Mariagiovanna Sami, and Fabio Scotti

New Trends in Intelligent System Design for Embedded and Measurement Applications

Intelligent systems adopt soft-computing techniques (encompassing neural networks, fuzzy logic, genetic algorithms, and expert systems) to solve complex problems by mimicking human reasoning. On the other hand, conventional algorithmic approaches are extremely powerful and efficient in tackling applications for which a procedural solution can be easily defined. By themselves, each of these techniques may be the optimal solution for a subproblem, but not efficient enough to solve the problem as a whole. Composite systems, consisting of conventional and soft-computing components in cooperation, are now more than a promise to face complex application needs. In this article we present recent advances in the design of composite systems, with specific reference to embedded and measurement applications.

Complex Applications Need Complex Solutions

The design of embedded systems usually requires a preliminary specification phase involving identification and definition of a number of different aspects, such as:

- The identification of the relevant application variables (such knowledge must characterize the process insofar as possible)
- The selection of the most appropriate sensors
- The definition of the correct output signals (e.g., to control the system or inform the user)

©Corel Corporation

1094-6969/99/$10.00©1999IEEE

- The characterization of the actuators and definition of the output signal interfaces
- The definition of the input-output relationship

A formal specification is by now widely recognized as a support for a preliminary validation phase (allowing, among other things, the checking of the initial specifications for completeness, congruence, etc.) as well as a starting point for the subsequent design steps. In particular, design will focus on the identification of an abstract architecture realizing the input-output relations. Available system-level approaches adopt "conventional" abstract architectures; the subsequent design flow usually involves a hardware-software partitioning phase prior to the synthesis phase.

Soft-computing techniques [1], [2] have been developed to deal with problems whose solutions cannot be efficiently realized with classic techniques. In fact, researchers observed that even if the formalization of a problem's solution is difficult, humans are usually able to quickly identify an acceptable solution. This task can be obtained by somehow comparing, through similarity and generalization, the given problem with personal experience. These nontraditional approaches encompass neural networks, fuzzy logic, genetic algorithms, and expert systems. Their flexibility and expressiveness are acquiring an increasing relevance with a primary impact in the instrumentation-and-measurement community. The literature provides several application solutions, with a particular emphasis on industrial control, signal and image processing, sensor fusion, system diagnosis, and high-level processing.

Even if soft-computing techniques are extremely accurate in solving specific tasks, in general, they are not a panacea to fulfill all requirements of complex industrial applications (e.g., real-time requirements, computational complexity, hardware resources, and throughput). Soft-computing paradigms may hence become too complex and inaccurate in capturing the behavior of complex systems. Besides, they could introduce further difficulties in defining a suitable procedure to tailor them to the envisioned application.

The use of modularity in a top-down system design allows for partitioning the whole application into smaller and, possibly less-complex, subproblems. Such problems, in general, interact and may be intended as atomic units that can be tackled separately. The possibility of reusing existing processing modules reduces the complexity and the time necessary to provide an overall design solution for the application; here personal experience is a fundamental requirement.

The concept of using modularity to tackle complex systems was recently introduced in the soft-computing literature. However, its use was basically conceived to deal with very specific cases. When the global behavior of a system cannot be described satisfactorily by a unique soft-computing model, several models may be needed, each acting locally to solve an application subtask. Hierarchical paradigms are thus envisioned either to select the most suitable model or to merge the different models that are each valid in a working point. It is worth noting that this approach is not sufficient, in general, to partition a complex application into smaller interacting parts, all of which can be described by means of soft-computing paradigms.

Composite Systems: An Integrated Comprehensive Approach

In many cases, the application can be split so that some parts of the solution can be found quite easily by using traditional data processing, while other parts can be treated more effectively by considering soft-computing methods.

An improved solution can then be achieved through cooperation of traditional conventional and soft-computing methods. Integration of these technologies into a ***composite system*** allows for exploiting the best and most effective characteristics of each of them.

Some typical examples can be found in multisensor architectures. Sensor fusion is required either because of the extreme heterogeneity of sensors or because of the complexity of the data-collection phase, possibly in a noise-affected environment. Moreover, the problem of sensor aging and drift, hardly tackled in most conventional solutions, cannot be neglected and needs adaptive management of the sensor system.

Design of Composite Systems: From Specification to Implementation

At present, system-level design techniques developed for telecommunication and DSP environments do not appear capable of coping satisfactorily with the kind of problems outlined above—those that quite often require adoption of both conventional and non-algorithmic (e.g., neural, fuzzy, genetic, expert-system) solutions within a single composite system.

The crucial step in developing composite systems and, in particular, composite embedded systems, is to integrate current system-level design solutions with innovative techniques oriented to take into account the problems mentioned previously. These systems will be of great interest in a number of applications related to industry and daily life; in many cases, such applications will encompass problems typical of the instrumentation and measurement areas. The focus is, in fact, on applications requiring adaptive sensor data management (by soft-computing approaches) simultaneously with standard data processing supported by a microprocessor system running a real-time kernel.

Even if it is reasonably easy, although not trivial, to intuitively characterize what the expected solution for a given problem will be, a formal specification of the requirements of the solution and the partitioning between conventional and soft-computing solutions is anything but obvious. Do we need a robust solution, or is accuracy the hot point? Are our data affected by noise with unknown features, or is the available information noise-free? Is real-time computation a problem? The set of requirements drives designer toward the most appropriate partitioning of the problem's solution—into traditional and soft-computing parts.

For high-level composite system design, a number of methodological problems must be solved, leading to the definition of the innovative design approach shown in Fig. 1:

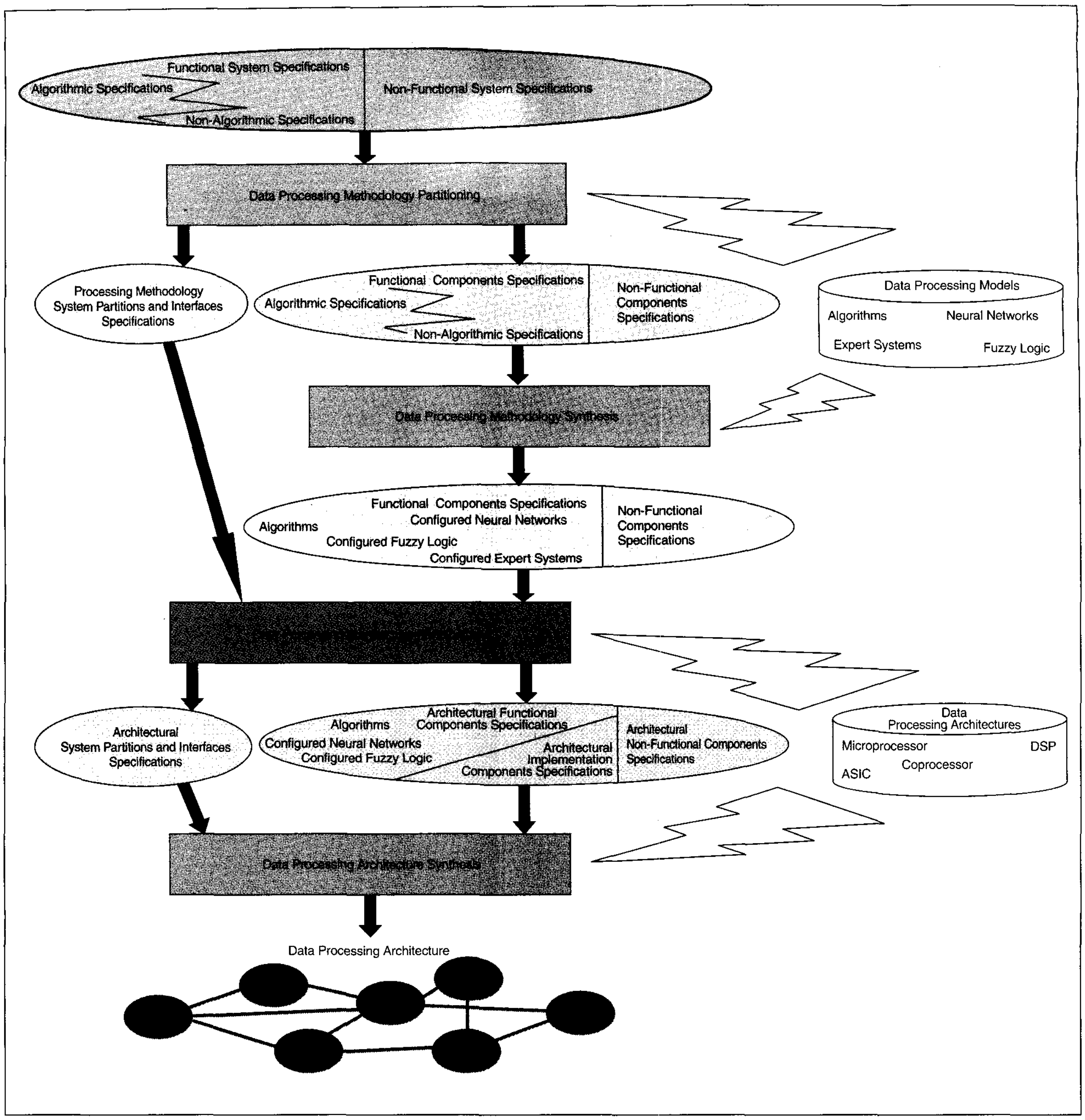

Fig. 1. A comprehensive design methodology for composite systems.

- Analysis and selection of the specification language related to an abstract reference problem. This allows the system to be designed as a collection of modular components (objects), independent of their implementation (as conventional algorithms or as soft-computing computational paradigms), or if they are mapped into software or hardware
- Partitioning of system activities among traditional and soft-computing activities
- Hardware/software partitioning for the application
- Definition of the most suitable soft-computing paradigms, taking into account sensitivity and discretization issues
- Design and development of a framework that allows the system to be tested, and its performance to be evaluated, in an emulation environment on a host system

In summary, we need to specify the system, synthesize the data-processing methods, and synthesize the hardware/software architecture for data processing.

The above is an extension of present co-design techniques [3], including two orthogonal perspectives: the hardware/software partitioning aspect (where several research groups are currently focusing their attention), and the traditional/soft-computing partitioning problem, (still addressed by a few researchers, e.g., at Politecnico di Milano). To afford all points, a comprehensive design methodology is required

Modeling a Thermoelectric Power Plant for Monitoring and Control

Efficient simulation environments for complex applications on low-cost computing systems are mandatory for limiting costs and allowing competitiveness in several industrial areas related to system and plant automation. These include the implementation of advanced control systems, training of the personnel working in complex plants, providing real-time decision support, and creating computer-based intelligent automatic systems. As a result, the quality of industrial plants and products, as well as the performance of operators can improve.

As a relevant example we consider a thermoelectric plant for power generation (Fig. 2). A complete and accurate model can be obtained by relying on the physical description of the system (i.e., by writing the set of mathematical equations that describe all the physical phenomena.) The resulting equations are complex, nonlinear, and (generally) differential; the complexity of the description itself may become impractical for small- and medium-sized computers and, as a consequence, may require powerful computing systems to deal with real-time application constraints. In addition, not all variables in the mathematical description can be easily measured with standard sensors and instruments, due to either their characteristics or their location in the system, so that measurement may become expensive or even impossible for the envisioned application. Many state variables in the system equations are intrinsically unmeasurable since they do not correspond to measurable, discrete physical quantities, and must be deduced from physical quantities through suitable equations. Finally, the physical phenomena underlying the system behavior may not be known completely and accurately, as generally happens when there are strong uncertainties and nonlinearities.

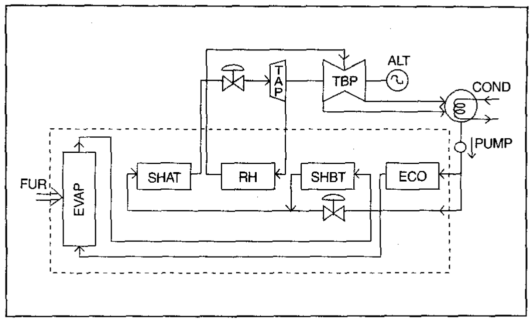

Fig. 2. The simplified scheme of a thermoelectric plant for power generation: high-pressure turbine TAP, low-pressure turbine TBP, alternator ALT, condenser COND, circulation pump PUMP, furnace FUR, boiler EVAP, high-pressure superheater SHAT, low-pressure superheater SHBT, recycling superheater RH, and economizer ECO.

---

for composite systems by extending and expanding solutions and standards already proposed in the literature.

The research focuses on various aspects of the comprehensive design methodology. The first objective consists of analysis and possible extension of design methodologies to support an object-oriented description of embedded systems, including both functional specifications and nonfunctional constraints (e.g., accuracy, timing, power, and scalability), in a way suited to represent both traditional tasks and soft-computing paradigms within a homogeneous development framework.

To specify the desired system behavior, functional characteristics must first be defined. In static systems, the expected output must be given for each input. In digital implementations, this characterizes the combinatorial function solving the application. In dynamic systems, each pair of inputs and system states must be related to the outputs and following states. Digital realization is derived from the state diagram characterizing the finite state machine. Formal specifications of these kinds of systems are now typically performed by using sequencing graphs [4]. Industrial plants and processes are typical analog systems, having the input-output relationships described by differential equations, typically continuous-valued and possibly involving partial derivatives.

Crisp values, possibly with a given uncertainty, are traditionally used to represent and process data. Fuzzy values can be viewed as a generalization of the crisp ones when the envisioned characteristic is not represented by one crisp value, but by a deterministic collection of them. Fuzzy systems are described by fuzzy rules, i.e., by means of an algorithmic—even if nontraditional—description of operations to be applied to generate the desired outputs. Expert systems apply rules to the operands to explore the space of possible solutions, looking for an acceptable one. When the behavior cannot be formally defined by one of the previous relationships' mapping inputs and states onto outputs and next states, description by examples can be considered. Neural networks are defined by the training set. In static networks, the desired behavior is specified by the input-output pairs or by the input set for the supervised or unsupervised networks. In dynamic networks, the state of the system is captured by the ordered sequences of input-output pairs.

The large composite system was, therefore, partitioned into smaller subsystems with clear and specific interactions among them. Partitioning was obtained by analyzing the power plant operation and the main exchanges between modules. Conventional equation-based or innovative soft-computing models were adopted to describe the component behavior via a black-box approach, and integrated in the simulation environment to better exploit the unique characteristics of conventional and soft-computing techniques. Neural modeling was shown to be particularly effective in compacting the model description for the furnace, the superheater, and the turbine inlet.

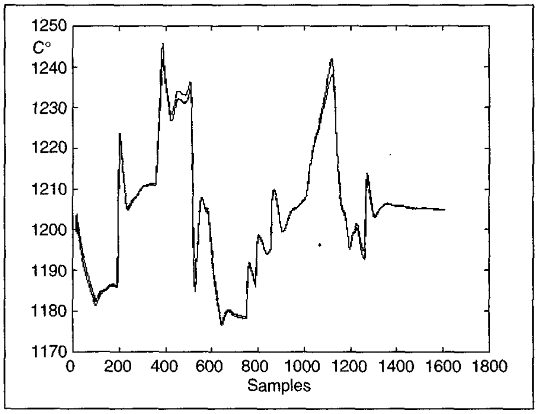

Fig. 3. The superheater: validation of the smoke temperature.

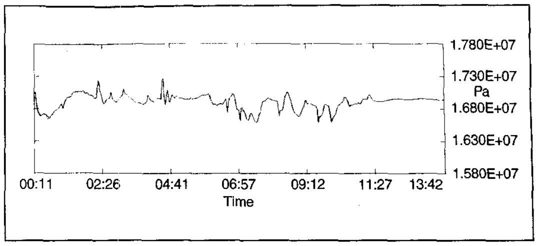

Fig. 4. The turbine inlet: validation performance in the highly perturbed case.

For the other components, we considered more traditional black-box approaches such as ARMAX (for simplicity) or physical models (for accuracy). Accuracy achieved with neural modeling is comparable or even better than that of the traditional model for the fume temperatures of the high-pressure superheater SHAT and the turbine inlet pressure, respectively (Figs. 3 and 4). The computational complexity of the entire software simulator containing composite models was dramatically reduced to about one order of magnitude so that effective and realistic simulations were performed on PCs based on a Pentium 166 MHz instead of requiring a $30,000 workstation.

From the high-level point of view, we can, for example, adopt the ordered sequences of input-output pairs as a good general system description, independent of the computing paradigm that will be used to realize each component. Specification of a complex system comprises interconnected groups of ordered sequences. Incomplete specification occurs whenever some of the sequences of input-output pairs are unavailable. We can easily see that all of the above cases can be derived from this general description by forcing one of the characteristics. For example, the combinatorial functions are obtained by assuming that each sequence contains only one input-output pair of digital data.

The second objective of the comprehensive design methodology is the development of techniques supporting partitioning between traditional and soft-computing components by also taking into account the nonfunctional constraints. Traditional and soft-computing models can be functionally equivalent, even if their expressiveness, completeness, conciseness, and nonfunctional specifications (e.g., accuracy) may differ. The adoption of one of these models depends on the balanced (not necessarily optimum) satisfaction of all the application requirements. For example, if a linear model describes the system with sufficient accuracy with respect to the envisioned application, there is no need to look for a more effective, but more complex, neural model. The choice of the model to be used for a component may have great impact on the subsequent implementation phase since it may induce, among others, different computational complexity, performance, and power consumption.

Partitioning of the system specifications must, therefore, identify boundaries among components and the related interfaces so that each of these components can be efficiently and effectively implemented. Partitioning looks first to natural and evident boundaries defined in the specifications by the designer. Then, it needs to be guided through splitting the components into simple subsystems that can be tackled by one modeling technique directly with good results. Measurement of the expected complexity (e.g., the number of input-output pairs in the specifications), as well as of the quality of the result must be introduced as indices of the partitioning quality. Aggregation and separation techniques should be taken into account to group homogeneous components.

Electronic Control for Automotive Applications

We are experiencing an increasing interest in problems related to the environment, with a particular focus on pollutants generated by vehicles in industrialized countries. These strong constraints pushed research toward the development of suitable electronics, embedded systems, and mechanical and chemical devices to reduce noxious emissions. Unleaded fuel, catalytic converters, and fine control of the variables involved in the fuel-combustion process are relevant ingredients in reaching such a goal. Fig. 5 shows an example of a fuel-injection system comprising a spark-ignition engine with a catalytic converter and a linear oxygen sensor on the exhaust manifold to measure the air-to-fuel ratio (AF) after the combustion process. The electronic control module (ECM) must also perform, in addition to various service tasks, the real-time control of the electronic injection to guarantee the stoichiometric value of the AF ratio for optimal catalysis of pollutants.

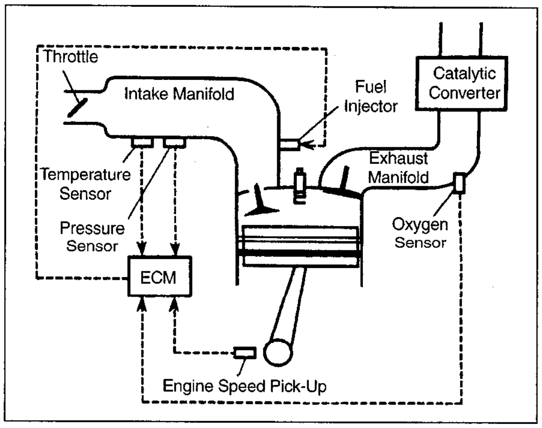

Fig. 5. The fuel-injection engine and the control system.

Often, classical control systems in automotive applications are not able to deal completely and accurately with strong nonlinearities, noisy data, and sensor aging, as well as adaptation to the installed catalytic system. We developed and successfully tested a composite solution based on neural technologies for the injection controller. The operations performed in the ECM were partitioned via conventional and soft-computing techniques according to their known abilities in dealing with nonlinearity and noise. Among the conventional ones are monitoring and control of fluid levels, braking, lighting, and indicators. Whenever necessary, data acquisition and sampling are performed by using conventional A/D converters, while signal generation is accomplished through D/A converters. The digital ECM can then be redesigned by incorporating the FPGA-based implementation of the neural computation into the microprocessor-based structure for the other operations.

The controller was developed by adopting the classic indirect control configuration (Fig. 6). The block I identifying the behavior of the process P is used to configure the controller C so as to act as the reference R. From an accurate analysis of the physical model of the combustion engine, some uncertainties are evident, namely the quantity of

Strict relationships and dependencies between the component model and the system partition make the definition of partitioning and the choice of the model for each component inseparable. Even if abstract specification is independent from the implementation methodology, the evaluation of the expected characteristics at a high abstract level relies on the prospective realization technologies envisioned as possible alternatives to producing the component. Partitioning can be performed by grouping small homogeneous components and by splitting large system specification blocks. Then, some exploratory models for each component are created and tested with respect to the relevant figures of merit for the application. For traditional components, the procedure describing the desired computation is given. For soft-computing components, the suitable corresponding synthesis is performed. For example, in neural models, the learning procedure is applied to configuring the network, resulting in an algorithmic description of the network operation. For statistical models, the parameters are identified on the available data by using the traditional statistics. The various exploratory models are evaluated and compared with respect to the suitable figures of merit. The best modeling result is adopted as a reference in the possible subsequent exploration of the solution space.

At the end of this modeling and partitioning phase, we obtain the set of traditional and soft-computing components constituting the system as well as the interfaces. It is worth noting that, at this point, the synthesis of the data-processing method has been perfected and all components are described algorithmically. In conventional methodologies concerning only traditional components, the synthesis of the processing method is implicit and coincides with the algorithms defined in the specifications.

The third objective of the integrated design methodology is the extension of design methodologies capable of hardware/software partitioning to include the implementation of

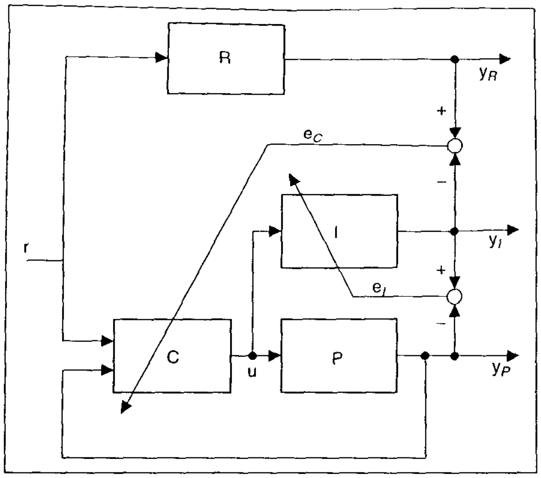

Fig. 6. The indirect control configuration.

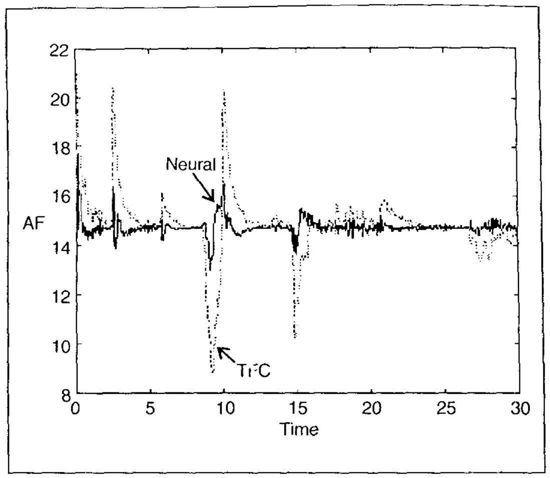

Fig. 7. The neural and the TFC control performance.

fuel that condenses on the manifold walls and the fuel-evaporation time constant. For each of these quantities, the identified neural model consists of a feedforward neural network of the regression type, namely, a three-layered neural network (with 13 and 10 hidden neurons in the optimal structure). This is a clear example of integration in which information coming from soft-computing analysis is then used by more traditional ones (the AF controller) to improve the description of the process.

The neural identifier for the AF ratio was then created by using a recurrent single-layered neural network with 15 hidden units and a single linear output. The suitably delayed output, as well as external inputs and their delayed values, was presented to the network inputs. The controller of the AF ratio was finally derived by configuring this last neural paradigm; for the controller, only 10 hidden neurons were necessary. The achieved accuracy is shown in Fig. 7, where, as mentioned, the ultimate goal is to keep the AF ratio at its stoichiometric value of 14.67. The accuracy is compared to that of the transient fuel film compensation (TFC) control, which is the most frequently used traditional technique. The improvement is about 30% on the average, while the computational complexity is about the same.

soft-computing subsystems. In the literature, hardware/software co-design techniques are available and widely used to realize dedicated digital systems from algorithmic specifications. Because we've reduced any modeling method to the algorithmic description of the operations computing the model itself—via model parameter configuration on the envisioned application—the co-design techniques can be directly applied to composite systems.

Various architectural solutions are taken into account, encompassing hardware, software, and mixed parts. These range from the fully programmable, general-purpose structures based on microprocessors to the computing-intensive solutions containing DSP processors; from the co-processor-based structures to the special-purpose processors; and from the configurable-computing architectures based on field-programmable gate arrays (FPGAs) to the fully dedicated, application-specific integrated circuits (ASICs). Similar counterparts are considered for analog modules.

Partitioning between hardware and software is guided by suitable figures of merit concerning the nonfunctional specifications (e.g., performance, real-time operation, throughput, operation complexity, circuit complexity, and power consumption). Also, in this case, the partitioning is based on the evaluation of exploratory implementations of the processing architecture. Hardware/software partitioning is independent of the traditional/soft-computing partitioning. An algorithm can be implemented in hardware or translated in software running on a microprocessor or DSP processor. Similarly, for example, a neural network (once configured) can be realized with a dedicated neural ASIC, an FPGA, a programmable neurocomputer, a DSP processor, or a microprocessor.

At the end of the hardware/software partitioning, we obtain the overall architecture of the processing system as well as the detailed structure to be used for each component. Synthesis of the processing architecture can thus be performed by using traditional techniques, namely, programming for software components and digital/analog synthesis for hardware parts.

To increase the efficiency of the integrated synthesis methods as well as the resulting quality, component libraries can be introduced, possibly for specialized classes of applications.

On-Line Profile Reconstruction

Real-time profile analysis and reconstruction is an important problem in many industrial applications, encompassing, for example, product-quality monitoring and assessment in steel and mechanical industries, object recognition in automated machinery, wear monitoring and diagnosis in production plants and railways, and quality analysis and control in the automotive production processes.

Tactile techniques were traditionally adopted, but they have great limitations due to wear, especially in the case of moving objects. Laser and X-ray imaging are now attracting great interest due to the non-intrusive and non-contact nature of profile acquisition allowed by these approaches. Figs. 8 and 9 present the system setup and the image captured by the charge-coupled device (CCD) on structural steel, respectively. Whenever the frequency of image acquisition is high, suitable real-time processing architectures must be envisioned.

When the amount of data acquired from sensors (i.e., the images) becomes too large, storage may be difficult or even impossible. In addition, in several practical applications, only a few parameters per image may be relevant, so storage of entire images for subsequent analysis is not of interest. On-line image processing and high-performance architectures are, therefore, mandatory to deal with these cases. Accuracy is a key factor in industrial applications, and it is relevant to guarantee the quality of the production process; the design of the measurement systems must include such requirements in the specifications so as to drive the subsequent design steps and to certify final quality. Adapting to aging, wear, noise, reflections, and environmental variations is very useful in incorporating these characteristics into the system model, even if they may not be completely known in advance.

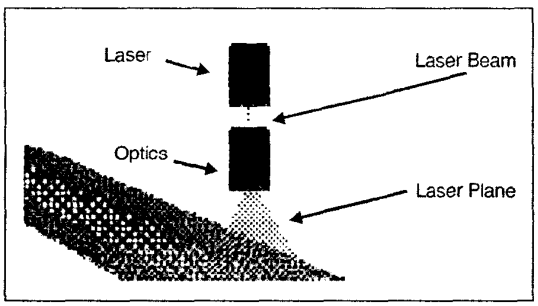

Fig. 8. The detection system.

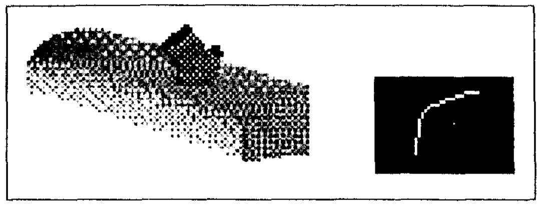

Fig. 9. A CCD image and the profile reconstruction.

In many applications (e.g., the ones mentioned above for analysis of quality and wear of structural steel) the position of the profile in the image is usually known in advance, even if to a large approximation, since the relative position of the steel and the cameras is approximately given. In this case, some simplifications in the image analysis for profile identification can be introduced. Specifying this additional information is of great help in the system design as it allows for taking advantage of better a priori knowledge, reducing the computational complexity and enhancing accuracy.

By adopting a composite system, we can consider a conventional pattern-matching filter to identify the area of interest containing the profile, while a neural network may perform the fine profile reconstruction at subpixel accuracy.

The final synthesis of the system prototype leads to the creation of a distributed architecture comprising one low-cost, DSP processor per image acquisition channel and one central, PC-based processing system. Each DSP processor runs the algorithmic pre-filtering and the neural profile reconstruction, while the central system performs profile comparison and storage. System implementation is, therefore, obtained in software on standard, general-purpose and DSP processing components. For higher throughput, ASICs or FPGAs on dedicated boards could be considered.

This allows for storing results of previous experiences in a shared database for future reuse, by mitigating the top-down design approach with intelligent consideration of bottom-up issues. In addition, as in any good engineering design methodology, feedback loops can be introduced along the design path to modify the adopted choices whenever appropriate to enhance the overall quality of the final comprehensive solution.

Future Research and Development

The definition of an integrated design methodology for composite systems and the related CAD environment is a challenging research area for the near future. Diversity and specialization within a coordinated, homogeneous, interacting framework are, in fact, key factors for optimum solutions of many application problems.

Benefits are expected in the optimization of many systems, including embedded ones, which are increasingly surrounding our daily life even if we are not fully aware of them. Many of these systems are either directly or indirectly related to measurement issues, especially when accurate control and monitoring are concerned.

The main final goals of this applied research are system optimization with regard to both performance and features, and design automation to reduce the time to market. However, we are only at the beginning of these promising activities. Much effort will be required in the near future to make this integration real and easily accessible to engineers.

References

[1] A. Ferrero and V. Piuri, "Soft-computing and virtual environment-based application," in *Instrumentation and Measurement Technology and Applications*, E.M. Petriu, Ed., New York: IEEE Press, 1998, pp. 453-457.

[2] W. Pedrycz, *Computational Intelligence, an Introduction*, New York: CRC Press, 1998.

[3] *Hardware/Software Codesign*, G. De Micheli and M.G. Sami, Eds., NASO ASI Series, Norwell, MA: Kluwer Academic Publishers, 1995.

[4] D.D. Gajski, F. Vahid, S. Narayan, and J. Gong, *Specification and Design of Embedded Systems*, Englewood Cliffs, NJ: Prentice Hall, 1995.

Cesare Alippi *obtained his Ph.D. in Computer Engineering in 1995, from Politecnico di Milano, Italy. He is Associate Professor in Computer Science at Politecnico di Milano and Research Fellow at the Italian National Research Council. His interests include neural networks (learning theories, implementation issues, and applications), genetic algorithms, nonlinear signal and image processing, and advanced processors. Main results have been published in more that 70 technical papers in international journals and conference proceedings.* ***Stefano Ferrari*** *obtained his Dr. Ing. degree in Computer Science from the University of Milan, Italy, in 1996. He is a Ph.D. student at Politecnico di Milano, Italy. His research interests include signal and image processing, neural technologies for image processing, high-level synthesis of complex systems.* ***Vincenzo Piuri*** *obtained his Ph.D. in Computer Engineering in 1989, from Politecnico di Milano, Italy. Since 1992, he has been an Associate Professor in Operating Systems at Politecnico di Milano. His research interests include distributed and parallel computing systems, theory and applications of neural techniques. Results have been published in more than 120 papers in book chapters, international journals, and proceedings of international conferences. He is founding Co-Chair of both the Technical Committee on Emerging Technologies and the Technical Committee on Intelligent Measurement Systems of the IEEE Instrumentation and Measurement Society. He is Associate Editor of the* IEEE Transactions on Instrumentation and Measurement. ***Mariagiovanna Sami*** *obtained her Dr. Ing. degree in Electronic Engineering from Politecnico di Milano in 1966 and her Libera Docenza (Computing and Switching Theory) in 1971. Since graduation, she has been with the Department of Electronics, Politecnico di Milano, where she obtained various research and teaching positions and where she has been a full professor since 1980. Her research interests are in the areas of fault tolerance, highly parallel VLSI and WSI architectures, and high-level synthesis. She has published approximately 150 papers in international journals and conference proceedings; she is co-author of the book* Fault Tolerance Through Reconfiguration in VLSI and WSI Arrays, *published by MIT Press. She is editor-in-chief of the* EUROMICRO Journal *and she has been on the board of editors of the* IEEE Transactions on Computers. ***Fabio Scotti*** *obtained his Dr. Ing. degree in Electronic Engineering from Politecnico di Milano, Italy, in 1998. He is a Ph.D. student at Politecnico di Milano, Italy. His research interests include signal and image processing, neural technologies for image processing, and optical technologies.*