Title of the thesis: Safe-Guarded Multi-Agent Control for Mechatronic Systems: Implementation Framework and Design Patterns

Summary:

This thesis addresses two issues: (i) developing an implementation framework for Multi-Agent Control Systems (MACS); and (ii) developing a pattern-based safe-guarded MACS design method.

The Multi-Agent Controller Implementation Framework (MACIF), developed by Van Breemen (2001), is selected as the starting point because of its capability to produce MACS for solving complex control problems with two useful features:

- MACS is *hierarchically structured* in terms of a coordinated group of elementary and/or composite controller-agents;
- MACS has an *open architecture* such that controller-agents can be added, modified or removed without redesigning and/or reprogramming the remaining part of the MACS.

However, this framework still had some shortcomings that give room for improvement. An enhancement scheme has been realized: developing a new implementation framework for MACS that inherits and improves the advantages of the MACIF and simultaneously provides the missing features for the MACIF. Through evaluating four possible approaches, that can be applied to develop real-time MACS using concepts and operation mechanisms of the MACIF, the solution using the OROCOS framework (Orocos, 2009a) has been selected for developing a new implementation framework for MACS. After studying the resemblance between the MACIF and the OROCOS framework, a functional combination of the two frameworks has been realized. As a result, we obtain an OROCOS-based Implementation Framework for MACS (OROMACS framework), which supports the development of *multi-threaded MACS* with deterministic *real-time* control behavior, *thread-safe* real-time inter-process communication mechanism, and the capability of handling events. The way of integration used in this combination results in a low coupling between these frameworks. Hence, change of the OROCOS framework will not require much modification of the MACS developed by using the OROMACS framework.

In addition, the port-based polymorphic modeling approach (De Vries, 1994) has been brought to the OROMACS framework. Polymorphic modeling is the division of a subsystem description into a subsystem type and a subsystem specification, and the expression of a subsystem type in terms of one or more designated other types. This approach has been applied to the OROMACS framework in such a way that a controller-agent with a particular Type can be implemented in the form of different Elementary and/or Composite Specifications. This "one Type with multiple Specifications" approach makes the controller-agent and MACS polymorphic. This property, called *polymorphism*, opens the possibility to create libraries of structures for which the detailed implementation is unspecified. As a result, with a sufficiently rich library of multiple specifications, the design and programming of a control system becomes a matter of *configuration and* *composition of controller-agents*. Moreover, the OROMACS framework allows designers to decide beforehand a desired control strategy by selecting suitable coordinators.

Although the OROMACS framework brings with it the improvements, it still faces two shortcomings: (i) the trade-off between the desire to achieve a MACS design with good control performances and a short development time; and (ii) the lack of support for reusability of design results from previous projects into new projects. These shortcomings are tackled by using a combination of the OROMACS framework with the pattern-based design method, which results in *a pattern-based safe-guarded MACS design method*. This design method is demonstrated by means of two case studies.

First, we design a safe-guarded MACS for the DemoLin setup, a simple single-axis electro-mechanical motion system with the dominant compliance in the transmission. The design is required to meet three particular requirements (multi-operation modes, good control performances, and safe-guarded control equiped with capabilities: error detection, error handling, graceful degradation, and error recovery). Based on this design, we have formulated *a generalized safe-guarded control solution for simple mechatronic systems,* i.e. motion systems with one degree-of-freedom (1-DoF).

Next, we design a safe-guarded MACS for the TriPod setup, a complex three-axis electro-mechanical motion system. This design reuses the design results of the DemoLin setup. This reusability is proven through reusing two parts of the MACS design: the operation control and the safe-guarded control. The only thing that remains to be done is to modify application-specific settings (e.g. trajectory, controller parameters, coordinators, etc.). Based on this design, we have formulated *a generalized safe-guarded control solution for complex mechatronic systems*, i.e. motion systems with multiple degrees-of-freedom (n-DoF). The design method makes the design and programming of real-time safe-guarded MACS become *a matter of configuration and composition of the whole design*. This is done through the application of proper design patterns and selection of suitable specifications for controller-agents to quickly build up a complete MACS. As a result, the *short time-to-market objective* with regard to the control system development can be obtained.

This thesis has developed control system design patterns in which the Safe-Guarded Agent is one of main design patterns. This design pattern can flexibly handle faults and particularly fault propagations that may happen in n-DoF motion systems. Specifically, the Safe-Guarded Agent deals with two possibilities of fault propagations: (i) the propagations of influence spheres of faults, i.e. from faults occurring on a single axis to faults involving multiple axes; and (ii) the propagations of criticality levels of faults, i.e. from warning to serious, from serious to dangerous, and from warning to dangerous.