# 1

# Introduction to networked control systems

J. Lunze and L. Grüne

*This chapter gives an informal introduction to networked systems, explains the main problems and gives a survey of the new results reported in this monograph.*

**Chapter contents**

## 1.1 What are networked control systems?

### 1.1.1 Motivation and examples

The control of dynamic systems necessitates the communication of sensor information towards a controller and of information about the control input from the controller towards the actuators. The implementation of this communication by means of digital networks has been common practice for more than 20 years. The great current interest in the field of networked control systems results from the fact that a large variety of digital networks are becoming available everywhere and can be used for the implementation of feedback loops without additional installation cost. Wireless connections facilitate the extension of the application area of automatic control towards mobile objects, because measured variables and control variables can now be transmitted to the controller from nearly every place of a technological plant.

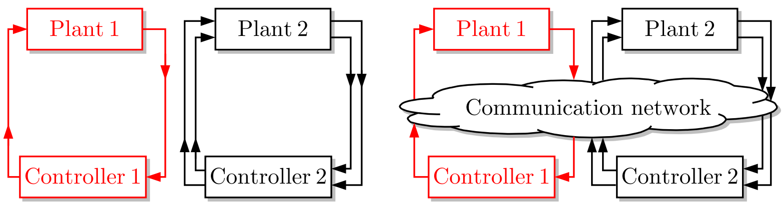


**Fig. 1.1.** Subject of classical control theory (left) and of the theory of digitally networked control systems (right)

A first rough comparison of the new theory of digitally networked systems and the traditional control theory is depicted in Fig. 1.1, which shows, on the left-hand side, two independent control loops. These loops may be implemented by means of a communication network, but if the network is dedicated to these control loops, the overall system behavior does not depend upon this network and both control loops can be considered as independent entities. They can be analyzed and designed by means of multivariable control theory.

If, however, the network does impose constraints on the information transmission, it influences the behavior of the overall system. Therefore, the network has to be dealt with as a separate component in the closed-loop system as shown on the right-hand side of Fig. 1.1. This new situation has severe consequences. As the figure shows, the two loops, which are separated in the classical sense, influence one another in the networked situation. The influence may deteriorate the performance of both loops, because the network introduces additional delays into the feedback loops, but the coupling may also be used to improve the behavior if the flexibility of the communication network is used to couple both control loops in favor of a quicker disturbance attenuation or set-point following.

The new control structures cannot be handled with traditional methods for two reasons. First, the event-driven mode of operation of digital networked systems violates the precondition of discrete-time control theory that data are processed and transmitted with a constant sample rate. Second, the structure of the network is usually not pre-defined but the subject of control design. The data links may even adjust themselves to the needs and the technical constraints. Research on digitally networked control systems deals with novel questions concerning the modeling of dynamic systems, the analysis of feedback systems, and the design of distributed controllers.

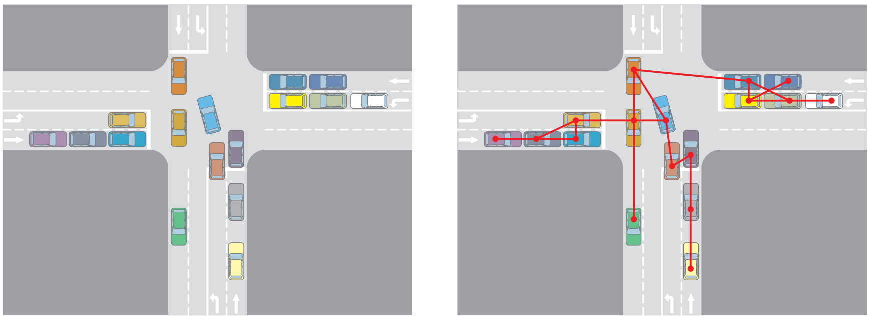

**Fig. 1.2.** Future crossroad management with wireless vehicle-to-vehicle communication

**Examples for networked control systems.** In the application scenarios considered in the theory of networked control systems, the assumption of classical control theory that information can be transmitted within control loops instantaneously, reliably and with sufficient precision is no longer satisfied as the following examples show:

- **Telerobotics:** Long communication links between the (human) controller and the robots to be controlled may bring about time delays and partial loss of communication.

- **Traffic control:** If floating-car data are used for traffic supervision, state estimation and control has to use wireless components and moving sensors which do not work in accordance with a centralized sampling scheme, but they are driven by the traffic circumstances and by the tuning range of the wireless communication.

- **Future crossroad management:** Advanced communication and control systems will make it possible to bring vehicles safely over a crossroad by using vehicle-to-vehicle communication. The vehicles decide autonomously who goes next (Fig. 1.2). Only those vehicles are connected that are in the surrounding of the crossroad. The number of vehicles involved depends upon the current traffic and the communication effort changes during the flow of the traffic.

- **Smart grid:** Energy distribution networks are gaining more intelligence by the introduction of smart components at all layers from the autonomous switches towards smart measuring components [28, 112, 262]. Likewise, smart cities should be created where economy, mobility, environment, living conditions and governance are enhanced by information and communication technologies in order to reach a sustainable economic development [184].

- **Ambient intelligence:** If people are supervised by wireless sensors distributed network structures are used that have no coordinator [113, 318].

**Properties of digital communication networks.** These examples show that the kind of networks considered in the field of "networked control systems" have the following properties:

- The networks are open and inhomogeneous with changing topology and nodes.
- The networks behave non-deterministically in dependence upon the number of nodes, the used links and the load.
- The networks provide a flexible communication structure, even for mobile objects, that can be used whenever necessary.

Hence, the behavior of the overall systems that include the physical systems to be controlled, the controllers and the communication network is severely influenced by the communication network. The theory of networked control systems should elaborate new methods for dealing with the phenomena appearing in such systems. There are, in summary, two challenges: the communication constraints imposed by the network and the flexibility to be used for solving novel control problems.

**Communication constraints.** Traditional control theory uses the assumption that all communication links that are necessary for solving a control problem at hand can be implemented and used with the required quality and reliability. Moreover, the communication is assumed to be instantaneous and lossless. These assumptions are not satisfied by the networks described above. Hence, a control theory of networked systems has to take into account communication constraints with respect to timing, information loss, variable communication topology etc.

**Flexibility of communication.** With the new communication technology, information can be transferred among arbitrary nodes within control loops and among different control systems. In particular, the communication topology can be adapted to the current needs. In this new situation, the value of information for solving control tasks gains a particular importance and has to be the subject of scientific research. Whereas traditional control methods start from a fixed communication topology, the flexibility brought about by modern communication networks makes it necessary to elaborate criteria for deciding which value a certain piece of information has for the solution of a control task. These criteria show under what conditions and with which quality information links within control loops and among control systems should be invoked and used.

Due to these properties, the communication network is drawn in the figures of this monograph as clouds without a well-defined boundary rather than as a block with well-known properties.

The examples above have shown that the communication constraints of the networks used will not be overcome simply by advances of technology. Networks should be realized by cheap components, which have restricted energy and computing power and, hence, provide information with restricted sampling rate and accuracy. Moving objects impose constraints on the tuning area of the components and, thus, make the communication process nondeterministic. Temporal loss of communication links cannot be avoided. To repeat sending data packets until they reach the receiver introduces time delay, which may exceed tolerable bounds. Multi-hop data transmissions are associated with large time delays because every communication step necessitates the synchronization of all senders and receivers involved.

**Networked control systems.** Having these properties of the communication network in mind, the term *networked control systems* can be defined as follows:

> Networked control systems are closed-loop systems that have to be considered as networked systems.

The term "closed-loop system" includes the fact that controllers are connected in feedforward and feedback structures with physical systems, where both components together determine the behavior of the overall system. The aspect of the "networked system" emphasizes the fact that the overall system consists of several components that are connected physically or by a digital communication network and have to be analyzed by considering the network structure. Hence, networked control systems are typically represented as an interconnection of nodes with dynamic properties.

An important aspect of using graph-theoretic representation forms lies in the fact that the overall system may change its structure during operation. Nodes may disappear or new nodes may join the network. So, the dynamics of the network do not only lie in the properties of the nodes but also in the properties of the overall system.

This definition of networked control systems leads to an answer to the question, under what conditions a feedback system has to be considered as a networked system. The communication network has to be considered as an explicit component in the control loop,

- if the task is to investigate which communication links are necessary (i.e. the communication topology is a design goal),
- if communication constraints (e.g. scheduling, time delay, packet loss) have to be taken into account,
- if control objectives have to be satisfied that necessitate coordinated actions of subsystems (e.g. multi-agent systems).

### 1.1.2 Cyber-physical systems

The close connection of the physical world with embedded (control) systems coupled through a global network is currently investigated under the buzz word of cyber-physical systems. Cyber-physical systems appear as an extension of embedded computer systems by global networks. One expects that the bandwidth of modern digital communication networks will double every two years, which leads to the question which information links are necessary. From a global viewpoint, all computer networks are expected to be coupled in the future to form the upcoming *Internet of things and services* [206, 218, 284, 298].

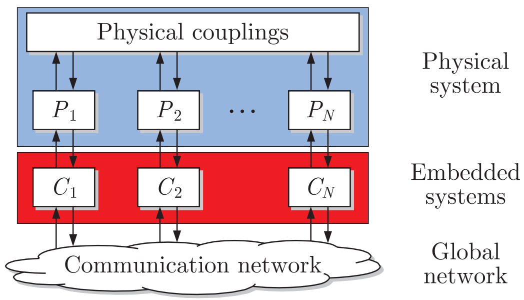


**Fig. 1.3.** Networked control system as cyber-physical system

The networked control systems investigated in this book are such cyber-physical systems, as Fig. 1.3 shows. The embedded systems $C_i$ are used to implement local controllers of the subsystems $P_i$, $(i = 1, 2, ..., N)$. The network allows a direct information exchange among the local controllers. The novel situation provided by modern digital networks is characterized by the fact that this communication can be used extensively whenever communicated

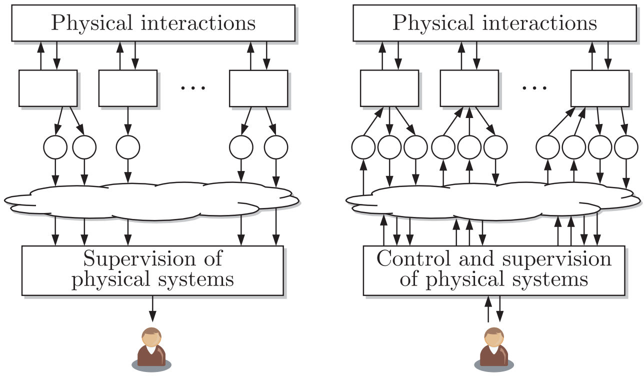


**Fig. 1.4.** Network of information (left) and network of action (right)

information can be utilized to improve the overall system performance. The theory of networked control systems emphasises the interaction of the cyber-system (controller, network) and the physical system (plant).

The future development of networking technology can also be seen as an evolution of the current *network of information* towards a future *network of actions*. In the network of information, the information flow goes mainly from the physical system towards the supervision system and the human operators (Fig. 1.4 (left)). The new aspect of the network of actions is the fact that the subsystems can communicate among each other and with the human operator and that this communication goes in closed loops (Fig. 1.4 (right)).

**Control, communication and computation.** A control theory of networked systems has to deal with the new challenges that originate from this new network structure and has to develop new ideas and methods for networked control systems.

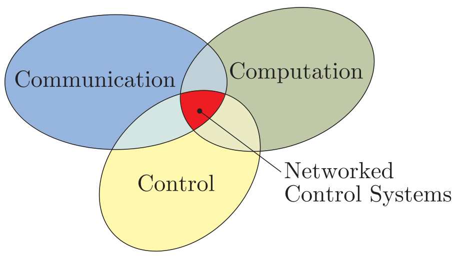


**Fig. 1.5.** Integration of control with communication and computation

For the scientific development, the network of actions mean that the fields of communication, computation and control have to be developed in an integrated way (Fig. 1.5). Currently, these three aspects are considered separately.

For the control loops, one can assume that communication and computation are quick enough not to bring about substantial time delays into the closed-loop systems. In contrast, in networked control systems, communication constraints and scheduling problems have to be taken into account. Hence, the models used to represent networked control systems combine aspects of the physical description of the plant with models of the communication network and of the computation schedule. Section 1.2 will summarize different classes of models together with the corresponding analysis and design methods to show how the main properties of the communication network extend the dynamic models of the plant or the closed-loop system, respectively.

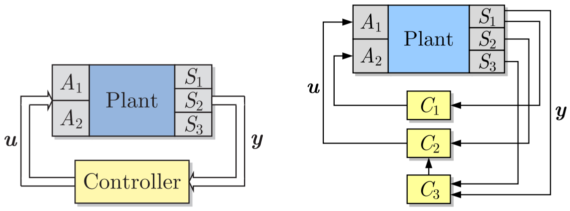


**Fig. 1.6.** Multivariable system vs. networked control system

Methods and tools have to be investigated to integrate the engineering view on physical systems with the computer science view on computation and communication. Consequently, new technologies have to include

- heterogeneous modeling with various model abstractions, and
- analysis and design methods for heterogeneous systems.

This combination of communication and control has direct consequences for the development of advanced control systems in industry. As long as the communication can be assumed to be ideal, the design of control systems can be separated from the design and implementation of the controllers and the communication network. Separate design decisions can be made in both fields and, if brought together, the communication network and the controllers will satisfy the overall system goals.

However, if communication and control have to be considered together, design decisions in one field have direct consequences on design decisions in the other field. Trade-offs have to be made from a global viewpoint and an inter-disciplinary cooperation between experts in communication and control is necessary.

### 1.1.3 Structures of networked control systems

For control engineering applications, the trends outlined above lead to the question what are the main challenges for control theory if a systematic analysis and design of closed-loop systems should be facilitated. Figure 1.6 compares conventional multi-input multi-output feedback loops with networked control systems. Traditionally, a control loop is considered to consist of one plant and one controller, where all sensor information is lumped together to form the output vector $\boldsymbol{y}(t)$ and all the control inputs are considered as a single vector $\boldsymbol{u}(t)$. The individual handling of the sensor information $y_i$, controllers $C_i$ and inputs $u_i$ to the actuators results in event-driven operating modes, distributed controllers and goal-dependent communication regimes (Table 1.1).

| **Traditional control theory** | **Theory of networked control systems** |
|---|---|
| Time-driven, sampled-data systems | Event-driven, asynchronous systems |
| Centralized, coordinated controllers | Decentralized, distributed controllers |
| Fixed communication structure | Goal-dependent communication structure |

**Table 1.1.** Comparison of traditional control theory and the theory of networked systems

The main characteristics of networked control systems can also be seen in novel structures that are characterized by

- distributed sensing, computation, and actuation,
- decentralized and distributed control structures with intelligent nodes, and
- resource constraints and imperfect communication.

## Bibliographical notes

During the last ten years, several survey articles, special issues of journals and some monographs have appeared, which can be used as starting points to study the theory of networked control systems in more detail:

- The survey articles [8, 178] give an answer to the question what is new in networked control systems.
- Special issues of journals collect new results in this field: [7, 159, 182, 189]
- Monographs deal with special aspects of networked systems. [254] concentrate on estimation problems.
- The material developed for PhD schools and workshops give a broad and didactical introduction to the field, e.g. [26]

## 1.2 Theory of networked control systems

### 1.2.1 Overview

This section describes the main problems to be tackled by the theory of digitally networked dynamic systems and gives a survey of the models and solutions described in this monograph. The methods are ordered according to the model of the networked system used.

The main question that a theory of networked control systems has to answer asks

> Which information is necessary to solve a given control task?

This question has two important aspects:

- **Topology:** Which information links are necessary?
- **Quality:** How often and how quickly has information to be communicated and which accuracy of information is necessary?

In order to answer these questions, new modeling paradigms for networked control systems have to be elaborated, the main idea of which is to incorporate the important properties of the communication network into the model of the plant. The models combine methods that have been developed in the past in control theory, computer science, and communication theory.

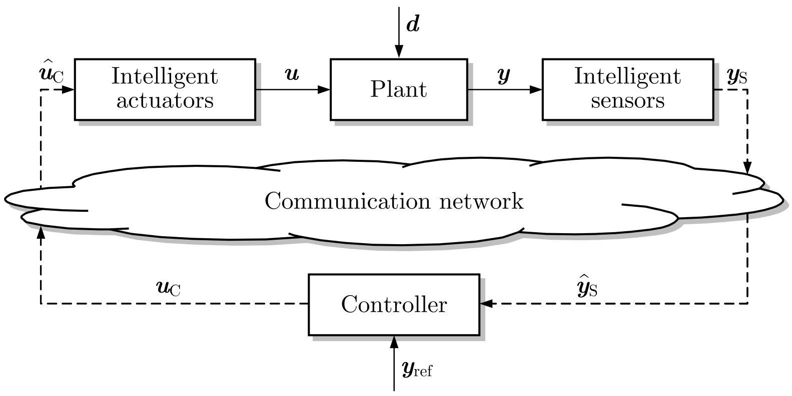


**Fig. 1.7.** Basic networked control system

The principal ideas used for modeling networked systems can be explained for the single control loop shown in Fig. 1.7. In order to represent effects of the communication network on the data transfer from the sensors towards the controller as well as from the controller to the actuators, the sensor signal $\boldsymbol{y}_{\mathrm{S}}(t)$

is distinguished from the controller input $\hat{\boldsymbol{y}}_{\mathrm{S}}(t)$ and the controller output $\boldsymbol{u}_{\mathrm{C}}(t)$ from the actuator input $\hat{\boldsymbol{u}}_{\mathrm{C}}(t)$. One can say that the signals $\hat{\boldsymbol{y}}_{\mathrm{S}}(t)$ and $\hat{\boldsymbol{u}}_{\mathrm{C}}(t)$ are the "network versions" of the original signals $\boldsymbol{y}_{\mathrm{S}}(t)$ or $\boldsymbol{u}_{\mathrm{C}}(t)$, respectively, sent over the corresponding communication channels.

The notions of intelligent sensors and intelligent actuators used in the figure should emphasize that these components may include computing elements, for which generally no restrictions concerning the computing power and the memory are imposed. The solid arrows show continuous-time signals, whereas the dashed arrows denote information links that are only used at certain transmission times $t_k$, $(k = 0, 1, ...)$.

The methods to incorporate the communication network into the model of the plant or of the closed-loop system can be classified as follows:

- **Control-theoretic approaches:** From a control-theoretic viewpoint, data transfer over the network introduces time delays, which have to be considered when analyzing the overall system and designing the controller. Furthermore, information feedback is only necessary if disturbances have to be attenuated or model uncertainties to be tolerated. Control-theoretic approaches should answer the question at which time instances communication is necessary.

- **Information-theoretic approaches:** The data transfer is restricted to a certain bit-rate, which means that the signals $\boldsymbol{y}_{\mathrm{S}}(t)$ and $\boldsymbol{u}_{\mathrm{C}}(t)$ sent over the network is restricted to a certain bit length. The question is whether this information flow is sufficient to solve the control task given for a certain application.

- **Network-theoretic approaches:** The overall system is considered as a graph, the nodes of which are dynamic elements and the edges show where information among the nodes may be exchanged. The communication topology represented by the graph may change over time if data packets get lost or if communication links are created or disappear because it cannot be used any longer. This approach is useful if the question should be answered which information links are necessary to solve a given control task.

As a result, heterogeneous models are used to represent the control system together with the communication network. The following sections give a survey of these models and the problems that can be solved by using these models. In applications, several of these approaches have to be combined to deal with all practical constraints of networked systems.

### 1.2.2 Control with information rate constraints

The communication network may impose strong restrictions concerning the data rate, particularly if wireless networks are used. To include these restrictions into the model of the control loop, the sensor nodes are equipped with an encoder that, from a control-theoretic viewpoint, introduce a signal space partition. The encoder operation is denoted by $[\cdot]$

$$[\cdot] : \mathbb{R}^n \to \{1, 2, ..., 2^{\bar{r}}\}$$

and the symbol $[\boldsymbol{x}(t_k)]$ sent at time $t_k$ is the number of the partition in which the state $\boldsymbol{x}(t_k)$ lies at that time $t_k$. $\bar{r}$ denotes the maximum bit length of the data sent over the network.

As shown in Fig. 1.8, it is usually assumed that the state $\boldsymbol{x}$ is measurable and its encoded version $[\boldsymbol{x}]$ is transmitted to the intelligent actuator, which includes a decoder together with a state observer and the controller. The fundamental question to be answered is how many bits per second have to be transmitted over the network in order to solve a given control task.

Investigations along this line have led to *data-rate theorems*, which give lower bounds on the bit rate necessary to stabilize an unstable plant. For the standard situation, where the controller should stabilize a linear discrete-time system

$$\Sigma : \begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) \end{cases}$$

the minimum number of bits to be sent at every sampling instant is bounded by

$$\bar{r} > \sum_{|\lambda_i|>1} \log_2 |\lambda_i|, \tag{1.1}$$

where the sum concerns all unstable eigenvalues $\lambda_i$ of the matrix $\boldsymbol{A}$. It is interesting to see that the amount of data to be sent over the network directly depends upon the unstable eigenvalues of the plant.

First papers on this subject include [14, 269, 358, 389, 390] and a survey is given in [271]. The main idea to prove the inequality (1.1) is to design a state observer that reconstructs the smallest set of states in which the current plant

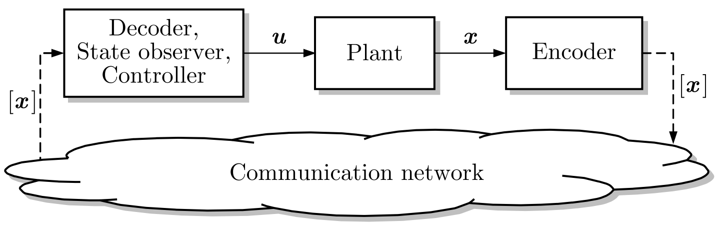


**Fig. 1.8.** Feedback loop with information rate constraints

state is at time $k$. This state set expands from one sampling time to the next time due to the unstable modes of the system but it is reduced by the new quantized measurement information. The inequality (1.1) ensures that the expansion is smaller than the reduction and, hence, after a sufficient number of sampling instants the state $\boldsymbol{x}(k)$ is known to the observer with sufficient precision. Then a state feedback can be applied to stabilize the plant.

Investigations along this line are reported in the following parts of this book:

- **Minimal bit rates and entropy for control tasks:** In Section 2.3, results on the minimal transition data rate for making a subset of the state space invariant are described.

- **Quantized event-based control:** Section 5.4 develops an event-based control strategy where the information sent from the sensor towards the controller is restricted to the quantized state. This information is communicated only whenever the state changes from one quantization region to another one. Hence, the feedback information used in the networked overall system is reduced with respect to both the time instants at which it is sent and the contents.

### 1.2.3 Control subject to networked-induced time delays

If the communication network introduces temporarily or permanently severe time delays in the feedback path from the sensors via the controller towards the actuators, the network can be modeled as a time-delay system. In Fig. 1.9 the overall time delay on the feedback path is denoted by $\tau$. Hence, the behavior of the networked system on the left-hand side of the figure can be represented as the time-delay system shown on the right-hand side, where $\tau = \tau_{\mathrm{y}} + \tau_{\mathrm{u}}$ holds.

In the structure shown, the network version $\hat{u}(t)$ of the control input $u(t)$ generated by the controller is delayed so that the overall system can be

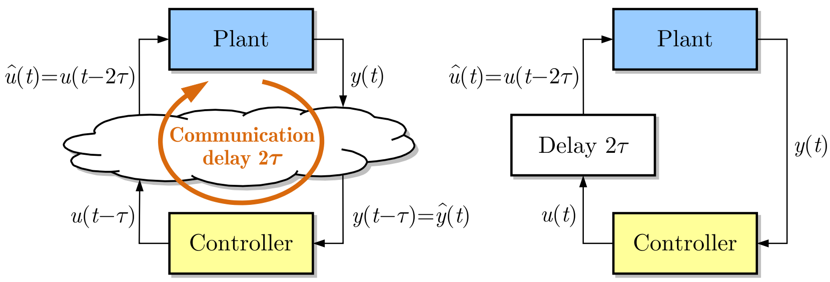


**Fig. 1.9.** Networked control system with delayed communication

described by

$$\text{Plant:} \begin{cases} \dot{\boldsymbol{x}}_{\rm p}(t) = \boldsymbol{A}_{\rm p}\boldsymbol{x}_{\rm p}(t) + \boldsymbol{B}_{\rm p}\boldsymbol{u}(t-\tau) + \boldsymbol{E}\boldsymbol{d}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}_{\rm p}\boldsymbol{x}_{\rm p}(t) \end{cases}$$

$$\text{Controller:} \begin{cases} \dot{\boldsymbol{x}}_{\rm c}(t) = \boldsymbol{A}_{\rm c}\boldsymbol{x}_{\rm c}(t) + \boldsymbol{B}_{\rm c}\boldsymbol{y}(t) \\ \boldsymbol{u}(t) = \boldsymbol{C}_{\rm c}\boldsymbol{x}_{\rm c}(t) \end{cases}$$

The well-known theory on time-delay systems can be applied for analyzing such feedback systems. However, in networked systems, the delay $\tau$ is usually unknown. This fact has motivated new investigations on time-delay systems with stochastic delays, on systems with delays that are described by upper bounds, and on systems for which the delay can be determined online. Overviews can be found, for example, in [236, 405].

**Control with asynchronous communication and computation.** In a more detailed analysis, event-triggered components of the control loop work in different time schemes. Figure 1.10 shows the situation, in which the sensors work time-triggered at every clock time, whereas the controller and the actuator computes or applies the new input signal in an event-driven way as soon as it has received new information. Under the standard assumption of discrete-time control theory, all components work synchronously at the same time (Fig. 1.10 (left)). This assumption is satisfied as long as the time delays are small in comparison to the main time constants of the control loop.

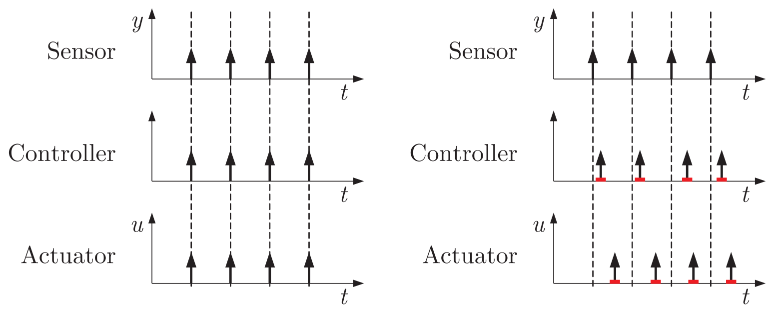


**Fig. 1.10.** Synchronous (left) and asynchronous behavior (right) of the components of a control loop

If the delays are larger, the missing synchrony has to be represented in the model and leads to an interesting and important extension of discrete-time models. Assume, for simplicity, that there is only a delay between the sensor and the controller in the control loop, but the actuator receives the output of the controller without delay. Then the actuator has to implement the input

$$\boldsymbol{u}(t) = \begin{cases} \boldsymbol{u}(k-1) & \text{for } kT \le t < kT + \tau \\ \boldsymbol{u}(k) & \text{for } kT+\tau \le t < (k+1)T, \end{cases}$$

where $T$ denotes the sampling time. In this formula, $\boldsymbol{u}(t)$ denotes the continuous-time signal and $\boldsymbol{u}(k)$ the $k$-th value of the discrete-time version of the input $\boldsymbol{u}$. Instead of getting the usual sampled-data plant model

$$\begin{aligned} \boldsymbol{x}_{\mathrm{p}}(k+1) &= \boldsymbol{A}_{\mathrm{d}}\boldsymbol{x}_{\mathrm{p}}(k) + \boldsymbol{B}_{\mathrm{d}}\boldsymbol{u}(k) \\ \boldsymbol{y}(k) &= \boldsymbol{C}_{\mathrm{p}}\boldsymbol{x}_{\mathrm{p}}(k) \end{aligned}$$

with

$$\boldsymbol{A}_{\mathrm{d}} = \mathrm{e}^{\boldsymbol{A}_{\mathrm{p}}T}, \quad \boldsymbol{B}_{\mathrm{d}} = \int_0^T \mathrm{e}^{\boldsymbol{A}_{\mathrm{p}}\tau}\,\mathrm{d}\tau \boldsymbol{B}_{\mathrm{p}}$$

one now gets the extended plant model

$$\Sigma_{\mathrm{d}} : \begin{cases} \boldsymbol{x}_{\mathrm{p}}(k+1) = \boldsymbol{A}_{\mathrm{d}}\boldsymbol{x}_{\mathrm{p}}(k) + \boldsymbol{B}_{\mathrm{d1}}\boldsymbol{u}(k) + \boldsymbol{B}_{\mathrm{d2}}\boldsymbol{u}(k-1) \\ \qquad\quad \boldsymbol{y}(k) = \boldsymbol{C}_{\mathrm{p}}\boldsymbol{x}_{\mathrm{p}}(k), \end{cases}$$

which has two input terms with the matrices

$$\boldsymbol{B}_{\mathrm{d1}} = \int_\tau^T \mathrm{e}^{\boldsymbol{A}_{\mathrm{p}}\tau}\,\mathrm{d}\tau \boldsymbol{B}_{\mathrm{p}} \quad \text{and} \quad \boldsymbol{B}_{\mathrm{d2}} = \int_0^\tau \mathrm{e}^{\boldsymbol{A}_{\mathrm{p}}\tau}\,\mathrm{d}\tau \boldsymbol{B}_{\mathrm{p}}.$$

In order to replace the two addends in the state equation by the usual input term, one can "lift" the model by introducing the input as a new part of the state vector leading to the model

$$\Sigma_{\mathrm{d}} : \begin{cases} \begin{pmatrix} \boldsymbol{x}_{\mathrm{p}}(k+1) \\ \boldsymbol{u}(k) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}_{\mathrm{d}} & \boldsymbol{B}_{\mathrm{d2}} \\ \boldsymbol{O} & \boldsymbol{O} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_{\mathrm{p}}(k) \\ \boldsymbol{u}(k-1) \end{pmatrix} + \begin{pmatrix} \boldsymbol{B}_{\mathrm{d1}} \\ \boldsymbol{I} \end{pmatrix} \boldsymbol{u}(k) \\ \qquad\qquad \boldsymbol{y}(k) = (\boldsymbol{C}_{\mathrm{p}} \;\; \boldsymbol{O}) \begin{pmatrix} \boldsymbol{x}_{\mathrm{p}}(k) \\ \boldsymbol{u}(k-1) \end{pmatrix}. \end{cases}$$

For both models of the discrete-time plant $\Sigma_{\mathrm{d}}$ the usual theory of sampled-data systems has to be extended to cope with the new model structure brought about by the time delay $\tau$, which may vary in dependence upon the sampling time.

Time delays play an important role in the following sections of this book:

- **Observability of networked systems:** Section 2.2 investigates the observability of networked systems for linear and nonlinear systems with transmission time delays.

- **State estimation in networked control systems:** Section 3.2 proposes two architectures for estimating the overall system state of interconnected systems, where information about the subsystems are transmitted over a communication network. The main problem is to make the estimation result tolerate the imperfections of the network like packet delays and packet loss.

- **Stochastic model-based control based on virtual control inputs:** The idea to compensate missing or deliberately ignored information of subsystems by stochastic a-priori and a-posteriori information is explained in Section 4.5.

### 1.2.4 Control subject to packet loss

Packet loss describes a usual phenomenon in digital networks. Some network protocols ensure that the lost packet is sent again until it reaches the receiver, but even then time has gone and, hence, packet loss may deteriorate the control performance. Therefore, several methods have been proposed to deal with this situation.

In the network-theoretic approaches to this problem, stochastic models for packet dropping are used, where $p$ denotes the probability with which a packet does not arrive at the receiver. The critical packet drop rate is the probability $\bar{p}$ for which the average estimation error of the state estimator implemented in the intelligent actuator shown in Fig. 1.8 using the communicated information is bounded. Like in the data-rate theorems, $\bar{p}$ depends directly on the unstable eigenvalue $\lambda$ of the plant to be stabilized [340]:

$$\bar{p} < \frac{1}{\lambda^2}.$$

Further methods to handle packet loss are published, for example, in [404].

Model predictive control provides a suitable means to overcome the effects of packet loss, because at every sampling time $k$ the controller does not only determine the current control input $\boldsymbol{u}(k)$, but a sequence of future control inputs $\boldsymbol{u}(k+1)$, $\boldsymbol{u}(k+2)$,..., $\boldsymbol{u}(k+h)$ (Fig. 1.11), where $h$ is the prediction

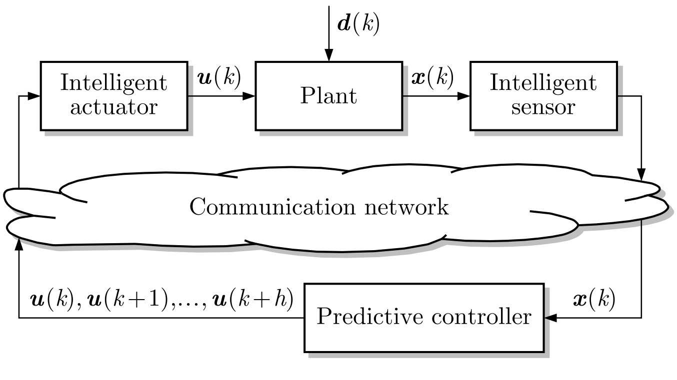


**Fig. 1.11.** Predictive control as a means to overcome the effect of packet loss

time horizon. If this sequence is transmitted to the actuator, the actuator may use the future input values if the next packets are lost.

Although this principle seems to work easily, a more detailed analysis made in Chapter 4 shows that there are several important problems to be solved. In particular, the controller can predict the future plant behavior only if it knows which inputs the actuator has applied to the plant. In case of packet loss, the controller will use a different control input for the prediction as the actuator, which will apply "old" input values. The analysis in [120, 146, 154, 297] shows that a synchronization of the controller with the actuator is only possible if acknowledgements are sent from the actuator towards the controller whenever a new input sequence has arrived.

This control structure is investigated in this monograph also with respect to delayed information transmission:

- **Compensation of delays and packet loss by model predictive control:** Section 4.2 describes how the principle of model predictive control to determine at every time step a sequence of future control inputs can be used to compensate time delays and packet loss induced by the communication network.

- **Compensation of delays and packet loss by means of virtual control inputs:** In Section 4.5 time delays and data loss in the network connection between the controller and the actuators are compensated by buffered input sequences.

- **State estimation as part of control-communication co-design:** The state estimator developed in Section 7.2 has the aim to reconstruct the plant state even in case of packet loss between the sensor and the controller and combines this estimate with a controller that is implemented by wireless communication.

### 1.2.5 Control under restrictions on the network access

The communication network can be used by only one sender at any time. Many network protocols include a scheduler that prescribes an order in which the different nodes have access to the network. If substantial time passes before a node is allowed to send, the communication constraints imposed by the network protocol influence the overall system performance and have to be included in the model of the feedback system.

In Fig. 1.12 only one signal of the set $\{y_1, y_2, u_1, u_2\}$ can be sent and, hence, only one signal of the set $\{\hat{y}_1, \hat{y}_2, \hat{u}_1, \hat{u}_2\}$ can be updated at a certain time $t_k$

$$\hat{y}_i(t_k^+) = y_i(t_k) \quad \text{or} \quad \hat{u}_i(t_k^+) = u_i(t_k) \quad \text{for one index } i \in \{1, 2\},$$

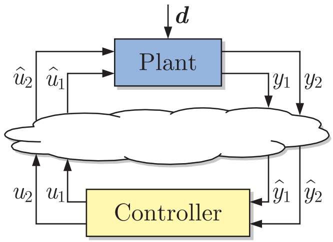


**Fig. 1.12.** Feedback loop in which only one signal may be transmitted at any time

where $t_k^+$ denotes the time just after the signal update at time $t_k$. The networked-induced errors

$$\begin{aligned} e_{yi}(t) &= y_i(t) - \hat{y}_i(t) \\ e_{ui}(t) &= u_i(t) - \hat{u}_i(t), \quad i = 1, 2 \end{aligned}$$

can be described by the differential equation

$$\dot{e}_{yi}(t) = \frac{\mathrm{d}}{\mathrm{d}t} y_i(t) - \underbrace{\frac{\mathrm{d}}{\mathrm{d}t}\hat{y}_i(t)}_{=0}$$

together with the update relation

$$e_{yi}(t_k^+) = \begin{cases} 0 & \text{if } \hat{y}_i(t_k) = y_i(t_k) \\ e_{yi}(t_k) & \text{else.} \end{cases}$$

Hence, the plant model and the controller

$$\text{Plant:} \quad \begin{cases} \dot{\boldsymbol{x}}_{\mathrm{p}}(t) = \boldsymbol{A}_{\mathrm{p}}\boldsymbol{x}_{\mathrm{p}}(t) + \boldsymbol{B}_{\mathrm{p}}\hat{\boldsymbol{u}}(t) + \boldsymbol{E}\boldsymbol{d}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}_{\mathrm{p}}\boldsymbol{x}_{\mathrm{p}}(t) \end{cases}$$

$$\text{Controller:} \quad \begin{cases} \dot{\boldsymbol{x}}_{\mathrm{c}}(t) = \boldsymbol{A}_{\mathrm{c}}\boldsymbol{x}_{\mathrm{c}}(t) + \boldsymbol{B}_{\mathrm{c}}\hat{\boldsymbol{y}}(t) \\ \boldsymbol{u}(t) = \boldsymbol{C}_{\mathrm{c}}\boldsymbol{x}_{\mathrm{c}}(t) \end{cases}$$

together with the model of the network-induced errors lead to the following model of the closed-loop system:

$$\begin{pmatrix} \dot{\boldsymbol{x}}_{\mathrm{p}}(t) \\ \dot{\boldsymbol{x}}_{\mathrm{c}}(t) \\ \dot{\boldsymbol{e}}_y(t) \\ \dot{\boldsymbol{e}}_u(t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}_{\mathrm{p}} & \boldsymbol{B}_{\mathrm{p}}\boldsymbol{C}_{\mathrm{c}} & \boldsymbol{O} & -\boldsymbol{B}_{\mathrm{p}} \\ \boldsymbol{B}_{\mathrm{c}}\boldsymbol{C}_{\mathrm{p}} & \boldsymbol{A}_{\mathrm{c}} & -\boldsymbol{B}_{\mathrm{c}} & \boldsymbol{O} \\ \boldsymbol{C}_{\mathrm{p}}\boldsymbol{A}_{\mathrm{p}} & \boldsymbol{C}_{\mathrm{p}}\boldsymbol{B}_{\mathrm{p}}\boldsymbol{C}_{\mathrm{c}} & \boldsymbol{O} & -\boldsymbol{C}_{\mathrm{p}}\boldsymbol{B}_{\mathrm{p}} \\ \boldsymbol{C}_{\mathrm{c}}\boldsymbol{B}_{\mathrm{c}}\boldsymbol{C}_{\mathrm{p}} & \boldsymbol{C}_{\mathrm{c}}\boldsymbol{A}_{\mathrm{c}} & -\boldsymbol{C}_{\mathrm{c}}\boldsymbol{B}_{\mathrm{c}} & \boldsymbol{O} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_{\mathrm{p}}(t) \\ \boldsymbol{x}_{\mathrm{c}}(t) \\ \boldsymbol{e}_y(t) \\ \boldsymbol{e}_u(t) \end{pmatrix} + \begin{pmatrix} \boldsymbol{E} \\ \boldsymbol{O} \\ \boldsymbol{C}_{\mathrm{p}}\boldsymbol{E} \\ \boldsymbol{O} \end{pmatrix} \boldsymbol{d}(t) \tag{1.2}$$

$$\begin{pmatrix} \boldsymbol{e}_y(t_k^+) \\ \boldsymbol{e}_u(t_k^+) \end{pmatrix} = \phi_k \left( \begin{pmatrix} \boldsymbol{e}_y(t_k) \\ \boldsymbol{e}_u(t_k) \end{pmatrix} \right). \tag{1.3}$$

This model shows that the closed-loop system behaves like an *impulsive system* with Eq. (1.2) describing the continuous flow and Eq. (1.3) the state jumps. The time points $t_k$, $(k = 0, 1, ...)$ are prescribed by the network protocol. At these time points a node has access to the network and the network-induced error of this node is set to zero. The mapping $\phi_k$ depends upon the network protocol. It shows which element of the vectors $\boldsymbol{e}_y(t_k^+)$ or $\boldsymbol{e}_u(t_k^+)$ are set to zero.

For a survey on hybrid dynamic systems cf. [247] and, in particular, on impulsive systems cf. [22, 179]. Further representations of control loops implemented by digital communication networks as hybrid dynamic systems can be found in [81].

From an engineering viewpoint, it is obvious that the network should not bring about large delays into the closed loop if the performance should not be deteriorated. The important aspect of systematic ways to analyze the relation between the quality of service (QoS) of the communication network and the quality of performance (QoP) of the control loop is the fact that upper bounds on the *maximum allowable transmission interval* (MATI) can be derived [230].

Reference [394] shows that the MAC protocol (medium access control protocol) is responsible for the time delay of the network. It analyzes several protocols and shows their implication for implementing control loops. Analytical results based on the hybrid representation (1.2), (1.3) of the control loop have been obtained in [63, 108, 168].

The investigations on the effect of communication constraints on the closed-loop system performance have started more than ten years ago with extensive simulation studies and the development of dedicated software tools like the MATLAB toolboxes *TrueTime* [50] and *jitterbug* [49].

**Control-communication co-design.** In the co-design of the controller and the network scheduler, the timing of the communication events and the computation steps to be performed by the controller are selected simultaneously [75]. The main question to be answered concerns the selection of the node that is allowed to send its data at a certain time point with the aim to stabilize an unstable plant or to optimize the closed-loop performance. If the sequence of sending nodes is denoted by $\sigma = (\sigma(0), \sigma(1), ..., \sigma(k_\mathrm{e} - 1))$ and the sequence of control inputs by $U = (\boldsymbol{u}(0), \boldsymbol{u}(1), ..., \boldsymbol{u}(k_\mathrm{e} - 1))$ the co-design problem canbe formulated as the following optimization task:

$$J(\boldsymbol{x}_0, \sigma) = \sum_{k=0}^{k_\mathrm{e}-1} \left( \boldsymbol{x}^\mathrm{T}(k) \boldsymbol{Q} \boldsymbol{x}(k) + \boldsymbol{u}^\mathrm{T}(k) \boldsymbol{R} \boldsymbol{u}(k) \right) + \boldsymbol{x}(k_\mathrm{e}) \boldsymbol{Q}_\mathrm{e} \boldsymbol{x}(k_\mathrm{e}) \rightarrow \min_{\sigma, U}.$$

In a simplified version, the co-design problem can be reduced to a scheduling problem by considering the given state-feedback controller

$$\boldsymbol{u}(k) = -\boldsymbol{K}\boldsymbol{x}(k),$$

which should be implemented by using a digital communication network in the best possible way. With the notation for the network-induced errors introduced earlier the closed-loop system consisting of the components

$$\begin{aligned} \text{Plant:} &\ \begin{cases} \dot{\boldsymbol{x}}_{\mathrm{p}}(t) = \boldsymbol{A}_{\mathrm{p}}\boldsymbol{x}_{\mathrm{p}}(t) + \boldsymbol{B}_{\mathrm{p}}\hat{\boldsymbol{u}}(t), \quad \boldsymbol{x}_{\mathrm{p}}(0) = \boldsymbol{x}_{\mathrm{p}0} \\ \boldsymbol{y}(t) = \boldsymbol{C}_{\mathrm{p}}\boldsymbol{x}_{\mathrm{p}}(t) \end{cases} \\ \text{Controller:} &\ \left\{ \boldsymbol{u}_{\mathrm{p}}(t) = -\boldsymbol{K}\hat{\boldsymbol{x}}_{\mathrm{p}}(t) \right. \end{aligned}$$

is represented by

$$\begin{pmatrix} \dot{\boldsymbol{x}}_{\mathrm{p}}(t) \\ \dot{\boldsymbol{e}}_{\mathrm{x}}(t) \end{pmatrix} = \underbrace{\begin{pmatrix} \boldsymbol{A}_{\mathrm{p}} - \boldsymbol{B}_{\mathrm{p}}\boldsymbol{K} & \boldsymbol{B}_{\mathrm{p}}\boldsymbol{K} \\ \boldsymbol{C}_{\mathrm{p}}\boldsymbol{A}_{\mathrm{p}} - \boldsymbol{C}_{\mathrm{p}}\boldsymbol{B}_{\mathrm{p}}\boldsymbol{K} & \boldsymbol{C}_{\mathrm{p}}\boldsymbol{B}_{\mathrm{p}}\boldsymbol{K} \end{pmatrix}}_{\bar{\boldsymbol{A}}} \begin{pmatrix} \boldsymbol{x}_{\mathrm{p}}(t) \\ \boldsymbol{e}_{\mathrm{x}}(t) \end{pmatrix}$$

$$\begin{pmatrix} \boldsymbol{x}_{\mathrm{p}}(t_k^+) \\ \boldsymbol{e}_{\mathrm{x}}(t_k^+) \end{pmatrix} = \boldsymbol{J}_{\sigma(k)} \begin{pmatrix} \boldsymbol{x}_{\mathrm{p}}(t_k) \\ \boldsymbol{e}_{\mathrm{x}}(t_k) \end{pmatrix},$$

where $\boldsymbol{e}_{\mathrm{x}}(t)$ denotes the network error of the state $\boldsymbol{x}(t)$. If, for simplicity, the communication events occur equidistant in time

$$t_k - t_{k-1} = T,$$

the closed-loop system has the discrete-time model

$$\text{Control loop}: \begin{cases} \begin{pmatrix} \boldsymbol{x}_{\mathrm{p}}(k+1) \\ \boldsymbol{e}_{\mathrm{x}}(k+1) \end{pmatrix} = \bar{\boldsymbol{A}}_{\sigma(k)} \begin{pmatrix} \boldsymbol{x}_{\mathrm{p}}(k) \\ \boldsymbol{e}_{\mathrm{x}}(k) \end{pmatrix} \\ \boldsymbol{y}(k) = (\boldsymbol{C}_{\mathrm{p}} \ \ \boldsymbol{O}) \begin{pmatrix} \boldsymbol{x}_{\mathrm{p}}(k) \\ \boldsymbol{e}_{\mathrm{x}}(k) \end{pmatrix} \end{cases}$$

with

$$\bar{\boldsymbol{A}}_{\sigma(k)} = \mathrm{e}^{\bar{\boldsymbol{A}}T}\boldsymbol{J}_{\sigma(k)}.$$

This model shows again that the overall system behaves like an impulsive system where the jump condition represented by the matrix $\boldsymbol{J}_{\sigma(k)}$ depends upon the choice $\sigma(k)$ of the node allowed to send data at time instant $k$. Thismodel can be used to solve the following simplified design problem

$$\hat{J}(\boldsymbol{x}_0, \sigma) = \sum_{k=0}^{k_{\mathrm{e}}-1} \boldsymbol{x}^{\mathrm{T}}(k)\hat{\boldsymbol{Q}}\boldsymbol{x}(k) + \boldsymbol{x}(k_{\mathrm{e}})\boldsymbol{Q}_{\mathrm{e}}\boldsymbol{x}(k_{\mathrm{e}}) \rightarrow \min_{\sigma}.$$

References [9, 10] describe a solution of this problem that minimizes the objective function over a restricted time horizon $k_{\mathrm{e}}$ under the assumption that after this time horizon a predefined communication sequence is periodically applied.

Results along this line of research are found in the following sections of this monograph:

- **Control and communication co-design:** Section 7.2 proposes a decomposition of dedicated wireless networked control systems into a control layer and a communication layer to investigate the connections between both layers on a common methodological basis.

- **Control requirements on network protocols:** The reduction of information sent over the communication channels is useful only if it really reduces the network load and, hence, improves the quality of service. Section 7.3 relates properties of available network protocols with requirements of the control loop to send information whenever this is necessary for ensuring a given level of performance.

**Event-based control.** Whereas the methods mentioned above have the aim to distribute the capability of the communication network in a reasonable way among the nodes of the feedback loop, event-based control aims at reducing the need for sending information. The use of the data links should be adapted to the current plant state and new information should be sent over the network only if this information is necessary for ensuring a certain level of performance of the closed-loop system.

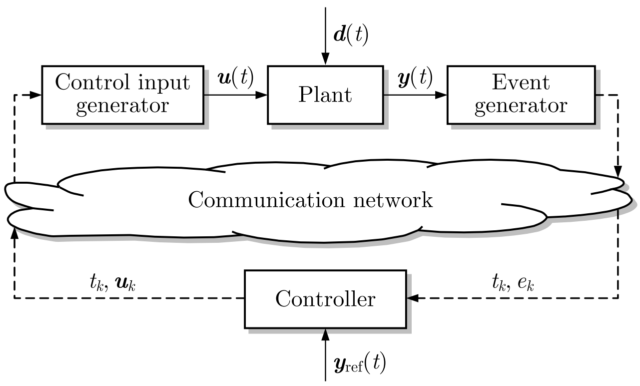


**Fig. 1.13.** Event-based control loop

The basic structure of event-based control loops is shown in Fig. 1.13. In comparison to the usual sampled-data control loop, the sampler is replaced by an event generator and the hold by an control input generator. The event generator decides at which instant of time, denoted by $t_k$, sensor information is sent to the controller and information about the control input from the controller towards the actuators. The control input generator transfers the input symbol $e_{\mathrm{u}k}$ obtained at time $t_k$ into a continuous-time signal $\boldsymbol{u}(t)$ that is applied to the plant by the actuators.

Event-based control can be applied in various situations

- to reduce the data flow over the network and, hence, mitigate the constraints on the network access,
- to reduce the energy consumption of wireless sensors or actuators by reducing the data traffic from or towards these components,
- to adapt the working principle of the control loop to the event-based nature of sensors.

Furthermore, the question under what conditions information feedback is necessary for reaching a control goal is important for the principal understanding of feedback control.

Event-based control is studied extensively in Chapter 4 with the following aims:

- **Stabilization and disturbance attenuation:** Sections 5.2 and 5.4 propose new event-based control strategies that bring the system state into a bounded region around the set-point and make this region invariant under bounded disturbances.

- **Distributed event-based control:** For interconnected systems, event-based controllers consist of decentralized or distributed controllers for the subsystems. Sections 5.3, 5.5 and 5.6 explain how the performance of the overall system can be analyzed in terms of the subsystem properties.

- **Stochastic event-based control:** The event generator and the control input generator can be designed separately only if the overall system has a nested information structure. This result is described in Section 5.7 on stochastic event-based control.

### 1.2.6 Distributed control of interconnected systems and multi-agent systems

With the new communication structures, the architecture of the future control systems will change. The current hierarchically organized networks with physical subsystems at the bottom and hierarchical coordinators on higher layers (Fig. 1.14 (left)) will be replaced by distributed control structures, where the nodes represent physical subsystems together with their embedded control units (Fig. 1.14 (right)). The main question to be answered asks which groups of nodes in this network have to communicate directly.

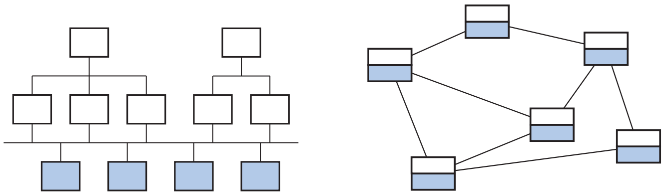

**Fig. 1.14.** Hierarchical vs. distributed control systems

The answer to this question depends upon the control aim and, consequently, different lines of research have been followed in the past. The following paragraphs survey the problems of distributed control and of the control of multi-agent systems, to both of which this monograph provides new methods and results.

**Distributed control.** If the plant consists of several subsystems, which are physically coupled, the overall system can be described as a collection of $N$ subsystems $\Sigma_i$ and a coupling relation $K$ as follows:

$$\Sigma_i : \begin{cases} \dot{\boldsymbol{x}}_i(t) = \boldsymbol{A}_i \boldsymbol{x}_i(t) + \boldsymbol{B}_i \boldsymbol{u}_i(t) + \boldsymbol{E}_i \boldsymbol{s}_i(t), \quad \boldsymbol{x}_i(0) = \boldsymbol{x}_{i0} \\ \boldsymbol{y}_i(t) = \boldsymbol{C}_i \boldsymbol{x}_i(t) \\ \boldsymbol{z}_i(t) = \boldsymbol{C}_{\mathrm{z}i} \boldsymbol{x}_i(t), \quad i = 1, 2, ..., N \end{cases}$$

$$K : \left\{ \begin{pmatrix} \boldsymbol{s}_1(t) \\ \boldsymbol{s}_2(t) \\ \vdots \\ \boldsymbol{s}_N(t) \end{pmatrix} = \boldsymbol{L} \begin{pmatrix} \boldsymbol{z}_1(t) \\ \boldsymbol{z}_2(t) \\ \vdots \\ \boldsymbol{z}_N(t) \end{pmatrix} \right. .$$

The problem is to find a controller of the overall system that works predominantly locally at the subsystems. That is, the control input $\boldsymbol{u}_i(t)$ of the subsystem $\Sigma_i$ should be determined mainly in dependence upon the output $\boldsymbol{y}_i(t)$ of the same subsystem and communication among the control stations of the subsystems should be reduced to a minimum.

Early research on distributed control has been concentrated on decentralized controllers, which do not at all interact directly and, hence, have the control laws of the form

$$\boldsymbol{u}_i(t) = \boldsymbol{K}_{ii} \boldsymbol{y}_i(t)$$

(cf. [245, 339]). In a more general setting, distributed control allows the control stations to communicate directly with the consequence that the control law extends towards

$$\boldsymbol{u}_i(t) = -\boldsymbol{K}_{ii} \boldsymbol{y}_i(t) - \sum_{j=1, j \neq i}^{N} \boldsymbol{K}_{ij} \boldsymbol{y}_j(t). \tag{1.4}$$

In this general setting, only the matrices $\boldsymbol{K}_{ij} \neq \boldsymbol{O}$ necessitate to communicate the state vector $\boldsymbol{x}_j(t)$ of subsystem $\Sigma_j$ towards the control station of subsystem $\Sigma_i$.

Distributed control has two aspects. First, it has to be decided, which communication links among the control stations are necessary for reaching the control aim. The result is a communication topology, which can be described by a directed graph, in which each vertex represents a subsystem together with the corresponding control station and each edge a communication link. The main problem is to find the reasonable communication topology for a given control task.

The second aspect of distributed control concerns the design step. *Centralized control design* means to use a model of the overall plant to find the controllers of all subsystems together. If optimal control methods are used, the design problem is usually stated as the optimization problem

$$\min_{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_N} \sum_{i=1}^{N} \int_0^\infty \boldsymbol{x}_i^{\mathrm{T}}(t) \boldsymbol{Q} \boldsymbol{x}_i(t) + \boldsymbol{u}_i^{\mathrm{T}}(t) \boldsymbol{R} \boldsymbol{u}_i(t) \, \mathrm{d}t \tag{1.5}$$

subject to the constraint (1.4).

In *decentralized control design* the control station $C_i$ should be found by using the model of the subsystem $\Sigma_i$ only. As a corresponding formulation of the design problem

$$\min_{\boldsymbol{u}_i} \int_0^\infty \boldsymbol{x}_i^{\mathrm{T}}(t) \boldsymbol{Q} \boldsymbol{x}_i(t) + \boldsymbol{u}_i^{\mathrm{T}}(t) \boldsymbol{R} \boldsymbol{u}_i(t) \, \mathrm{d}t$$

does not take into account the couplings among the subsystems, the optimal solutions to all subsystem problems do not represent an optimal solution to the overall problem (1.5) and modifications of the subsystem problems have to be made to shift the design results of the subsystems towards an optimal solution to the overall system problem. One way of accomplishing this step is to introduce a coordinator as shown in Fig. 1.15. The coordinator communicates with the "optimizers" $O_i$ at the subsystems in order to influence the result such that the behavior of the subsystems $P_i$, which include the plant subsystem $\Sigma_i$ and the control station $C_i$, satisfies the global control goal.

Distributed control is investigated in the following sections of this book:

- **Two-layer optimal control of distributed systems:** Section 3.3 gives a survey of results on distributed optimization and the design of optimal controllers, where the controller parameter should be found by the subsystems without any complete model of the overall system.

- **Distributed model predictive control:** Chapter 4 describes several algorithms for model predictive control, where the optimization problem is solved by the subsystems with partial information about the overall system state.

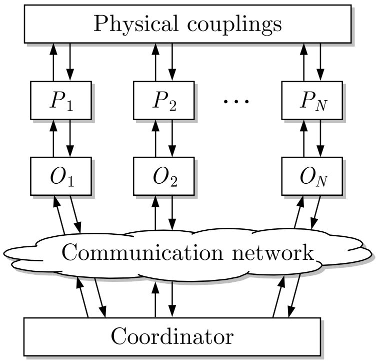


**Fig. 1.15.** Hierarchical structure for designing the distributed controller of an interconnected system

- **Distributed control with situation-dependent communication:** For physically interconnected systems, disturbances can be attenuated either by decentralized controllers of the subsystems or by interacting controllers. Section 6.5 shows how both methods can be combined by exchanging information among the control stations only if the current state of the system necessitates such communications.

**Multi-agent systems.** The availability of wireless communication networks has increased the interest in the control of multi-agent systems, where physically uncoupled subsystem should be made to satisfy a common goal. Examples of such systems are vehicle platoons, where the vehicles should follow a leader with the same velocity and prescribed distances, or cooperating robots that should distribute over a certain region to get measurements or that should meet at a common place. Hence, multi-agent systems comprise a special class of interconnected systems, where the interconnections have to be introduced by the networked controller (Fig. 1.16).

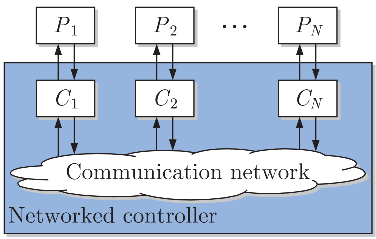


**Fig. 1.16.** Multi-agent system

Many control problems for multi-agent systems can be formulated as consensus or synchronization problems. The networked controller has the task to bring all agent outputs $\boldsymbol{y}_i(t)$, $(i = 1, 2, ..., N)$ onto the same trajectory $\boldsymbol{y}_{\mathrm{s}}(t)$, which is called the consensus trajectory or synchronous trajectory:

$$\lim_{t\to\infty} \|\boldsymbol{y}_i(t) - \boldsymbol{y}_{\mathrm{s}}(t)\| = 0, \quad i = 1, 2, ..., N.$$

There are two important aspects of multi-agent systems with respect to networked control. First, the coupling structure of the overall system is usually not given, but has to be found as a result of the controller design. As research on consensus and synchronization problems have shown, the choice of this structure is more important than the selection of specific controller parameters within this structure.

Second, the synchronous trajectory $\boldsymbol{y}_{\mathrm{s}}(t)$ appears as a result of self-organization. It depends upon the dynamics and the initial state of all agents and is not prescribed by an external entity. The exception from this general case is the leader-follower structure where one agent acts as a leader prescribing the common trajectory for all other agents, which act as followers.

The majority of papers on consensus and synchronization problems assume that the agents have identical dynamics [117, 187, 278, 319, 328, 363]. Then the synchronous trajectory is a solution to the common agent model. Under weak conditions on the agent dynamics [241], synchronization can be obtained by a static feedback, where the local components $C_i$ of the networked controller have no dynamics (Fig. 1.16).

The problems becomes more involved if agents with individual dynamic properties are considered. Then a common trajectory only exists if the agents have some dynamics in common [243, 385]. This necessary condition on synchronizability can be satisfied by introducing dynamic components $C_i$ into the networked controller.

Several consensus and synchronization problems are dealt with in this monograph:

- **Distributed model predictive control of multi-agent systems:** The idea of using model predictive control for cooperative control tasks is presented in Section 4.3 as an extension of distributed model predictive control.

- **Synchronizable subspaces:** The question under what conditions agents with identical linear dynamics can be synchronized is answered in Section 6.3 by deriving conditions on the network structure in terms of the Laplacian matrix of the communication graph.

- **Synchronization of agents with individual dynamics:** Whereas the literature on consensus problems and synchronization is mainly restricted to sets of agents with identical dynamics, Section 6.2 deals with agents with individual dynamics. It is shown that synchronization can only occur if the agents have some dynamics in common, which is represented by a (virtual) reference system.

- **Cluster synchronization:** Section 6.4 shows interesting cluster phenomena of nonlinear agents and develops methods to identify which agents form a group of mutually synchronized agents. Due to the nonlinearities in the interconnections, synchronization can be achieved only by groups of agents rather than by all agents in the network.

## 1.3 New mathematical concepts for networked control systems

Control theory and control engineering have used sophisticated mathematical concepts at least since the times of Routh and Hurwitz and their development of easy checkable stability criteria for linear systems. It is, therefore, not surprising that also a theory of networked control systems substantially relies on mathematical concepts. In this section we give an informal overview of the main techniques used in this monograph, referring to the respective sections in the book for details. Some of these concepts, like graph theory, have primarily gained interest in control because of their applicability to networked control systems. Other concepts, like mathematical optimization or Lyapunov functions, have been successfully used in control theory for many decades. For the latter, we particularly explain which extensions are used in this monograph in order to apply them in a network setting.

### 1.3.1 Optimization and optimal control

Optimization and optimal control techniques are ubiquitous in this monograph. Already in the overview in the previous sections optimal control techniques have been mentioned in several places, either in the form of linear quadratic optimal control or in the form of model predictive control. The popularity of optimization and optimal control techniques for handling networked control problems has various reasons.

To begin with, optimization algorithms have been studied in decentralized settings for many years. Section 3.3 gives an introduction into primal and dual decomposition techniques which allow to solve optimization problems by decomposing them into several subproblems that can be solved in parallel. In the case of networked control systems with many subsystems such decompositions are naturally induced by the system structure and in this way they are used, e.g., in Sections 3.2, 3.3, 4.3 and 6.5.

Second, on-line optimal control methods form the basis of the currently very popular model predictive control (MPC) approach for controller design.

Clearly, all the possibilities to decompose optimization problems also apply to MPC and this is used in different ways in Section 4.3. An additional benefit of MPC is that instead of just a single control value a whole sequence of control inputs along with a corresponding prediction of the future system state is available in each sampling period "for free". On the one hand, this allows to use the additional control values as backup in order to compensate for packet loss or unacceptably large delays. Moreover, the prediction mechanism already available in MPC can be used in order to compensate for delays. These techniques are described in Section 4.2. On the other hand, in the case of several subsystem, MPC allows for communicating the predictions to the other subsystems. Different approaches using this technique are described in Sections 4.3–4.5.

Another benefit of optimal control based controller design is the possibility to encode different goals in both the objective and the constraints. In this monograph, this fact is used in distributed control approaches, in which the individual subsystems do not communicate but rather the other subsystems' influence is taken care of by a judicious formulation of the optimal control problem. For instance, the optimal control problem can ensure that the subsystems are dissipative (Section 4.3), robust w.r.t. errors in the predictions communicated by the other subsystems (Section 4.4) or input-to-state stable (Section 5.5). Another way to exploit the flexibility of optimal control based design is to incorporate network induced additional requirements like low communication effort directly into the optimization objectives. Such approaches are discussed in Section 5.7 in an event-based setting and in Section 7.2 in the context of a cross-design approach. A study of network effects on the performance of optimal controllers can be found in Section 7.3.

Finally, optimization techniques can not only be used for controller design but also for estimation as described in Section 3.2 and for the analysis of the systems behavior. Examples for the latter in this book are the stability analysis for MPC subject to packet loss in Section 4.2, the LMI criterion for synchronization presented in Section 6.3 and the cluster synchronization analysis in Section 6.4

### 1.3.2 Dynamical properties of control systems: controllability, observability and stability

The goal of most control-theoretic approaches is to either verify or to enforce (by design of a suitable controller) desired dynamic properties of the control system under consideration. Just like for their non-networked counterparts, for networked control systems the classical properties of controllability, observability and stability are essential for understanding the basic properties of the system.

Observability and controllability are thoroughly investigated for networked linear systems in Section 2.2. At a first glance, the mathematical tools for their verification like coprime factorization and rank conditions are similar to the analysis of non-networked systems. However, the “twist” in the mathematical analysis of networked systems comes from the special structures of the matrices under consideration. Controllability – though in the weaker form of asymptotic controllability to a desired equilibrium – also pops up in a nonlinear context in the analysis of MPC schemes without terminal constraints in Sections 4.2 and 4.3.

Stability is one of the key properties a controlled system should satisfy and it is, therefore, featured in many contributions in this monograph. In fact, even several approaches which do not consider classical stability concepts but rather invariance properties as in Section 2.3, ultimate boundedness as in Sections 5.2–5.4 or synchronization as in Section 6.3 use mathematical techniques similar to those from stability analysis.

The two main approaches for verifying stability found in this monograph are spectral theory for linear systems and and Lyapunov functions for both linear and nonlinear systems. In the time-invariant case, spectral theory boils down to computing eigenvalues of matrices. Similar to what was said for controllability above, in the networked context the matrices under consideration also reflect properties of the communication network and the coupling structure between the subsystems. Examples where such spectral information is used can be found, e.g., in those results of Section 2.3 which deal with linear systems as well as in Sections 5.3 and 6.3. In case the coupling structures are explicitly represented by a matrix $\boldsymbol{L}$ (as, e.g., in Section 5.3), respective conditions can be given in small-gain form, which essentially amounts to demanding that the couplings between the individual subsystems are sufficiently weak.

Small-gain type arguments can also be employed for nonlinear systems as in Sections 5.5 and 5.6 in an ISS Lyapunov function framework. Note that here the Lyapunov functions are not merely tools for the analysis but also used for design purposes, e.g., for determining suitable event times in Section 5.5. ISS Lyapunov functions are an appropriate tool for distributed control in which the mutual influences of the subsystems are considered as perturbations and no communication takes place.

However, Lyapunov function based stability analysis can also be employed in case the subsystems communicate, as in the MPC approaches in Sections 4.3–4.5. In order to address delays, extensions of the Lyapunov function concept in the sense of Krasovskii or Razumikhin can be used (Section 4.2).

It should finally be remarked that the analysis of dynamic properties is not necessarily restricted to the time-invariant or autonomous case, even if the underlying control system is time-invariant. This is, for instance, the case in the analysis of the topological entropy in Section 2.3, where the non autonomy is induced by the control input and methods from the theory of non-autonomous dynamic systems are applied. Another example is the networked MPC scheme in Section 4.2, in which the non autonomy is induced by possible network failures.

### 1.3.3 Graph theory and structured matrices

In contrast to the concepts discussed in the previous two subsections, which are carried over from non-networked control systems to the networked setting, graph theory is more directly linked to the network aspect. Indeed, except for the more “exotic” use of graph theory in the set-oriented solution of quantized optimal control problems in Section 5.4, graphs in this monograph always serve for formalizing the interconnection structure between the different subsystems in a networked control system.

As an alternative to graphs - particularly in a linear setting - the interconnection structure can also be expressed via a matrix whose entries represent the coupling gains between the subsystems (Section 2.2). Formally, the graphs and these matrices are connected via the concept of the adjacency matrix (Section 3.2). More generally, interconnections are expressed by nonlinear gain matrices in Sections 5.5 and 5.6.

Regardless of the particular form, both graph-theoretic as well as matrix-valued representations have the same goal: they encode the coupling structure and allow for identifying and formulating structural conditions leading to desired system properties.

Matrix theoretic representations of coupling structures are used in Sections 2.2, 3.2, 3.3 and 5.3. The results obtained this way range from genericity results for controllability and observability via the exploitation of sparsity patterns in distributed optimal estimation and control to the formulation of small-gain conditions for stability.

Graph theoretic conditions are, on the other hand, used in Sections 4.4, 6.2 and 6.4. Here, graph theoretic concepts are used, e.g., to introduce a no cycle condition for robust MPC, to establish a relation between the existence of spanning trees and synchronization in leaderless multi-agent systems and for the formalization of clustering phenomena occuring in synchronizing systems.

This last application, which can be found in Section 6.4, combines the graph-theoretic representation with optimization approaches. Along with many other examples in this book it shows that the mathematical concepts surveyed in this section are typically not used isolated from each other but that a combination of such methods is needed in order to address analysis and design problems for networked control systems.