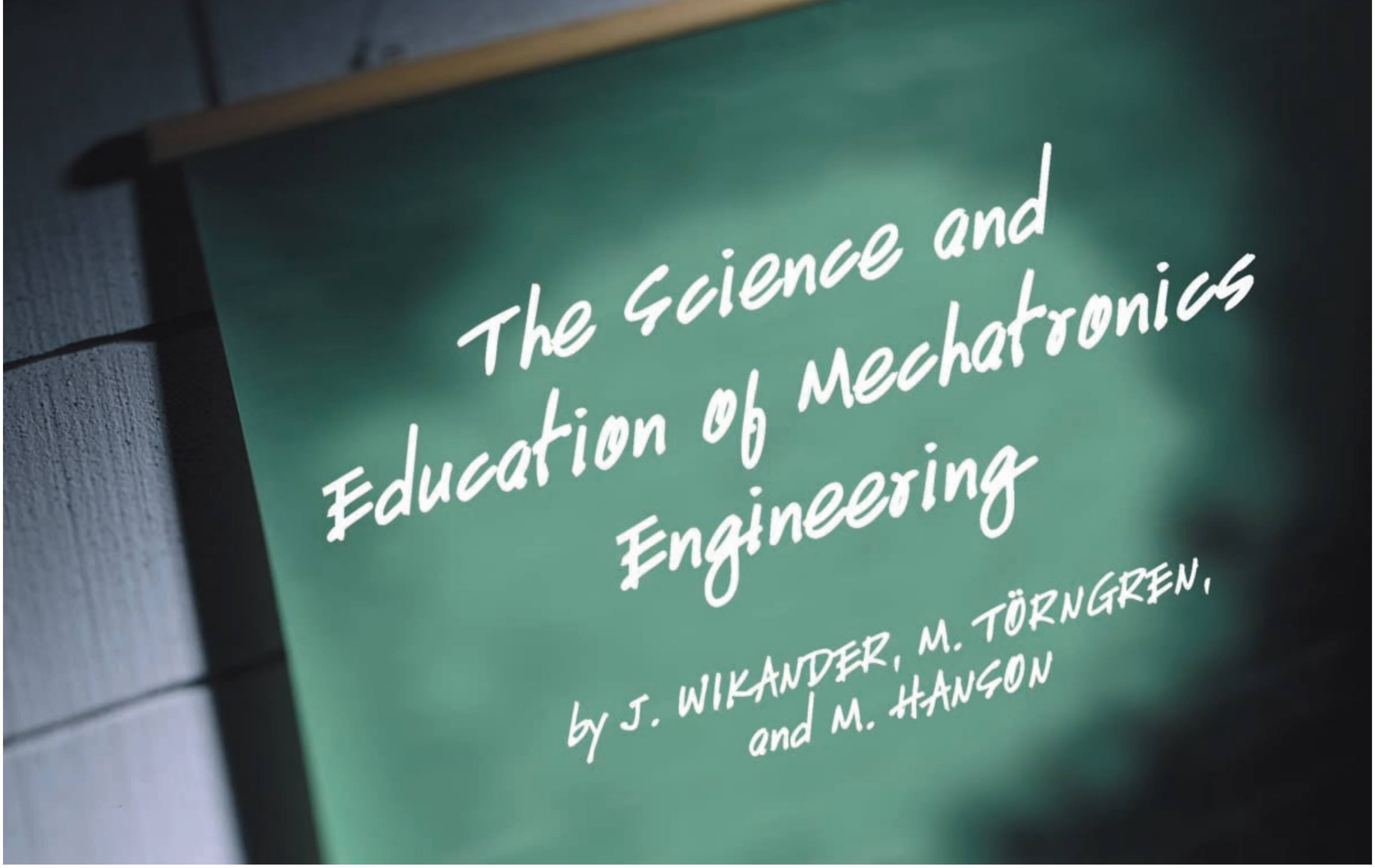

Emphasizing Team Building in a Problem- and Project-Based Curriculum to Meet the Challenges of the Interdisciplinary Nature of this Field

So far there is no common and widely accepted understanding of what mechatronics really is. Many different notions similar to or including mechatronics have been used in various contexts; micromechatronics, optomechatronics, supermechatronics, mecanoinformatics, contromechanics and megatronics are some of these, each coined to put forward a specific aspect or application of mechatronics. Examples of attempts to describe mechatronics include the following.

- Mechatronics encompasses the knowledge and the technologies required for the flexible generation of controlled motions [1].
- Mechatronics is the synergistic combination of mechanical and electrical engineering, computer science, and information technology, which includes control systems as well as numerical methods used to design products with built-in intelligence [2].
- Hewit in [3] states: A precise definition of mechatronics is not possible, nor is it particularly desirable, because the field is new and expanding rapidly; too rigid a definition would be constraining and limiting, and that is precisely what is not wanted at present.

Mechatronics as an interdisciplinary subject tends to attract contributions from all related fields without really putting forward the opportunities and challenges arising specifically due to the interdisciplinary interactions. An example of this is that many mechatronics conferences have been unfocused and thereby have not attracted the most adequate contributions, which definitely exist. This is a disadvantage in that it hampers the development of mechatronics as an engineering science. Scientific publications in mechatronics, to help in making the subject more focused, are still quite rare. One of the earlier publications is *Mechatronics—an International Journal* published by Elsevier Science, first published in 1991.

1070-9932/01/$10.00©2001IEEE

The *IEEE/ASME Transactions on Mechatronics*, a more recent publication, began in 1996.

This article is not just another attempt to describe the research community's definition of the term mechatronics. Rather, we try to get to the heart of multidisciplinary engineering, of which mechatronics is an excellent example, and point out how the integration of disciplines leads to new degrees of freedom in design and corresponding research directions that otherwise would not have been investigated. This is the major contribution achieved by a multidisciplinary approach to engineering science; it leads to a new important research field and at the same time helps to push research in related fields into new fruitful directions.

The Importance of Interdisciplinarity

A general discussion on interdisciplinarity in research, its claimed lack in academia, and its tremendous importance for the next century is presented by Mayer-Krahmer in [4]. Large studies are referred to in which it is concluded that the technology at the beginning of the next century cannot be partitioned according to conventional disciplines and further that important innovations often stem from the interaction of several previously unconnected streams of scientific and technological activity. Further, the full potential of interdisciplinarity includes bridging the gap between real applications, including research, and the scientific disciplines. This is also discussed in [4] and referred to as the dynamics of research and technology; a science-push cycle is followed by a demand-pull cycle, a process that is nonlinear and that requires efficient feedback and interaction.

The Traditional Mechatronics Approach

Engineering of mechatronic systems and products is well established in a substantial number of industrial branches like automotive, manufacturing systems, aircraft control, construction equipment, etc. Such engineering typically applies a subsystem-based approach to mechatronics. By *subsystem based* we refer to a product development strategy by which integrated *systems* are built from technology homogeneous subsystems (mechanics, electronics, control and software). The subsystems are developed in a concurrent manner with an important focus on subsystem interfaces. Once the interfaces are designed, each subsystem is designed in a fairly traditional way. This means that the focus has been on team building to improve communication and multidisciplinary understanding between engineers of different expertise such that the interfaces can be properly defined.

In the subsystem-based approach there is no real demand on development of a certain technology as a result of its closer integration with other technologies; e.g., the close integration of automatic control and computer science. The performance of the mechatronic system is instead merely a result of a sound integration of existing technology. In the existing engineering literature on mechatronics, the subsystem-based approach is likewise totally predominant, but with a too limited coverage of the development process and corresponding team building. Typically, books of this literature devote the first chapter to defining or explaining what mechatronics is, and then the remaining chapters each cover a subject (modeling, sensors, actuators, control, computer hardware, interfaces, communication, etc.) in a traditional but short-form way (see, e.g., [2, 5]). Successful mechatronics engineering can hardly be based solely on such literature.

The subsystem-based approach to mechatronics is still a drastic improvement from the early days when the mechanical engineers first designed the mechanical system.

The subsystem-based approach to mechatronics is still a drastic improvement from the early days when the mechanical engineers first designed the mechanical system, which was then handed over to the control engineers doing a control design. Concurrently, a computer system was designed by the electrical engineers and finally programmers were given the impossible task of designing and implementing a complex controller due to an odd mechanical design, on a too slow computer system.

An Approach for Next-Generation Mechatronic Systems

The advances in digital electronics have enabled the possibility to invent, create, or improve systems that rely on mechanical components to perform their *intended* dynamic actions. The key disciplines to be mastered concurrently and in an integrated manner are mechanical engineering, software engineering, control engineering, and computer engineering. The major paradigm shift enabled by mechatronics is that of shifting the implementation of functionality from mechanical hardware to computer software, but still and most importantly the end-effecting components are mechanical. Notably, we consider software, rather than microelectronics or microprocessors, as being the major new paradigm as it is the software that provides the new and extensive flexibility and freedom in design. However, in many cases the actual software design is implemented in electronic hardware (hardware/software co-design), but in both cases there is a software design level.

Some figures from a world leader in industrial robot design and manufacturing show clearly the trend of moving functionality from mechanics to software. About 12 years ago development engineers were split between mechanics, electronics, and software in roughly estimated shares of 60, 25, and 15, respectively, whereas in 1998 the shares were 30, 15, and 55, respectively. Similar figures have been presented for many embedded control system applications related to mechatronics. Still, there are situations in which passive mechanical components are added in order to cope with certain functional requirements that cannot readily be solved in control software. An example is a passive compliant element added to facilitate the integration of force and position control. The important message in terms of mechatronics here is that an integrated and concurrent design approach is essential to find an "optimal" combination of the best from several engineering domains.

A fundamental aspect of mechatronics is that theories and concepts for mechanical design that have evolved over centuries are about to be replaced by software solutions implemented in embedded microcomputer systems with only a fraction of maturity. This is an aspect of significant importance in many applications, especially those that are safety critical. Related to this, the key feature of software is its versatility and flexibility that, unfortunately, easily leads to complexity problems requiring special attention.

We claim that mechatronics as an engineering science should focus on the interdisciplinary interactions, and based on these identify, formulate, and conduct new research.

The new flexibility in mechatronics design is based upon a number of design choices with respect to how (i.e., with which technology and with which partitioning) certain functionality is implemented. This choice of how functionality is implemented determines the characteristics of transfer functions, data flows, and energy transfer. As the choices regarding partitioning of functionality and corresponding implementation technology are made early in the design chain, there are limited possibilities to take resulting implementation characteristics into account to further improve the design. In order to improve this situation there is an urgent need for theories, models, and tools that facilitate modeling, analysis, synthesis, simulation, and prototyping of multitechnological systems of which mechatronic systems is an important example. Ideally, this will lead to a development strategy involving more of conceptual and top-down design and less of ad-hoc and bottom-up design.

Mechatronics in the Engineering Curriculum

Many new mechatronic programs and courses have been developed over the last decades. The most common pattern is that mechatronic programs and courses have their origin in mechanical engineering. Courses in electrical engineering, computer science, and control engineering have been added or integrated into the mechanical engineering curriculum. One rarely finds programs in electrical engineering or computer science in which mechanical engineering is integrated to give a specialization in mechatronics. Interdisciplinary education is also a question of either integration of new courses within an existing program or tailored development of a completely new program. What is discussed here falls in the former category. The students are first given a solid base in mechanical engineering and then the advantage of receiving specific project and problem-oriented interdisciplinary training on top of this base.

As mechatronics is a holistic, synergistic, and interdisciplinary subject, it is not an easy task to integrate it into an engineering curriculum while still fulfilling the learning objectives. As a background for formulating the aims and objectives for a specific course, one should consider the balance between:

- technology and methodology;
- theoretical science and practical engineering skills;
- working in teams and the assessment of the individual student learning performance.

The following discussion regarding mechatronic course organization should be seen with respect to the fact the engineering curriculum at the (Swedish) Royal Institute of Technology is traditional in the sense that the first two years are very theoretical. To balance a five-year curriculum, the mechatronic courses can concentrate on the problem-solving, practical engineering skills.

Mechatronics is, on one hand, an engineering methodology and an approach to product design. On the other hand, it is an engineering science (further elaborated below) within which new theory and technology is developed. It is natural that specific mechatronic courses focus on the design process rather than the scientific issues. Thus, the students need to concurrently increase their understanding and knowledge of basic engineering topics like control design, digital electronics, computer engineering, and so on.

An easy solution is to use a hybrid model of problem-based learning; that is, to have separate courses in the different subjects, combined with a set of project- and problem-based courses. A typical example of such a course at the Royal Institute of Technology is the course "Microcomputers in Embedded Systems" (4F1131, 5 credits). The aim of this course is to provide the students with a fundamental understanding of embedded systems design, and the method is to use a small hands-on project as the mechanism for teaching. Here, two to three students work in a team responsible for a small development project, from specification to implementation of a prototype.

Our conclusion after 15 years of teaching practice is that mechatronic courses and the curriculum should be problem-based, product-design oriented, and project-team organized. This implies that, to a reasonable extent, the students should work with real industrial product development design cases in which the learning activity is organized in a form similar to professional industrial product development. Further, the teaching of a mechatronic product design process should include as many real situations as possible. In advanced and larger courses this means that concurrent traditional subject-oriented courses should be avoided in order for the students to concentrate instead of having to split the focus and energy between different learning styles. The "Advanced Course in Mechatronic System Design" (4F1160, 12 credits) is organized in this form. Here, the students typically work in larger teams of 15 to 30 people over an extended period of time of around five months. The complexity and technical content of the projects are substantial, and a full development cycle from specification to prototype implementation is performed.

Both courses mentioned above emphasize a balance of the content beween technology and methodology. The students' prime interest is typically focused on the technology, not the methodology. This means that formative assessment methods tend to turn the focus to the technology side. Validation assessments confirm that the methodology is very important.

Yet another mechatronics course named "Real-Time Control and Programming" (4F1140, 4 credits) takes a slightly different approach, in which the focus is on the integration of two separate subjects, namely control design and real-time programming. The course focuses on the interaction between control- and software engineering in order to bridge the gap between dynamic systems modeling and control design and software design and real-time implementation. A substantial part of this course is also devoted to a project involving control design and real-time implementation on a real mechanical process. The methodology part is here closely connected to the use of modern computer-aided engineering tools.

Finally, the university system also has an obligation to develop the students' social skills, the affective part of the education. A very intensive focus on tameability and the assessment of the individual student learning performance is essential. As a result, we have in the problem- and project-oriented mechatronics courses abandoned student grading. Instead we use pass/fail, and in the advanced course we also award an individual written certificate.

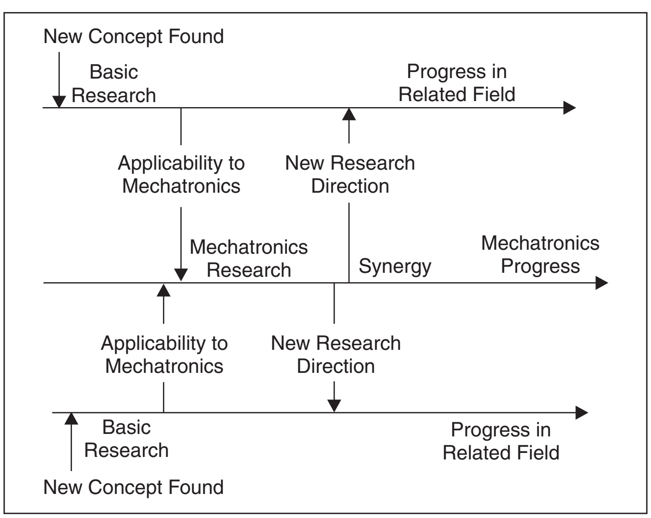

Figure 1. *Integration of disciplines provides synergy and new research directions.*

Mechatronics Research

As opposed to the subsystem-based approach to mechatronics outlined earlier, we claim that *mechatronics as an engineering science* should focus on interdisciplinary interactions, and based on these, identify, formulate, and conduct new research. Consequently, the important role of mechatronics is to bridge the gaps between related engineering sciences. In doing so, limitations due to these gaps become apparent. This has the effect that the application of a mechatronics and thus multidisciplinary perspective will assist in the identification of new research directions in individual disciplines. A fruitful interpretation of this is that mechatronics engineering is about mastering a multitude of disciplines, technologies, and their integration/interaction, whereas mechatronic science is about invention and development of new theories, models, concepts, and tools in response to needs evolving from interacting scientific disciplines. The corresponding scientific scenario is illustrated in Fig. 1.

One rarely finds programs in electrical engineering or computer science in which mechanical engineering is integrated to give a specialization in mechatronics.

Figure 2 below visualizes where the challenges and opportunities arise due to the integration between disciplines and technologies. It is clear that the control algorithms, and thus control engineering, is essential to mechatronic systems. As has been discussed in the introduction, the shift of paradigm is that of moving the implementation of functionality from hardware to software. A typical design scenario is that an early conceptual design phase results in a course partitioning of functionality between mechanics and control software. Based on this, an initial mechanical design is performed. This design provides a basis for modeling of system dynamics that in turn provides input to the control design. The control design is to

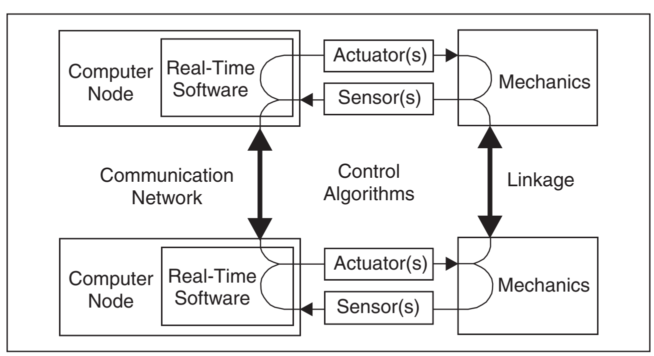

Figure 2. *Interactions within an integrated mechatronic system.*

be transformed into a real-time software design to be implemented on a computer system, which typically is distributed.

The strong integration of technologies (mechanics, control, software, electronics) leads to interdependencies that require special attention in terms of development strategy. A pure sequential development strategy is clearly not suitable, and neither is a traditional concurrent engineering approach. The reason is that the strong interdependencies may have substantial influence on, for example, performance and reliability in a way that cannot be forecasted without a new iterative and highly integrated development strategy. Such a strategy needs to be supported by theories, methods, models, and tools that are yet to be developed.

In the following, there is a discussion of a set of interdisciplinary research directions that have evolved from a multidisciplinary mechatronics approach and for which new research results are available but also for which further research is encouraged.

The ideal mechatronic solution would be that a deficiency of the mechanical system could be cost-effectively compensated by a suitable control engineering solution.

Control of Nonlinear Mechanical Systems

In the process of transferring the functionality from mechanics to control, the idea is typically to simplify the mechanical design and/or improve the control performance. This leads to the need for new control methods requiring also better models of the mechanical system. A typical example of this is the control of mechanical systems exhibiting nonlinear friction. The ideal mechatronic solution would be that a deficiency of the mechanical system could be cost-effectively compensated by a suitable control engineering solution. Due to the interaction of control and mechanical engineering, improved friction models have evolved [6] and model-based control methods [7] have been developed to compensate for friction. An application example employing both the recent advances in friction modeling and in control of a nonlinear system is the servo control of pneumatic cylinders described in [8]. There is a further need for research on control methods (potentially requiring also new models of nonlinear phenomena) for both low-complexity systems like single actuators and complex multidegree of freedom systems. A specific issue is the question of passive or active compliance to handle impact situations and combined force/position control. A related *research challenge* here is the development of control methods for systems involving multiple nonlinearities (friction, backlash, saturation, etc.) in combination with mechanical flexibility.

Real-Time Control Systems Modeling

To design and implement mechatronic systems there is a need for models that support the description and analysis of systems that incorporate among other things control algorithms, software, hardware, interfaces, and mechanics. Co-simulation and links between computer-aided engineering (CAE) tools are a step in this direction; however, such tools do not support conceptual design stages, e.g., performance analysis for evaluation of different structuring and allocation alternatives in distributed control systems. Most theoretical work on the other hand is highly discipline-oriented and has not yet been exploited in an interdisciplinary tool context. Compare, for example, the extensive amount of available theory on real-time scheduling and the relative lack of corresponding tools. Some exceptions in this regard include the Development framework and the Grape project (see [9] for an overview).

Based on this need, the AIDA project [9] was started at the Royal Institute of Technology. The intent is to develop a modeling framework that can describe a control design, the computer hardware and software, and parts of the mechanical structure. To focus the work it was decided that the models should primarily support the analysis of the timing behavior of control applications when implemented in a distributed computer system. The novelty of the work is the multidisciplinary perspective on what the models need to contain, resulting in a combination of existing techniques but also in research to develop modeling features that are missing.

The models, which are intended to be used to complement commercial CAE tools, for each corresponding system describe (control functions, software, etc.) the structure, the internal timing and triggering, and the structural building blocks.

The models in particular allow discipline-specific models to be used while links between these different views are maintained. One example of this is a control system block diagram with its required timing and triggering behavior versus the computer system implementation in terms of processes and scheduling algorithms for processors and communication resources. Another important point is that the internal behavior models exist in two versions; for example, a specification timing and triggering diagram specifies the required timing behavior (e.g., in terms of period sampling with tolerances) while the corresponding implemented timing behavior contains actual values as derived from analysis or experiments. More work remains to verify the models (so far used in two case studies), where the next step is to develop a prototype.

A related *research challenge* concerns how to best combine different modeling concepts in order to support the development of complex industrial systems characterized by:

- a combination of event- and time-triggered parts with respect to both requirements and implementation;
- a combination of discrete-event and discrete-time dynamic systems;

- requirements on functionality, timing, reliability, safety, etc. (i.e., requiring a number of related specifications and attributes to be included in the models).

Time-Varying Control Systems

In the field of automatic control, implementation aspects directly related to computer systems have received little attention. When a control system is implemented in a distributed computer system, it will be subject, to a lesser or greater extent, to [10]

- time-varying feedback delays and sampling periods;
- transient and permanent errors (e.g., lost data in communications).

From a computer engineering and science point of view, on the other hand, there is very little work that directly supports the modeling and analysis of sampled data systems. By studying the implementation of control systems and related problems from an automatic control and computer engineering point of view, it becomes apparent that there is much room for new research dealing, for example, with the following.

- The development of scheduling approaches that directly support the relative timing requirements posed by sampled data control systems [11, 12]. One may note that many useful results exist in the field of real-time scheduling theory but that their application is difficult and ad-hoc because of a mismatch of the underlying timing models and requirements.
- The development of methods for design of control algorithms that can manage time-varying feedback delays and sampling periods (possibly including asynchronous components) [13].
- The development of methods for the derivation of timing tolerances of, for example, feedback delays with respect to performance and dependability requirements [11, 13].
- A theory for systematic handling at the application level of errors due to timing and lost data in sampled data systems. This could be integrated with techniques for error detection and handling used in control engineering.

The expected results would provide designers with new alternatives and thereby the possibility to choose or to combine solutions from the computer and/or control side. A related *research challenge* concerns how and where to best introduce redundancy and diversity (system, software versus hardware) in order to build robust and dependable systems.

Mechatronic Systems Integration

Integration of complex systems, such as computer-controlled mechanical systems, is a difficult problem due to, among other things, the interdisciplinarity. Mechatronic systems integration requires methods from at least modular mechanical design, real-time systems, software architectures, software engineering, control engineering, and electronics design and packaging. A general framework and systematic principles for systems integration can thus only be achieved in a multidisciplinary environment of certain critical size. Research activities in systems integration should be directed toward operational systems in order to be really useful from an industrial viewpoint.

Fundamental research issues related to mechatronic systems integration are as follows.

- Design methodologies (synthesis and analysis) for dependable distributed real-time computer control systems.
- Scalable architectures in terms of control, software, hardware, and mechanics that are based on theoretically

The expected results would provide designers with new alternatives and thereby the possibility to choose or to combine solutions from the computer and/or control side.

sound methods. Such architectures should allow scalable integration of control functions, software, and hardware components.
- Control architectures for extremely complex mechanical systems. A specific issue here is the integration of different control paradigms (e.g., model based, behavior based) in the context of a multitude of control objectives and potentially conflicting controllers.

Conclusions

We have pointed to a number of areas that have benefitted from the interdisciplinary perspective and a focus on interactions between disciplines. We believe that there are many other such areas including, for example, dependability, architectures, and development tools. Within these areas there exist to some extent discipline-oriented methods and techniques, but again by studying the interactions between the disciplines the overlap and lack of capabilities of existing theories and methods can be identified. It is the role of mechatronics as an engineering science to pursue such integrated research activities to extend the current state of the art regarding the design of complex mechatronic systems.

Keywords

Mechatronics, integration, systems, team building, project-based curriculum

References

[1] H. van Brussel, "The mechatronics approach to motion control," in *Proc. Int. Conf. on Motion Control—The Mechatronics Approach*, Atwerp, Belgium, Oct. 1989.

[2] D. Shetty and R.A. Kolk, *Mechatronics System Design*. PWS Publishing Company, 1997.

[3] J.R. Hewitt, "Mechatronics—The contributions of advanced control," in *Proc. 2nd Conf. on Mechatronics and Robotics,* Duisburg/Moers, Germany, Sep. 27-29, 1993.

[4] F. Meyer-Kramer, "Science-based technologies and interdisciplinarity: Challenges for firms and policy," in *Systems of Innovation*, *Technologies, Institutions and Organizations,* C. Edquist (ed.). London: Pinter, 1997, pp. 298-317.

[5] W. Bolton, *Mechatronics—Electronic Control Systems in Mechanical Engineering*. Longman Scientific & Technical, 1995.

[6] B. Eriksson, "Optimal force control to improve hydraulic drives," licentiate thesis, DAMEK Research Group, Dept. of Machine Design, The Royal Institute of Technology, Stockholm, Sweden, Mar. 1996.

[7] C. Canudas de Wit and P. Lischinsky, "Adaptive friction compensation with dynamic friction model," in *Proc. 13th IFAC Triennial World Congress*, San Francisco, CA, 1996.

[8] F. Xiang and J. Wikander, "Nonlinear modeling and identification of pneumatic servo," in *Proc. 17th IASTED Int. Conf. on Modeling, Identification, and Control*, Grindelwald, Switzerland, 1998.

[9] O. Redell, "Modeling of distributed real-tim control systems," licentiate thesis, Department of Machine Design, The Royal Institute of Technology, Stockholm, Sweden, Sep. 1998.

[10] B. Wittenmark, J. Nilsson, M. Törngren, "Timing problems in real-time control systems: Problem formulation," in *Proc. of the American Control Conference*, Seattle, WA, Jun. 1995.

[11] M. Törngren, "Modeling and design of distributed real-time control applications," Ph.D. dissertation, Department of Machine Design, The Royal Institute of Technology, Stockholm, Sweden, Jul. 1995.

[12] S-T. Cheng and A.K. Agrawala, "Scheduling of periodic tasks with relative timing constraints," Dept. of Computer Science, University of Maryland, College Park, Tech. Rep. CS-TR-3392, Dec. 1994.

[13] J. Nilsson, "Real-time control systems with delays," Ph.D. dissertation, Dept. Automatic Control, Lund Institute of Technology, 1998.

Jan Wikander was born in Sweden in 1956. He received his M.Sc. degree in mechanical engineering and Ph.D. degree specialized in mechatronics in 1980 and 1988, respectively, from the Royal Institute of Technology (KTH), Stockholm, Sweden. Since receiving his Ph.D. degree he has been deeply engaged in creating and building up a strong basis for research cooperation between KTH and the Swedish industry. This has led to the foundation of a group of industries involved with mechatronics research activities and the foundation of an Industrial Research Program on Mechatronics. In 1996, he was appointed a professor of mechatronics at KTH (the first full professorship to be established in Sweden) and in 1997 he was appointed head of the Department of Machine Design. His current research interests include mechatronics in general and especially control strategies and control system implementations for the control of complex mechanical systems.

Martin Törngren received his M.Sc. degree in mechanical engineering, specializing in mechatronics, and a Ph.D. degree in 1987 and 1995, respectively, from the Royal Institute of Technology (KTH), Stockholm, Sweden. Since 1989, he has been devoted to mechatronic embedded control systems, in particular distributed real-time systems, where he is now heading the real-time control research group at the Division of Mechatronics at KTH. He has been involved in both building up undergraduate as well as graduate education in the area. He has also been involved in several industrial projects as a scientific/technical advisor. In 1999 he was appointed associate professor and chairman of the Swedish Real-Time Systems Association.

Mats Hanson has been a professor of mechatronics since 1997. He started the mechatronics research, courses, and mechatronics curriculum at KTH's Department of Machine Elements in 1979. In 1993 he was appointed as vice dean and since 1999 has been the dean for the Faculty of Mechanical and Materials Engineering at KTH. He is also the director of the KTH Learning Lab (www.learninglab.kth.se).

Address for Correspondence: J. Wikander, Mechatronics Lab, Department of Machine Design, Royal Institute of Technology, S-100 44, Stockholm, Sweden. E-mail: jan@damek.kth.se.