Supervisory Control of Concurrent Systems: A Petri Net Structural Approach – Marian V. Iordache and Panos J. Antsaklis (Boston: Birkhäuser, 2006). *Reviewed by Feng Lin*

The theory of supervisory control of discrete event systems studies various control problems in discrete event systems. Petri nets are a powerful model for complex and concurrent discrete event systems. This book provides a nice presentation of supervisory control of discrete event systems using a Petri net model. It covers a wide range of topics on supervisory control in the framework of Petri nets.

To put this book in perspective, let us briefly review the history and theory of supervisory control of discrete event systems. The supervisory control theory of discrete event systems was born in the University of Toronto in the 1980s. The first paper on the theory was written by Ramadge and Wonham. The full version of the paper was published in 1987 [6], and the conference version was published early in 1984. In the paper, Ramadge and Wonham proposed an automaton model for discrete event systems and a supervisory control mechanism, which is enablement and disablement. The event set is divided into controllable events (those that can be disabled) and uncontrollable events (those that cannot be disabled). Controllability was introduced to capture the necessary and sufficient condition for the existence of a supervisor. Controllability is defined for a legal or desired language K with respect to a plant language L(G) which corresponds to all possible trajectories of the uncontrolled system. Controllability of K with respect to L(G) essentially requires that all events to be disabled by the supervisor must be controllable. All events are assumed to be observable in [6]. To consider the case of supervisory control under partial observation, where events are divided into observable and unobservable events, Lin and Wonham introduced observability. The paper on observability [3] was published in 1988 and the corresponding technical report in 1986. In [3], the observation of a supervisor is described by the projection operator P that erases all unobservable events in a trajectory or string. In other words, if a string occurs in a system, the supervisor will see its projection. Observability is then defined for a legal language K with respect to the plant language L(G) and the projection P. It requires that if two strings look the same to the supervisor (having the same projection), then the control actions needed by the supervisor after these two strings must be consistent (that is, no event will be enabled after one string but disabled after the other). In other words, the supervisor must have sufficient information to make a control decision on each controllable event. It was proved in [3] that a necessary and sufficient condition for the existence of a partial observation supervisor is controllability and observability. After the establishment of controllability and observability, the next topic investigated in supervisory control was decentralized supervision. Two cases were considered: (1) specifications are given locally, and (2) specifications are given globally. Local specifications were discussed by Lin and Wonham in [4]. A property called normality was introduced in [4] to capture the existence condition for local or decentralized supervisors. A legal language K is normal with respect to the plant language L(G) and the projection P if a string belongs to K if and only if it belongs to L(G) and its projection belongs to the projection of K. Global specifications (which are more general than local specifications) were investigated by Rudie and Wonham in [7]. Co-observability was defined for global specifications as follows. A legal language K is co-observable with respect to the plant language L(G) and the sets of locally controllable events and locally observable events if the following conditions are satisfied: (i) if an event is controllable by only one supervisor, then that supervisor must have sufficient information to make a control decision on that event; (ii) if an event is controllable by several supervisors, then at least one supervisor must have sufficient information to make control decision on that event. Rudie and Wonham showed in [4] that co-observability, together with controllability, is necessary and sufficient for the existence of decentralized supervisors that achieve a global specification. Decentralized supervision and supervision under partial observation are independently studied in [2]. The book [1] by Cassandras and Lafortune gives a nice presentation of supervisory control.

The above definitions of controllability, observability, normality, and co-observability are presented in terms of languages. The reason for using languages is that it provides a "model independent" treatment. In other words, they are independent of how legal languages and plant languages are modeled. They can be regular or irregular, specified by finite automata or other methods. Theoretically, this is elegant. However, for practical applications, some finite representations of languages are needed. This is usually done by using either finite automata or Petri nets. The main advantage of using automata is that it is simple and general. For example, if we need to design a partial observation supervisor, one algorithm can be used for all automata (this is not the case for Petri nets). However, there are two important disadvantages of using automata: (1) For some systems, the number of states in the automata may be very large, or infinite. (2) The automaton model cannot handle concurrency efficiently: It assumes that events will not occur at exactly the same time. If two events must be synchronized to occur at exactly the same time, then it is better to define them as one event. This is sometimes inconvenient.

Those two disadvantages led to the use of Petri nets, which effectively overcome the disadvantages of automata. Petri nets can be used to model systems with many or infinite states and they can be used to model concurrent systems. There are different types of Petri nets and this book focuses on ordinary Petri nets and their generalizations using weight functions and labeling functions. It allows separate labels for observation and control (double-labeled Petri nets). A nice introduction to Petri net models is presented in Chapter 2.

In terms of control specifications, in automaton models, specifications are almost always given by languages and their associated automata. However, in Petri net models, specifications can be given in different ways: (1) specifications using linear constraints, (2) specifications using generalized linear constraints, and (3) language specifications. Chapters 3, 4, and 5 present supervisory controls for the above specifications. Supervision with full observation, supervision under partial observation, and decentralized supervision are all discussed. They are summarized in the following table.

Specifications	Full Observation	Partial Observation	Decentralized Supervision
specifications using linear constraints	Section 3.2.1	Section 3.2.2	Sections 5.4-5.7
Specifications using generalized linear constraints	Section 4.2.3	Sections 4.2.5 and 4.2.7	
Language specifications	Section 4.3	Section 4.3	

The issues of deadlock and liveness are important in supervisory control. In the automaton model, they are studied under the requirement of nonblocking; that is, the requirement that the system can always reach a given set of marked states. To ensure a controlled system is deadlock-free, we can designate a state as marked if there exists at least one event that is possible at the state. To ensure that a controlled system is live with respect to a given set of events, we can designate a state as marked if there exists at least one event from the given set that is possible at the state. Synthesizing nonblocking supervisors is more complex using the automaton model. This is also true in the Petri net model. Chapters 6-8 discuss the issues of deadlock and liveness in supervisory control of Petri nets. This is done not by examining all possible markings from a given initial marking, but rather by investigating structural properties of Petri nets. The structural approach avoids the need to explore a large (and possibly infinite) number of states. This approach is well developed and presented in the book.

The last two chapters of the book deal with hybrid systems. Hybrid systems are a natural extension of discrete event systems when dynamics of continuous variables must be considered. Several approaches to control of hybrid systems have been proposed with different models and objectives. This book uses the Petri net model and the control is exercised at two levels. The low level controls continuous variables of the system. The lower level is abstracted to the upper level as a discrete event system. Supervisory control of discrete event systems is then used at the upper level. Authors give details on how this can be done in Chapters 9 and 10.

The key feature of the book is that it presents the recent research in supervisory control of Petri nets in a comprehensive manner. The book is well written and self-contained. It provides the necessary background materials on discrete event systems and Petri nets. The book will be very useful to researchers in the field of supervisory control. It can also be used as a textbook for a graduate course on supervisory control and Petri nets. This book is complementary to another book [5] on supervisory control of discrete event systems using Petri nets from the same group (Moody and Antsaklis).

[1] C. G. Cassandras and S. Lafortune. *Introduction to Discrete-Event Systems*. Kluwer Academic Publishers, 1999.

[2] R. Cieslak, C. Desclaux, A. S. Fawaz, and P. Varaiya. Supervisory control of discrete-event processes with partial observations. *IEEE Transactions on Automatic Control*, 33(3): 249–260, 1988.

[3] F. Lin and W. M. Wonham. On observability of discrete-event systems. Information Sciences, 44:173–198, 1988.

[4] F. Lin and W. M. Wonham, "Decentralized supervisory control of discrete-event systems," Information Sciences, 44(3), pp. 199-224, 1988.

[5] J. O. Moody and P. J. Antsaklis, *Supervisory Control of Discrete Event Systems Using Petri Nets*, Kluwer Academic Publishers, 1998

[6] P. J. Ramadge and W. M. Wonham. Supervisory control of a class of discrete event processes. *SIAM Journal on Control and Optimization*, 25(1): 206–230, 1987.

[7] K. Rudie and W. M. Wonham. Think globally, act locally: Decentralized supervisory control. *IEEE Transactions on Automatic Control*, 37(11): 1692–1708, 1992.