RUHR-UNIVERSITÄT BOCHUM
Institute of Automation and Computer Control
Prof. Dr.-Ing. Jan Lunze

Universitätsstr. 150 Phone +49-(0)234 32-28071
D-44805 Bochum Fax +49-(0)234 32-14101

Plug-and-play reconfiguration in networked control systems

Sven Bodenburg
bodenburg@atp.rub.de

1 Introduction

Plug-and-play control is an approach to deal with the automated reconfiguration of the control algorithms executed on the control stations C_i (Fig. 1) after structure changes of the plant or modifications in the control objectives. Moreover, plug-and-play control provides an automated implementation of the new control algorithm at the control equipment at runtime.

Nowadays, the network connection for the control of large-scale systems is used to exchange signals among the control stations. In particular, communication links can be established or detached automatically at runtime depending on the current situation. Future applications relay on a more advanced information exchange that is not restricted to signals but include the automated exchange of algorithms (double dashed arrows in Fig. 1) between the control devices during operating time to enable structural changes of the plant (e.g., failures affecting the plant [3], add or remove subsystems) or modifications in the objectives (e.g., changing formation of UAVs).

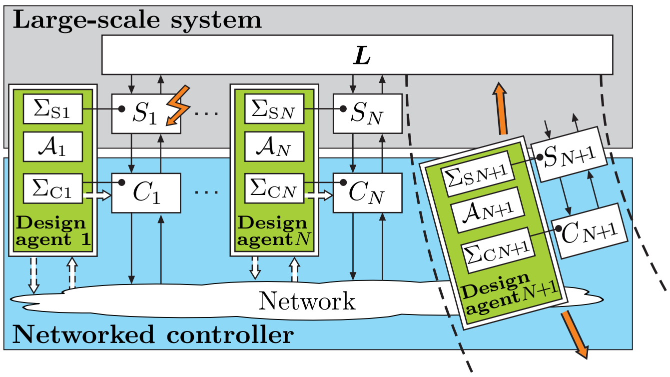

Figure 1: Structural changes affect the plant

Due to the complexity of large-scale systems, the reconfiguration task leads to severe difficulties [1]. In particular, it has to be dealt with the absence of the overall system model for reconfiguration. This is due to privacy of subsystems for example. Moreover, a controller reconfiguration has to cause minimal invasion in the control stations. To handle these restrictions, the reconfiguration task has to be decomposed and assigned to the control station.

Figure 1 illustrates the architecture of plug-and-play control. The local design agents have knowledge about the corresponding subsystem model $\Sigma_{\mathrm{S}i}$, control algorithm $\Sigma_{\mathrm{C}i}$ and a local control aim $\mathcal{A}_i$ and are able to exchange these information among each other.

An implementation of the algorithm exchange among control stations for MATLAB/Simulink driven plants is presented in [2]. The reconfiguration problem is stated as follows:

Problem: Changes in the plant result in violation of the nominal control aim or the nominal aim is modified itself. To satisfy the nominal or new requirements a controller reconfiguration is necessary.

Restriction:
- No overall system model available.
- The reconfiguration should be minimal invasive.

Given:
- Interconnection oriented structure $\Sigma_{\mathrm{S}i}$, $\Sigma_{\mathrm{C}i}$, $\boldsymbol{L}$.
- Global control aim $\mathcal{A}$.
- Result of the surveillance.

Find: On-line reconfiguration method to retrieve the nominal specifications or to satisfy new specifications with respect to limited model information and minimal invasion on the controller.

Require: Entire formalisation of the reconfiguration problem.

2 Stability condition with limited model information

To solve this problem, it is firstly of interest to find a decomposition of the given global control aim $\mathcal{A}$ such that this aims can the checked with limited model information. To do so, the main idea is to decompose the overall system into the controlled subsystems (S_i, C_i) and corresponding residual systems (R_i) as shown in Fig. 2.

The N controlled subsystems are represented in frequency domain

$$\bar{\Sigma}_i : \begin{cases} \boldsymbol{y}_i(s) &= \boldsymbol{G}_{\mathrm{yw}i}(s)\boldsymbol{w}_i(s) + \boldsymbol{G}_{\mathrm{ys}i}(s)s_i(s), \\ z_i(s) &= \boldsymbol{G}_{\mathrm{zw}i}(s)\boldsymbol{w}_i(s) + G_{\mathrm{zs}i}(s)s_i(s) \end{cases}$$

for $i = 1, ..., N$, where $\boldsymbol{y}_i, \boldsymbol{w}_i \in \mathbb{C}^{r_i}$ are the sensor signal and reference signal respectively and $s_i, z_i \in \mathbb{C}$ are the coupling input and coupling output signal respectively. The physical interconnection is given by

$$\boldsymbol{s}(s) = \boldsymbol{L}\boldsymbol{z}(s),$$

where $\boldsymbol{s}, \boldsymbol{z} \in \mathbb{C}^N$ denote the composed interconnection input and output respectively and $\boldsymbol{L} \in \mathbb{R}^{(N\times N)}$ is the interconnection matrix. The set of neighbours $\mathcal{N}_i$ of the subsystem S_i are those subsystems which influence S_i directly, i.e., $\mathcal{N}_i = \{j : l_{ij} \neq 0\}$. The residual system is represented by

$$\Sigma_{\mathrm{R}i}: \quad s_i(s) = G_{\mathrm{R}i}(s) z_i(s), \tag{1}$$

where $G_{\mathrm{R}i}$ have to describe the remaining system approximatively. These information $(\bar{\Sigma}_i, \Sigma_{\mathrm{R}i}, \mathcal{N}_i)$ are located at the design agents Dai shown in Fig. 2

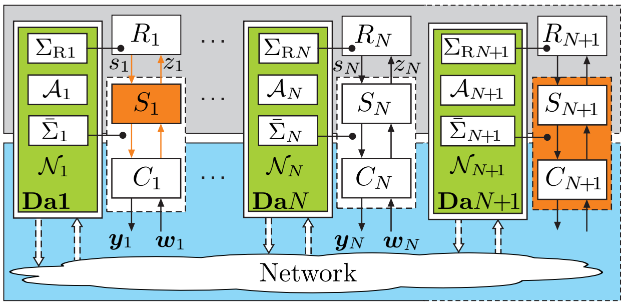

Figure 2: Decomposition of the overall system

The question of interest is the following:

> *Which are the additional requirements to be satisfied by the isolated controlled subsystem to guarantee the satisfaction of the global control aim, i.e.,*
>
> $$(\bar{\Sigma}_i, \Sigma_{\mathrm{R}i}, \mathcal{N}_i) \in \mathcal{A}_i, \forall i \quad \Rightarrow \quad (\Sigma_{\mathrm{S}}, \Sigma_{\mathrm{C}}) \in \mathcal{A} \ ? \tag{2}$$

Clearly, the answer of this question depends upon the chosen residual model (1).

Focusing on stability of the overall system, the global control aim yields

$$\mathcal{A}: \begin{cases} 1. \quad \text{Stability of } \bar{\Sigma}_i, \ \forall i \\ 2. \quad \text{The Nyquist plot } \det\big(\boldsymbol{I} - \mathrm{diag} G_{\mathrm{zs}i}(s) \boldsymbol{L}\big) \\ \quad\ \ \text{along } \mathcal{D} \text{ does not encircle the origin of the} \\ \quad\ \ \text{complex plane.} \end{cases} \tag{3}$$

which can only be verified with global system information.

If the upper bound of the amplitude response of the residual system (1) is chosen to

$$|G_{\mathrm{R}i}(s)| < |\boldsymbol{l}_i \mathbb{1}|,$$

where $\boldsymbol{l}_i$ is the ith row of $\boldsymbol{L}$ and locally known, the overall system is stable if all controlled subsystems guarantee the satisfaction of

$$\mathcal{A}_i: \begin{cases} 1. \quad \text{Stability of } \bar{\Sigma}_i, & (4) \\ 2. \quad |\alpha_i(s)| = \displaystyle\sum_{j \in \mathcal{N}_i} |G_{\mathrm{zs}i}(s) l_{ij}| < 1, \quad \forall s \in \mathcal{D} & (5) \end{cases}$$

which can be interpreted as small gain condition. Both conditions (4) and (5) can be checked with local model information such that the implication (2) holds.

3 Example: Multizone furnace for crystal growth

Temperature control of a multizone furnace for crystal growth is considered as an application example (Fig. 3). The physical interconnection yields a heat transfer between neighbouring heating zones. The multizone furnace is controlled by decentralised controller where each controlled zone is stable by design. Thus, condition (3) and (4) are already fulfilled. Now it is to check the satisfaction

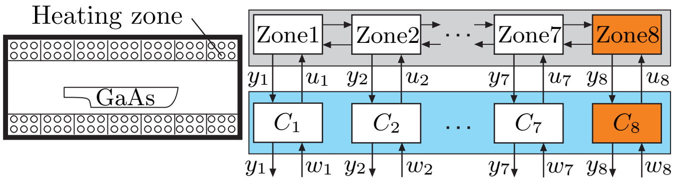

Figure 3: Multizone furnace for crystal growth

of condition (5) such that the implication (2) holds. In Fig. 4 a) the behaviour of $\alpha_i(j\omega)$ is shown which results from (5). Thus, the decomposed control aim $\mathcal{A}_i$ is fulfilled by each controlled zone. As illustrated in Fig. 4 b), the global control aim $\mathcal{A}$ is satisfied.

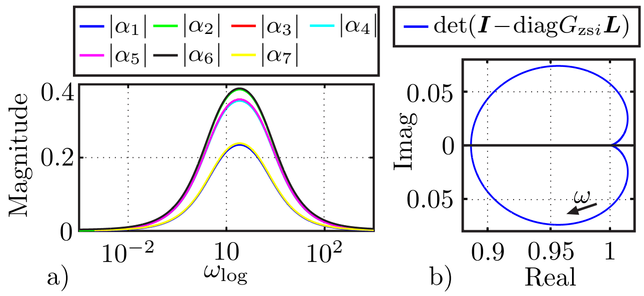

Figure 4: Stability test: a) with local model information, b) with global model information

Plug-and-play action. The activation of zone 8 at run time is considered. Initially, it is assumed that each controlled subsystem satisfies $\mathcal{A}_i$. The plug-in scenario is performed in the following two steps:

1. The controller of the new introduced subsystem has to be designed initially

$$\Sigma_{\mathrm{C}8} = \mathit{solve}(\Sigma_{\mathrm{S}8}, \Sigma_{\mathrm{R}8}, \mathcal{N}_8)$$

2. Since zone 8 is a new neighbour of zone 7, the condition $(\Sigma_{\mathrm{S}7}, \Sigma_{\mathrm{C}7}, \Sigma_{\mathrm{R}7}, \mathcal{N}_7) \in \mathcal{A}_7$ has to be checked. If the condition is violated, a reconfiguration of $\Sigma_{\mathrm{C}7}$ has to be performed.

References

[1] J. Lunze. *Feedback Control of Large-Scale Systems*, Prentice Hall, 1992

[2] S. Bodenburg and J. Lunze. Plug-and-Play Control - definition and realisation in MATLAB. *Automatisierungstechnik*, 61:487-494, 2013

[3] S. Bodenburg, S. Niemann and J. Lunze. Experimental evaluations of a fault-tolerant plug-and-play controller. In *Proceedings of the European Control Conference*, 2014, pp. 1945–1950.