Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-25 (2016) 318–323

Distributed Systems - A brief review of theory and practice

Stefan Misik * **Arben Cela** ** **Zdenek Bradac** ***

* *Department of Control and Instrumentation, Faculty of Electrical Engineering and Communication, Brno University of Technology, Brno, Czech Republic (e-mail: stefan.misik@phd.feec.vutbr.cz)*
** *Department of Computer Science and Telecommunication, Université Paris-Est, ESIEE Paris, Noisy le Grand, France (e-mail: arben.cela@esiee.fr)*
*** *Department of Control and Instrumentation, Faculty of Electrical Engineering and Communication, Brno University of Technology, Brno, Czech Republic (e-mail: bradac@feec.vutbr.cz)*

Abstract: Due to technological and economical reasons plants, manufacturing systems and networks are developed with an ever increasing complexity. Such high complexity systems are also called Large Scale Systems. The aim of this paper is to present a brief review of theory used to model specific class of distributed large scale systems. In second part of this paper also a practical example of modelling process and simulation of such system is presented.

Keywords: Large Scale Systems, Distributed Computer Control Systems, Modelling, Discretization, Distributed Models

1. INTRODUCTION

Dynamic systems divided into many component subsystems by sparsity of their inter-component connections are called distributed systems. As described in Shamma (2007), Scattolini (2009), and Riverso (2013), apart from the multitude of their components and limited inter-component connections, they are also characterized by information being sensed only locally at each component. This leads to distribution of information, when in contrast with centralized approaches, no component has access to the information gathered by all components.

Examples of such systems include mobile sensor networks, network congestion control and routing, transportation systems Papageorgiou and Kotsialos (2002); Bellemans et al. (2006), autonomous vehicle systems, distributed computation, power systems Saadat (1998); Riverso and Ferrari-Trecate (2012), micro-grids Etemadi et al. (2012); Bolognani and Zampieri (2013), building temperature models Ma et al. (2012); Oldewurtel et al. (2012), etc.

Due to technological and economical reasons plants, manufacturing systems and networks are developed with an ever increasing complexity, such system is shown in Fig. 1. These systems, also known as *Large Scale Systems*, often encompass many interacting subsystems and can be difficult to control using *classic control approaches* (Scattolini (2009)), which are mostly concerned only by input/output description of the dynamic systems Lewis (1992).

More *modern approach* in modeling and control is therefore suitable for above defined class of systems. This approach deals with time-domain state-space description (Lewis (1992)). Mathematical tool for expressing such models are usually vectors of first-order differential or difference equations for continuous-time or discrete-time systems respectively.

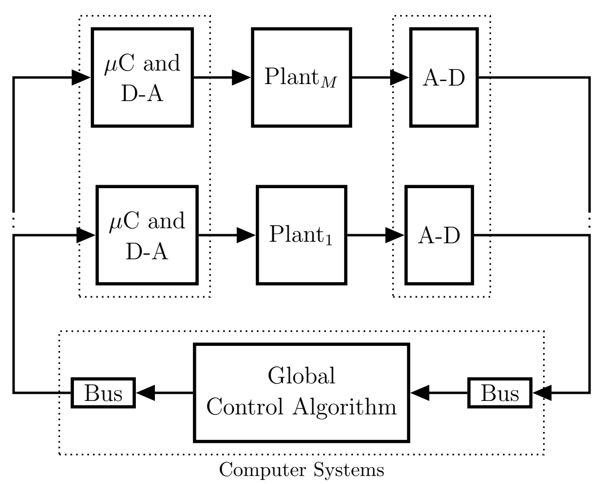

Fig. 1. Distributed computer control

The main aim of this paper is to present a review of the literature concerning distributed systems and large scale systems modelling and conjunction of these subjects. In the experimental part, an example of a large scale system model design is presented.

2. THEORETICAL PART

As mentioned in the introduction, to carry out design of the control in the discrete-time domain, a state-space mathematical model (1) comprised of first order vector difference equations are often used.

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.12.057

$$x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k) + Du(k) \tag{1}$$

where $k \in \mathbb{Z}^+$ is number of the sample and corresponds to the time kT_S, T_S is a sampling period, $x(k) \in \mathbb{R}^n$ is a vector of discrete-time internal states, $u(k) \in \mathbb{R}^m$ is a vector of discrete-time control inputs, $y(k) \in \mathbb{R}^p$ is a vector of discrete-time measured outputs, and $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, $D \in \mathbb{R}^{p\times m}$ are matrices whose elements specify the static and dynamic properties and behavior of the discrete-time plant model. Discrete-time dynamic systems describe either (I) inherently discrete systems, e.g. bank saving account balance at the k-th month, (II) Transformed continuous-time systems, while the latter is much more common case.

2.1 Large Scale Systems

Systems denoted by the name *Large Scale Systems* are often described as a result of many subsystems interacting through the coupling of physical variables or the transmission of information over a communication network. It is common to represent a LSS as coupling graph (Fig. 2), i.e. a directed graph where nodes represent subsystems and edges are couplings Riverso (2013). Couplings from subsystem i to subsystem j, in Fig. 2 represented by an arrow with label x_i, denote that, the dynamics of the j-th subsystem depend on the signal x_i. Also systems i and j are said to be in parent-child relation.

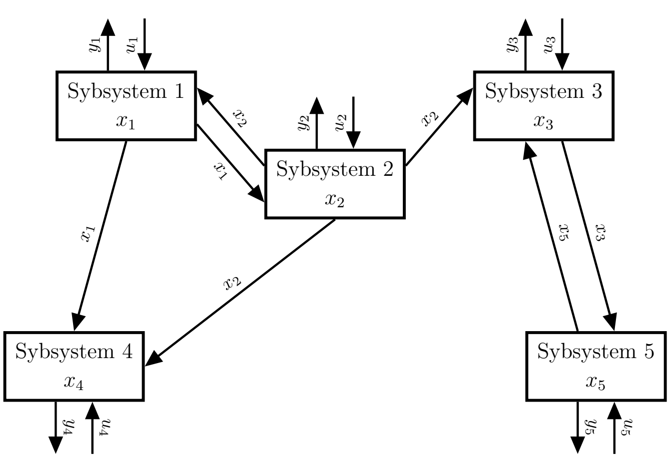

Fig. 2. LSS coupling graph

Interaction Modeling A considered LSS is comprised of M subsystems which together form whole system, which can be modeled by (2). System (2) is then partitioned into M low order interconnected non-overlapping systems Farina and Scattolini (2012) whose indices are contained in the set $\mathcal{M}$, i.e. $\mathcal{M} = \{i \in \mathbb{Z}^+; 1 \leq i \leq M\}$.

$$x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k) + Du(k) \tag{2}$$

Each decentralized node $i \in \mathcal{M}$ of the system (Fig. 2) is represented by the model of the subsystem Σ_i and its neighborhood $\mathcal{N}_i$, which is set of subsystem indices $\mathcal{N}_i = \{j \in \mathcal{M} : A_{ij} \neq 0, i \neq j\}$.

$$x_i(k+1) = A_{ii}x_i(k) + B_iu_i(k) + \sum_{j\in\mathcal{N}_i} A_{ij}x_j$$
$$y_i(k) = C_ix_i(k) + D_iu_i(k) \tag{3}$$

where $k \in \mathbb{Z}^+$ is number of the sample and corresponds to the time kT_S, T_S is a sampling period, $i \in \mathcal{M}$ and $j \in \mathcal{M}$ are subsystem indices, $x_i(k) \in \mathbb{R}^{n_i}$ is a vector of discrete-time internal states of the subsystem i, $u_i(k) \in \mathbb{R}^{m_i}$ is a vector of discrete-time control inputs of the subsystem i, $y_i(k) \in \mathbb{R}^{p_i}$ is a vector of discrete-time measured outputs of the subsystem i, and $A_{ij} \in \mathbb{R}^{n_i\times n_j}$, $B_i \in \mathbb{R}^{n_i\times m_i}$, $C_i \in \mathbb{R}^{p_i\times n_i}$, $D_i \in \mathbb{R}^{p_i\times m_i}$ are matrices whose elements specify the dynamic properties and behavior of the subsystem i. By comparing model of a subsystem (3) to a general model (1), it can be seen that in former a term with sum is added. This part of the (3) equation describes the coupling between subsystems.

From the (3) is clear that all the subsystem are input and output decoupled, as the B, C, D matrices of the overall system model (2) are block diagonal matrices, which is shown in (4).

$$A = \begin{bmatrix} A_{11} & \cdots & A_{1M} \\ \vdots & \ddots & \vdots \\ A_{M1} & \cdots & A_{MM} \end{bmatrix}$$
$$B = \begin{bmatrix} B_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & B_M \end{bmatrix}$$
$$C = \begin{bmatrix} C_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & C_M \end{bmatrix} \tag{4}$$
$$D = \begin{bmatrix} D_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & D_M \end{bmatrix}$$

2.2 Mixed Euler-ZOH Discretization

It is common that the models of LSS are developed in continuous time (i. e. models expressed as differential equations (5)), while in most cases control synthesis methods are designed for discrete-time systems. Especially for LSS, and in general *sparse systems*, the most recent synthesis methods based on methods such as Model Predictive Control are developed in discrete-time Scattolini (2009).

$$\dot{x}_c(t) = A_cx(t) + B_cu_c(t)$$
$$y_c(t) = Cx_c(t) + Du_c(t) \tag{5}$$

In (5) and (6) the matrices, vectors, sets and indices have similar meaning as in (2) and (3) respectively, the added subscript "$_c$" denotes that they are associated with the continuous-time system.

Similarly as a discrete-time LSS, at the beginning of the section, a continuous-time LSS (5) results from the interaction and/or coordination of a large number of interconnected subsystems (6). It is desirable that the discrete-time model, used to design controller(s), has the same