Chapter 1
Complex Systems and Control: The Paradigms of Structure Evolving Systems and System of Systems

Nicos Karcanias and Maria Livada

Abstract This chapter deals with two rather new notions of complexity emerging in Engineering Systems, reviews existing approaches and results and introduces a number of open problems defining a research agenda in the field. We examine these notions based on the fundamentals of a systemic framework and from the perspective of Systems and Control Theory. The two new major paradigms expressing forms of engineering complexity which have recently emerged are the new paradigms of *Structure Evolving Systems (SES)* and *Systems of Systems (SoS)*. The origin and types of complexity linked to each one of these families are considered, and an effort is made to relate these new types of complexity to engineering problems and link the emerging open issues to problems and techniques from Systems and Control Theory. The engineering areas introducing these new types of complexity are linked to the problems of *Integrated System Design* and *Integrated System Operations*.

1.1 Introduction

Complex Systems is a term that emerges in many disciplines and domains [9] and has many interpretations, implications and problems associated with it. The specific domain provides dominant features and characterizes the nature of problems to be considered. A major classification of such systems is to those linked with *physical processes* (physics, biology, genetics, ecosystems, social, etc.) and the artificial, which are *man-made* (engineering, technology, energy, transport, software, management and finance, etc.). We are dealing with *man-made* systems and we are interested in identifying generic types of system complexity among the different problem domains and then identify the relevant concepts and tools that can handle

N. Karcanias (✉) · M. Livada
School of Engineering and Mathematical Sciences, Systems and Control Research Centre, City, University of London, Northampton Square, London EC1V 0HB, UK
e-mail: N.Karcanias@city.ac.uk

M. Livada
e-mail: Maria.Livada@city.ac.uk

E. Zattoni et al. (eds.), *Structural Methods in the Study of Complex Systems*, Lecture Notes in Control and Information Sciences 482,
https://doi.org/10.1007/978-3-030-18572-5_1

the different types of complexity and then enable the design or redesign of complex systems–processes. There is a need to develop generic methodologies and tools that can be applied across the different problem domains. This research aims to identify Systems and Control concepts and tools which are important in the development of methodologies for the *Management of Complexity* of engineering-type complex systems.

Existing methods in Systems and Control deal predominantly with fixed systems, where components, interconnection topology, measurement–actuation schemes and control structures are specified. Two new major paradigms expressing forms of engineering complexity which have recently emerged are the new paradigms of

- *Structure Evolving Systems (SES)* [32]
- *Systems of Systems (SoS)* [23, 37, 50]

Using the traditional view of the meaning of the system (components, interconnection topology, environment), the common element between the first two new paradigms is that the interconnection topology may vary and evolve in the case of *SES*, whereas in the case of *SoS* the interconnection rule is generalized to a new notion of "systems play" [33] defined on the individual system goals. The paper deals with the fundamentals regarding representation, structure and properties of those two challenging classes, demonstrates the significance of traditional systems and control theory, and introduces a new research agenda for control theory defined by:

Structure Evolving Systems [32]: Such a class of systems emerges in natural processes such as Biology, Genetics, Crystallography [24], etc. The area of man-made processes includes Engineering Design, Power Systems under de-regulation, Integrated Design and Redesign of Engineering Systems (Process Systems, Flexible Space Structures, etc.), Systems Instrumentation, Design over the Life Cycle of processes, Control of Communication Networks, Supply Chain Management, Business Process Re-engineering, etc. This family deviates from the traditional assumption that the system is fixed and its dominant features, introducing types of system complexity related to the following:

- The topology of interconnections is not fixed but may vary through the life cycle of the system (*Variability of Interconnection Topology Complexity*).
- The overall system may evolve through the early–late stages of the design process (*Design Time Evolution*).
- There may be variability and/or uncertainty on the system's environment during the life cycle requiring flexibility in organization and operability (*Life Cycle Complexity*).
- The system may be large scale and multicomponent, and this may impact on methodologies and computations (*Large Scale—Multicomponent Complexity*).
- There may be variability in the Organizational Structures of the information and decision-making (control) in response to changes in goals and operational requirements (*Organizational Complexity Variability*).

The above features characterize a new paradigm in systems theory and introduce major challenges for Control Theory and Design and Systems Engineering. There are different forms of structure evolution. Integrated System Design has been an area that has motivated some of the early studies on *SES.* The integration of traditional design stages [28], such as Process Synthesis *(PS),* Global Instrumentation *(GS)* and finally Control Design *(CD)*, is an evolutionary process as far model system formation and two typical forms of evolution are the s*tructural design evolution*, the *early–late design evolution* and the *interconnection topology evolution* [32]. Methodologies and tools developed for *Fixed Structure Systems (FES)* cannot meet the challenges of the *SES* class and new developments on the level of concepts, modelling, analysis and synthesis methodologies are needed. The research is influenced by the need to address life cycle and redesign issues, and such problems have a strong technological and economic dimension.

System of Systems: The notion of "*System of Systems*" (*SoS*) has emerged in many fields of applications from air traffic control to constellations of satellites, integrated operations of industrial systems in an extended enterprise to future combat systems [23, 50]. Such systems introduce a new systems paradigm with main characteristic the interaction of many independent, autonomous systems, frequently of large dimensions, which are brought together in order to satisfy a global goal and under certain rules of engagement. These complex multisystems are very interdependent, but exhibit features well beyond the standard notion of system composition. They represent a synthesis of systems which themselves have a degree of autonomy, but this composition is subject to a central task and related rules defined as "system plays" [33] expressing the subjection of subsystems to a central task. This generalization of the interconnection topology notion introduces special features and challenging problems, which are different than those linked to the design of traditional systems in engineering. The distinguishing features of this new form of complexity are as follows [32]:

- The role of "*objects*" or "*subsystems*" of the traditional system definition is taken by the notion of the *autonomous agent,* and it is characterized by some form of intelligence. This is linked to the notion of "*integrated intelligent system*" defining an autonomous intelligent agent.
- The notion of "*interconnection topology*" of traditional systems is generalized to that of "*system*s play" which is expressed at the level of goals of autonomous intelligent agents [71].
- Decision-making and control are linked to the nature of the "*system*s play" which among other fields may be linked to cooperative control, game among the subsystems, etc.
- System organization (Hierarchical-Multilevel, Holonic [67], etc.) defines an internal form of system structure and this plays a central role in the characterization of the notion of *emergent properties*.

The problem of *Systems Redesign* has been only partially addressed in engineering as redesign of control structure in response to faults, and it has been an active area in business [65]. This problem may be considered within the framework of Integrated Systems Design and leads to problems in the *SES* area [32]. Understanding the issues linked to *SES* and *SoS* is critical in addressing the problem in its entirety from an engineering perspective. Addressing the issues of *SES* and *SoS* has important implications for the underpinning Control Theory and related Design methodologies. Control Theory and Design has developed considerably in the last 40 years. However, the underlying assumption has always been that the system has been already designed and thus control has been viewed as the final stage of the design process on a system that has been formed. The new paradigms deviating from the "*fixed system structure assumption*" introduce new challenges for Control Theory and Control Design. These force us to reconsider some of the fundamentals (viewing Control as the final design stage on a formed system) and create the need for new developments where Control provides the concept and tools intervening in the overall design process, even at stages where the system is not fixed but may vary, and may be under some evolution. Traditional Control has been capable to deal with uncertainty at the unit process level, but now has to develop to a new stage where it has to handle issues of structural, dynamic evolution of the system as well as control in the context of a "systems play". The paper aims to provide an overview of these new areas, deal with issues of representation, examine different forms of system evolution, define the relevant concepts and tools, provide a systems based characterization of *SoS,* and introduce a research agenda for these new paradigms. Integral part of the effort is the linking of these new challenges to well-defined systems and control concepts and methodologies.

The paper is structured as follows: Section 1.2 reviews the notions of the system and summarizes the emergent forms of complexity. In Sect. 1.3, we review the three major engineering problems which introduce types of complexity, that is, the problems of Integrated Design, Integrated Operations and Re-engineering, and identify the different types of systems complexity which will be the main subject of the subsequent sections. Section 1.4 deals with the evolution of models from the early to late design stages, different types of system evolution are considered and the problems associated with them are specified. We consider external and then internal system representations. We examine the notion of a Progenitor model and the derivation of models for control design. This is linked to a form of evolution where the input and output system dimensions are reduced and considered in Sect. 1.5. An alternative formulation based on internal descriptions, where a process graph is defined with fixed nodal cardinality and subsystem models of variable complexity, and or fixed dynamics of subsystems and variable nodal cardinality. The evolution of systems linked to the cascade design process is considered in Sect. 1.5. We consider an evolution type linked to system composition by design of the interconnection graph, and then additional types of evolution associated with the selection of sets of inputs and outputs, referred to here as "systems instrumentation". Within the latter category, we distinguish two distinct forms of evolution, the introduction of orientation in implicit models and the model projection problems. Section 1.6 deals with multidimensional system view linked to an integrated hierarchical structure and introduces system aspects related to the variable complexity and a different nature of subsystem models. We also provide a characterization of system and emergent properties for the system. The notion of *System of Systems* (*SoS*) is considered in Sect. 1.7. We review first the relative literature which provides an empirical definition of this notion. We then introduce the notion of the *Integrated Autonomous System* which is integral part of the new systemic definition for *SoS*. The crucial element of the new definition is the notion of the "*systems play*" and its characterization in terms of standard systems and control concepts and methods is considered. Finally, Sect. 1.8 provides the conclusions, which are in the form of a research agenda for such new families of complex systems.

1.2 The Notion of the System

The development of a systems framework for general systems is not a new activity [52]. Such developments have been influenced predominantly by the standard engineering paradigm. Addressing the variety of new paradigms emerging in man-made systems requires a further development of the standard notion [31]. We will reconsider existing concepts and notions from the general systems area, detach them from the influences of specific paradigms and generalize them appropriately to make them relevant for the new challenges. We use the following standard systems definition.

Definition 1.1 A *system* is an interconnection and organization of objects that is embedded in a given environment.

This definition is very general and uses as fundamental elements the primitive notions of *objects*, *connectivities–relations* (topology), and *environment,* and for man-made systems involves the notion of *system purpose, goal*. It can be symbolically denoted as in Fig. 1.1.

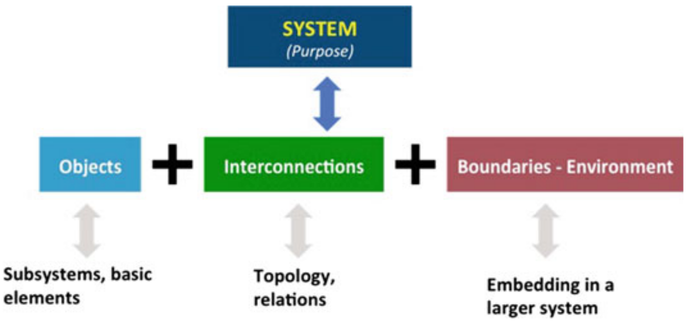

Fig. 1.1 The notion of the system

The notion of the *object* is considered to be the most primitive element, or a system and this allows us to use it in any domain. We define the notion of the object as:

Definition 1.2 An *object*, *B,* is a general unit (abstract, or physical) defined in terms of its attributes and the possible relations between them.

Remark 1.1 This definition of a system is suitable for the study of "soft", as well as "hard" systems and it is based on a variety of paradigms coming from many and diverse disciplines. It refers essentially to *simple systems* since issues of internal organization are reduced only to the interconnection topology. Systems with internal organization will be referred to as *integrated systems* and they will be considered in the following section. These definitions do not make use of notions such as causality, input–output orientation, definition of goal, behaviour, and so on. Quite a few systems do not involve these features, and thus they have to be introduced as additional properties of certain families.

A more explicit description of the notion of the system that involves some form of orientation and which also describes the basic signals is given in Fig. 1.2 where the basic variables are also included. These are the control inputs $\underline{u}$, the outputs $\underline{y}$, the internal variables $\underline{z}$, the input connections $\underline{e}$ and output connections $\underline{w}$. Note that input and output influences are the result of the given system being embedded in a larger system; $\underline{v}$ may also represent disturbances. For composite systems having μ subsystems $S_{a,j}$ we denote by $d_{v,j}$, $d_{q,j}$ the dimensions of the input and output influences of $S_{a,j}$; then μ will be referred to as the *order* and $\left\{\left(d_{v,j}, d_{q,j}\right), j = 1, ..., \mu\right\}$ as the *cardinality* of the order composite system.

Issues of complexity are naturally connected with the above description and they may be classified in the following categories:

- Objects, Subsystems nature and their variability

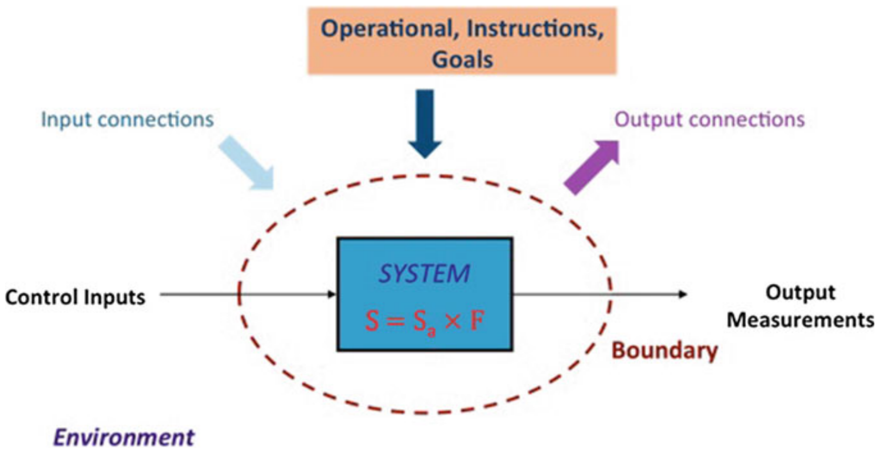

Fig. 1.2 The notion of the system with the basic variables

- Interconnection topology variability (variability of order and cardinality)
- Internal System Organization (non-simple systems)
- Embedding the system to a larger system
- System Design and Redesign
- System Operations
- System Dimensionality
- Support activities related to Data, Information and Computations
- Uncertainty in system description

Central to all above categories of system complexity are issues of system variability due to different types of evolution. The paper is considering the different types of evolutionary processes described above.

1.3 Integrated Design and Operations

The problem of system integration in engineering systems is a technological challenge, and it is perceived by different communities from different viewpoints. Systems Integration means linking the different stages of systems design in the shaping of the system, relating the functions of system operations and establishing a framework where operational targets are translated to design tasks. This problem has been treated mostly as a software problem, and the multidisciplinary nature of the problem (apart from software and data) has been neglected. The significance of integration has created some urgency in working out solutions to difficult problems and this has led to the development of interdisciplinary teams empowered with the task to create such solutions. The key issue here is the lack of methodology that bridges disciplines and provides a framework for studying problems in the interface of particular tasks. The problem of integrating design has been considered in [22, 28, 63]. Recent developments in the area of hybrid systems [5], new developments in the area of organization and overall architectures [67] contribute to the emergence of elements for the integration of system operations. There are, however, many more aspects of the effort to develop a framework of integration which are currently missing. A general view of manufacturing systems involves the following [22]:

1. System Design Issues
2. Operational Issues–Signals and Operations
3. Business Activities
4. Vertical Activities–Data, IT, Software

The diagram indicates a natural nesting of problem areas, where design issues provide the core, linked with the formation of the physical process that realizes production. Production-level activities take place on a given system, they are mostly organized in a hierarchical manner and they realize the higher level strategies decided at the business level. Vertical activities are issues going through the Business–Operations–Design hierarchy and they have different interpretations at the corresponding level. The

Physical Process Dimension deals with issues of design–redesign of the Engineering Process and here the issues are those related to integrated design [8, 22, 28, 49, 57, 58]. The Signals, Operations Dimension is concerned with the study of the different operations, functions based on the Physical Process and it is thus closely related to operations for production. In this area, signals, information extracted from the process are the fundamentals and the problem of integration is concerned with understanding the connectivities between the alternative operations, functionalities and having some means to regulate the overall behaviour. Both design, operations and business generate and rely on data and deploy software tools, and such issues are considered as vertical activities. Compatibility and consistency of the corresponding data structures and software tools express the problem of software integration.

The operation of production of the types frequently found in the Process Industries relies on the functionalities, which are illustrated in Fig. 1.3. Such general activities may be grouped as [22] (i) Enterprise Organization Layers, (ii) Monitoring functions providing information to upper layers and (iii) Control functions setting goals to lower layers. The process unit with its associated Instrumentation are the primary sources of information. However, processing of information can take place at the higher layer. Control actions of different nature are distributed along the different layers of the hierarchy.

The main layer of technical supervisory control functions involves [22, 58]: Quality Analysis and Control; State Assessment, Off Normal Handling and Maintenance; Supervisory control and Optimization; and Identification, Parameter Estimation, Data Reconciliation. These are of supervisory nature activities and refer to the process operator. The automated part of the physical process refers to Process control and involves [22, 58] Regulation, End Point and Sequence Control; Emergency Protection; and Process Instrumentation and Information System.

It is apparent that the complexity of operating the production system is very high. A dominant approach as far as organizing such activities is through a Hierarchical Structuring [53] considered here. However, other forms of organization have emerged [67], but their full potential has not yet been explored. The study of Industrial Processes requires models of different types. The borderlines between the families of Operational Models (OM) and Design Models (DM) are not always very clear and frequently the same model may be used for some functions. Handling the high complexity of the overall system is through aggregation, modularization and hierarchization [8], and this is what characterizes the overall OPPCP structure described in Fig. 1.3. The production system may be viewed as an information system, and thus notions of complexity are naturally associated with it [49].

It is clear that for engineering-type problems the notion of the system emerging is more elaborate than the notion of the *simple system* introduced in the previous section. Systems produced as results of design with operations expressing the functionalities related to the system goal may be referred to as *integrated systems*. Such systems have the design process linked to the physical (engineering) process and an internal organization referred to the different operational functionalities, and all these are supported by signals and data. The *integrated system* has forms of complexity which may be classified as

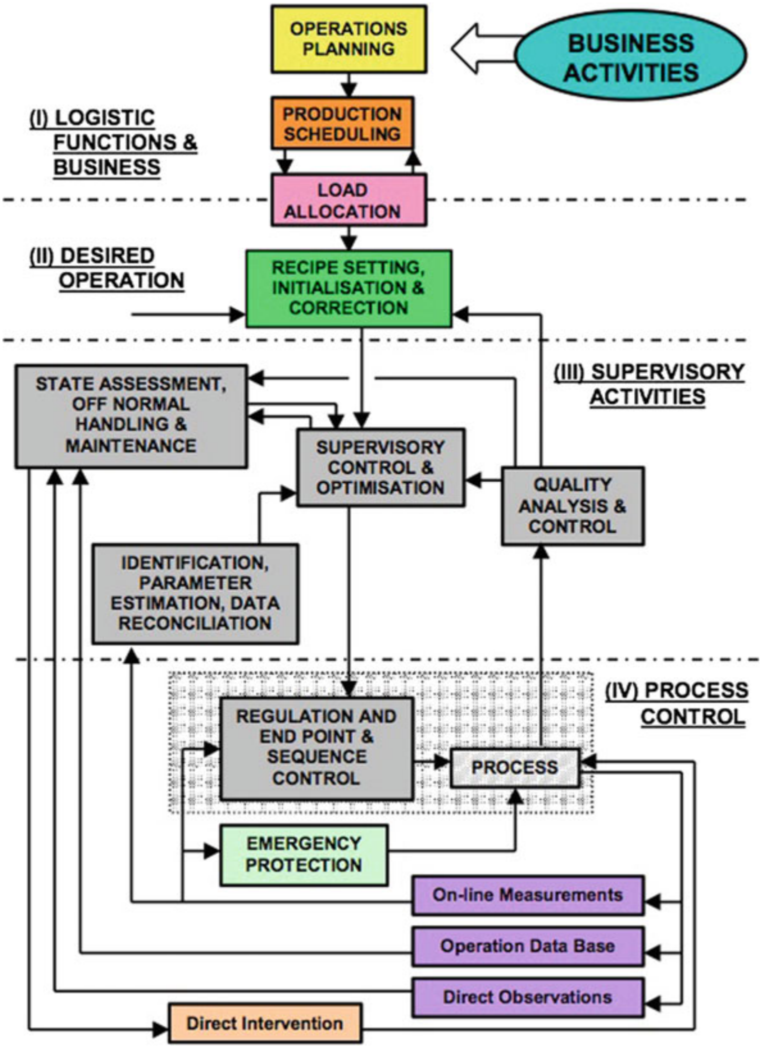

Fig. 1.3 System and its operational hierarchy © [2011] IEEE. Reprinted, with permission, from [22]

1. Integrated Design types of complexity
2. System organization types of complexity
3. System of Systems type of complexity
4. System Re-engineering types of complexity

Note that engineering design is an iterative process and we may distinguish *early stages of design* and *late stages of design* [32]. The transition from early to late design is expressed by models of variable complexity, and this introduces a notion of model embedding with associated complexity. Furthermore, design is a cascade process involving as distinct stages process synthesis, systems instrumentation and finally control design. The transition from one stage to the next expresses a specific form of evolution-type complexity. The large dimensionality (multicomponent nature) of the system refers to the physical system, and this introduces forms of complexity related to design and computations. The system organization is hierarchical or otherwise involves linking functionalities and corresponding models of different nature, and this introduces new forms of complexity with a number of new challenges for Systems and Control theory [22]. The establishing of links between integrated systems at the level of *goals* leads to a new type of complexity referred to as *System of Systems* (*SoS*) [23, 33, 50]. Re-engineering refers to changing the physical system and/or operational processes of an existing system and thus forms of complexity related to both design and operations emerge. These types of complexity are considered subsequently.

1.4 Integrated System Design and Model Complexity Evolution

1.4.1 Integrated Design

The process of overall design of a system is an iterative process [28] (described in Fig. 1.4) which is based on the following design stages:

- Process Synthesis
- Systems (Global) Instrumentation
- Control Design

These three design stages have a cascade nature with feedback loops between the various substages leading to the final structure. Process Synthesis describes the interconnection of the processing units, Systems Instrumentation deals with the problem of selecting the appropriate inputs and outputs and Control System Design is then performed on the final system model. There is an evolutionary process expressed as model shaping during the first two stages which also implies an evolutionary process on the structural properties linked to the final composite system model, which will be used for Control System Design. The Iterative nature of the design process implies that there is an evolutionary process of moving from simple to more complex system descriptions characterized by models of variable complexity. Assuming that the interconnection topology is fixed throughout the design the evolution process is characterized by Early and Late stages. At the *early stages*, simple modelling is required for subprocesses and physical interconnections, and at *late stages* of design, there is need for more detailed, full dynamic models for both subprocesses and physical interconnection structures. Describing the transition from simple modelling to full dynamic models that would enable the study of Systems and control proper-

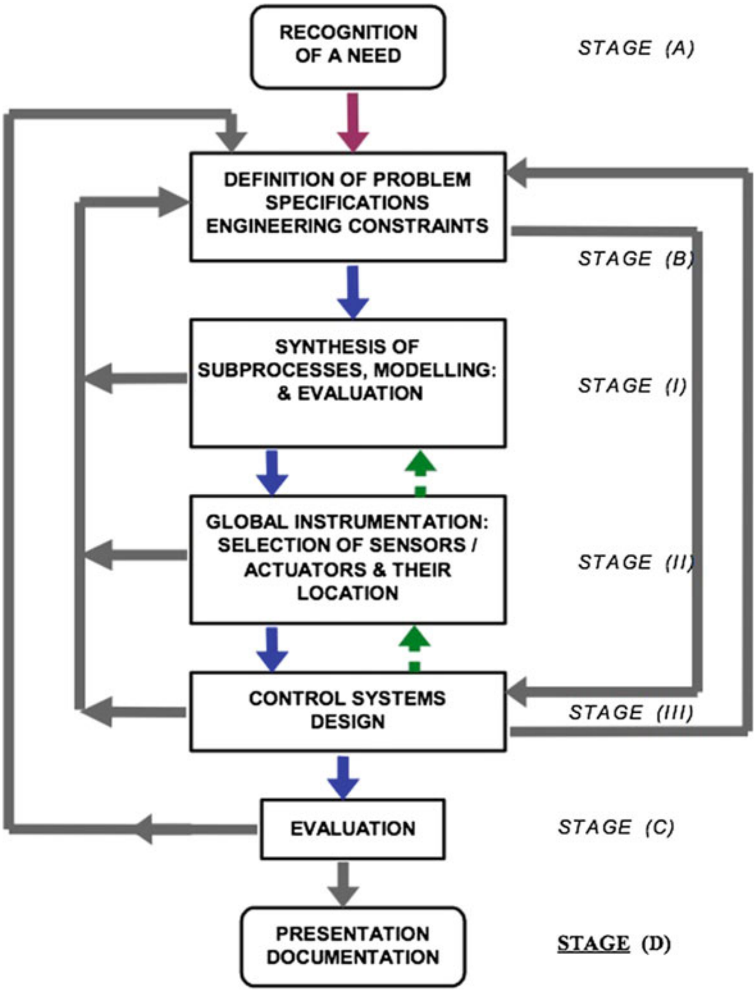

Fig. 1.4 Engineering design process

ties with regard to evolution is challenging. By keeping the same generic topology, the evolution process will develop models of increasing complexity for the subprocesses expressing the embedding of sequential processes and leading to a nesting of system models. An alternative assumption is to consider variations in the topology expressed by dimensional variability of system nodes and/or of the corresponding system cardinality. We distinguish the following types of complexity:

(i) Evolution of models from Early to Late Design stages, referred to as *Design Time Evolution* with *Fixed order* and *Cardinality.*
(ii) Evolution of models from Early to Late Design stages, referred to as *Design Time.*

(iii) Evolution that is naturally linked to the cascade nature of the design process and referred to as *Cascade Design Evolution.*
(iv) Evolution associated with growth and partial death of parts of the system, expressed as *Variable Order* and referred to as *Life Cycle Evolution.*

The study of such processes provides the basis for studying the evolution of system structure and related properties under the different evolutionary mechanisms. Assuming that the interconnection topology is fixed, there are two fundamental aspects of evolution in design, namely, the case of *Fixed Order* and *Fixed Cardinality* linked to the notion of *Dynamic Complexity Evolution* and the case of *Fixed Order,* but *Variable Cardinality*, linked to *Dimensional Complexity Variability* in "early–late" design. These evolution types are considered next.

1.4.2 Early–Late Design Models: The Family of Fixed-Order Models

For a fixed-order composite system at the Early Stages, all the subprocesses and the physical interconnections are represented with simple models, whereas at the Late Stages of design more detailed and may be full dynamic models are required for both subprocesses and physical interconnection structures. This process leads to the notion of *Dynamic Complexity Variability* in the design process. Modelling requires a framework that permits the transition from simple graphs to full dynamic models and allows study of Systems and Control properties in a unifying way. The process that generates families of models has as the simplest element the *Conceptual Model* of the process [15]. *Conceptual Modelling* is being used during the very first steps to translate all the Requirements and Objectives into sets of *Preliminary Designs* leading to the notion of *Conceptual Process Model*.

For composite systems with a fixed order and given cardinality (fixed or variable), the most elementary conceptual model is denoted by $\mathscr{M}_0^c$ [58] and acts as the generator for subsequent models of variable complexity [32]. Every stage of evolution defines a new model which is the successor of the previous one, and it has higher complexity from the previous stage model (for Linear Systems we use the McMillan degree as a measure of complexity). The overall set that contains all such models will be denoted by $\mathscr{M} = \left\{\mathscr{M}_i^c, i = 0, 1, ..., k\right\}$, where k represents the k-th stage of evolution and it is referred to as the *early–late design model set.* This evolutionary process expresses a nesting of models and the simplest model in the chain $\mathscr{M}_0^c$ is referred to as the *basic kernel model* of the chain. There is a need to develop a framework that enables the transition from simple modelling to full dynamic models and allows the study of Systems and Control properties under this form of evolution. Every model $\mathscr{M}_i^c$ in the chain defines a graph which is affected by the cardinality of the subsystems and the description of the physical interconnection streams. The notion of a graph associated with a composite system is defined as the *kernel graph* *model*, and it is the simplest representation of systems of a given order and with cardinality that may be also fixed, or variable. This is defined below.

Definition 1.3 Let us consider a composite system S_C of order μ and correspond to every subsystem S_i a pair of vertices $\left(\underline{e}_i, \underline{w}_i\right)$, denoting *input connections* and *output connections*, respectively, and denote by g_i an edge providing an $\underline{e}_i$input-$\underline{w}_i$output description of S_i. If f_{ik} denotes the physical/ information streams connecting the $\underline{w}_i$ and $\underline{e}_k$ vectors, then the set } will be called the *kernel graph* $\mathscr{J}_0$ of the system.

For fixed-order composite systems, we use the kernel model as a starting point in the effort to develop models of increasing complexity, generated from the same $\mathscr{M}_0$ model. There are two cases to distinguish:

(i) fixed cardinality;
(ii) variable cardinality.

In both cases, we use the kernel graphindexgraph!kernel, succeeded by models with increasing complexity for the subprocesses. The chain $\mathscr{M}$ is generated by the *basic kernel model* $\mathscr{M}_0$ which in turn generates a nested sequence of models where $\mathscr{M}_1$ evolves from $\mathscr{M}_0$, $\mathscr{M}_1$ generates $\mathscr{M}_2$ and this procedure goes on, where $\mathscr{M}_{0,nl}$ denotes the simplest nonlinear model.

The derivation of nesting chain is not a simple process, and there is no unique generic procedure for its construction. Specific applications define the chain of models from simple- to complex-based system models based on the knowledge of the particulars of the application. Developing, however, a generic framework requires techniques for generating such chains of models. There are two alternative approaches for such derivations. Both start with a full dynamic model. The first is using alternative methods for model reduction, and the second deploys the theory of partial realization [4, 25] to generate the chain of models. Chains based on model reduction depend on the specific technique used. A technique based on the partial realization for the derivation of chains is generic process and independent from the particulars of model reduction methodologies.

All methods used to generate chains preserve the *kernel model*. Such approaches generate the following sequences of models:

$$\mathscr{M}_0 \subset \mathscr{M}_1 \subset \mathscr{M}_2 \subset \ldots \subset \mathscr{M}_{k+1} \subset \mathscr{M}_{k+2} \subset \ldots \tag{1.1}$$

where $\mathscr{M}_0$ is the kernel model, $\mathscr{M}_1$ is the linear steady-state model, $\mathscr{M}_2$ corresponds to first-order dynamics, and subsequently we increase the complexity up to the most complex linear. The process may continue to the nonlinear case with the use of different complexity Volterra models. $\mathscr{M}_k$ may denote the first Volterra model, $\mathscr{M}_{k+1}$ the second Volterra description, etc. It is very important to note that there is a reversibility between Model Complexity Evolution and Model Simplification Approach. Model Evolution and Model Reduction may become completely reverse processes, if we use systems fixed order and cardinality. The generation of the chains defined in Definition 1.3 leads to three different basic types of model evolution based on the assumptions:

(i) Fixed order and cardinality and variability of complexity of subsystem models.
(ii) Fixed order, variable cardinality and fixed complexity of subsystem models.
(iii) Fixed order, variable cardinality and variability of complexity of subsystem models.

Clearly, the above cases may be also extended to the case of variable order, which is an issue considered later on. The study of system properties under such forms of evolution requires a description of the composite system at each design stage under the assumptions made for the order and cardinality. Thus let us assume that

$$\mathscr{M}_a^j = \left\{\mathscr{M}_i^j, i = 1, 2, ..., \mu\right\}$$

is the aggregate of the models of the subprocesses of the j-design stage and let $\mathscr{J}^j$ be the interconnection graph defined under the given order and cardinality assumption. The composite system is then defined by

$$\mathscr{M}^j = \mathscr{J}^j * diag\left\{\mathscr{M}_i^j, i = 1, ..., \mu\right\}.$$

The study of system properties requires a representation of the composite system model $\mathscr{M}^j$ which is a issue that will be considered in a subsequent section.

1.4.3 Early–Late Design: Model Complexity Evolution

1.4.3.1 Fixed-Order and Cardinality Systems

We consider a single system (order 1) and with fixed cardinality. The description of a linear system in terms of the infinite Laurent expansion provides a natural way of deriving approximations of variable complexity by truncation of the infinite series. This natural way of introducing models of variable complexity is linked to the classical problem of partial realization [4, 25]. It is assumed that the information available about a system S is an infinite sequence $(H_k)_{k=1}^{\infty} = \left(CA^{k-1}B\right)_{k=1}^{\infty}$, where $CA^{k-1}B$ are the *Markov parameters*. This input–output information is being used for the realization of a system (A, B, C) that would match only the first v terms of the infinite sequence. This realization is called partial realization. The partial realization establishes families of linear systems of variable dynamic complexity, and this is why our attention is now focused on looking at this classical problem from a different perspective, that is, the evolution in the family of models established by the partial realization [21].

We consider a rational transfer function $G(s) \in \mathbb{R}^{m\times p}(s)$ with a Laurent expansion:

$$G(s) = \sum_{i=1}^{\infty} H_i s^{-i} = H_1 s^{-1} + H_2 s^{-2} + ... \tag{1.2}$$

which defines an infinite sequence $\mathscr{H} = (H_1, H_2, ...)$ where the H_i's are real matrices. Taking the first ν $(\nu > 0)$ terms of Eq. (1.2), we have a finite sequence $\{H\}_\nu \equiv (H_1, H_2, ..., H_\nu)$. A natural way to approximate the rational matrix $G(s)$ is to define a new infinite power series:

$$G^{'}(s) = \sum_{i=1}^{\infty} H_i s^{-i} = H_1 s^{-1} + H_2 s^{-2} + ... + H_\nu s^{-\nu} + H_{\nu+1} s^{-\nu-1} + ... \quad (1.3)$$

with the first ν coefficients of the above power series being the corresponding H_i of $\{H\}_\nu \equiv (H_1, H_2, ..., H_\nu)$ of the original sequence and the remaining infinite number of terms $\left(H^{'}_{\nu+1}, H^{'}_{\nu+2}, ...\right)$ being dependent on the finite sequence $(H_1, H_2, \ldots H_\nu)$ in some appropriate way that will be defined later on [66]. Such an extension of the finite sequence will be referred to as proper extension and the mechanisms of achieving this are based on the principle of not increasing the McMillan degree of the sequence. This type of approximation is linked to the problem of partial realization [2, 66] and provides a natural way to define models of variable complexity for rational transfer function. In the following, we shall refer to $\mathscr{H} = (H_1, H_2, \ldots)$ as the parent series, the finite sequence $\{H\}_\nu \equiv (H_1, H_2, \ldots H_\nu)$ as the generator set and the infinite sequence based on $\{H\}_\nu$ which has been appropriately extended to $\mathscr{H}_\nu{}' \equiv (H_1, H_2, \ldots H_\nu, H'_{\nu+1}, H'_{\nu+2}, ...)$ with the $(H'_{\nu+1}, H'_{\nu+2}, \ldots)$ as linear functions of $\{H\}_\nu$ as a proper extension of $\{H\}_\nu$.

There always exist triplets of matrices $S \triangleq (A, B, C)$ with $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times p}$, $C \in \mathbb{R}^{m\times n}$ [25] such that

$$CA^{i-1}B = H_{i,}\ i = 1, 2, \ldots, \nu \text{ and } CA^{i-1}B = H_i\text{'}\ i = \nu + 1, ... \quad (1.4)$$

Such a triplet S will be called a realization of $\{H\}_\nu$, and the number n defines its *dimension*. Every finite sequence $\{H\}_\nu$ has a realization in the sense defined by (1.4), and the construction of such realizations is based on the rank properties of Hankel matrices [6]. Of all possible realizations of a finite sequence $\{H\}_\nu \equiv (H_1, H_2, \ldots H_\nu)$, there is a family with minimal dimension δ_ν defined by the maximal rank value of the sequence of Hankel matrices constructed from $\{H\}_\nu$, called the *McMillan degree* of $\{H\}_\nu$. A realization of dimension n equal to δ_ν is called *minimal* [4].

Definition 1.4 ([4]) A *realization* $S_\nu \triangleq (A_\nu, B_\nu, C_\nu)$ of $\{H\}_\nu \equiv (H_1, H_2, \ldots H_\nu)$ based on the proper extension, i.e., the infinite sequence

$$\mathscr{H}_\nu{}' \equiv (H_1, H_2, \ldots H_\nu, H'_{\nu+1}, H'_{\nu+2}, ...)$$

with dimension n and McMillan degree δ_ν is called a *partial realization* of the finite sequence $\{H\}_\nu$. If $n = \delta_\nu$, then is a *minimal partial realization* (MPR) $\{H\}_\nu$.

The process of considering $\{H\}_\nu$ finite sequences of the infinite sequence $\mathscr{H}_\nu{}'$ for varying values of ν gives rise to a family of systems

$$\{S\} = \{S_\nu : S_\nu \triangleq (A_\nu, B_\nu, C_\nu), \nu = 1, 2, ...\}$$

with corresponding transfer functions

$$\{G\} = \{G_\nu(s) : G_\nu(s) \triangleq C_\nu(sI - A_\nu)^{-1}B_\nu, \nu = 1, 2, ...\}.$$

The study of the properties of such sets is central to the effort to understand evolution of complexity in these chains. The construction of such systems linked to the finite sequences $\{H\}_\nu$ is considered in [2, 10, 66].

The parametrization of the minimal partial realizations is crucial in characterizing the structure evolution. The family of MPRs is characterized by the non-decreasing property of their McMillan degrees [25]. In fact, as we progressively include more and more terms into $\{H\}_\nu \equiv (H_1, H_2, \dots H_\nu)$, the following cases may occur: $\{H\}_\nu \equiv (H_1, H_2, \dots H_\nu)$ gives rise to $S_\nu \triangleq (A_\nu, B_\nu, C_\nu)$, $\{H\}_{\nu+1} \equiv (H_1, H_2, \dots H_{\nu+1})$ gives rise to $S_{\nu+1} \triangleq (A_{\nu+1}, B_{\nu+1}, C_{\nu+1})$ and $\delta_\nu \leq \delta_{\nu+1}$. If an inequality holds at a given position i of the infinite sequence, then this is called *jump point* [4, 25]. In the case, where the McMillan degree of $\{H\}_\nu$ and $\{H\}_{\nu+1}$ remains the same, there are again two different types of $H_{\nu+1}$ extensions: First, the case where the inclusion of $H_{\nu+1}$ results to a different (than the previous) realization of the same McMillan degree and second the case where the inclusion of $H_{\nu+1}$ produces exactly the same realization of the same McMillan degree [4, 25]. An alternative representation of the MPR family is provided by a continued fraction decomposition of rational transfer functions introduced in [2], where a formal power series $G(s) = H_1 s^{-1} + H_2 s^{-2} + \cdots$ a decomposition is introduced having an interpretation as a feedback interconnection of linear systems as shown in Fig. 1.5:

The construction and properties of the above family of linear systems are described in [2] and summary of their properties are as follows:

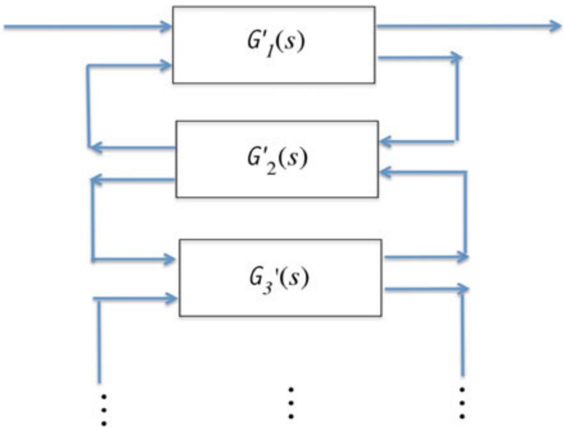

Fig. 1.5 Partial Realization as feedback interconnection of linear systems. Reprinted from [3], Copyright 1987, with permission from Elsevier

(i) There is a one-to-one correspondence between $\{H\}_\nu$ and the ν-th subsystem of the decomposition (some subsystems might be trivial; the interconnection of the first blocks defines a partial realization of $\{H\}_\nu$.
(ii) The decomposition defines a partitioning of the MacMillan degree of the j-th block, as well as the reachability and observability indices of the j-th subsystem.

The properties of the family of MPRs of a given rational transfer function are central in the study of evolution of structure for these families and it is a challenging issue. Results in this area such as those in [10] are linked to input–output canonical form, and they introduce a parametrization of the set of MPRs establishing links to the row, column Kronecker invariants. The stability properties of MPRs are considered in [11] and demonstrate that not all MPRs preserve important system properties. Results presented for the family of MPRs for a single rational matrix may be transferred to the case of composite systems [29], and this will be elaborated in the following section.

An alternative process for generating families of models with variable complexity has been introduced in [32, 39] and relates to the handling of classes of small numbers on system properties. This classification introduces a numerical form of model nesting and removing small numbers is a form of Robust Structural Simplification, and the derived family of models is referred to as numerical nesting. The derivation of this family is driven by the need that the structural properties of the original system have to be close to those of the reduced system.

Open Issues: The Development of the family of MPRs introduces a nesting of approximate models ordered by their degree of complexity (MacMillan degree). Among the issues related to the characterization of evolution of structure concern,

(i) Investigate whether key system properties of the original system (stability–instability, minimum–non-minimum phase) are preserved within the chain.
(ii) Determine the degree of complexity required to preserve key system properties.
(iii) Characterize the process of evolution of the Kronecker invariants (column, row minimal Indices, and finite and infinite elementary divisors) in the chain of models (the results in [10] deal only with controllability and observability indices).
(iv) Development of the properties of the family of numerically nested models.

1.4.3.2 Fixed-Order and Variable Cardinality Systems

For many engineering processes, the interconnection topology represents “natural flows” or “information flows”, referred to as *flow streams,* between the subsystems. The assumption that the dimensionality of these flow streams may vary as we move from early to late design is quite natural. In fact, in the case of a process system, a connection between two subprocesses may be defined in terms of a liquid flow; at early stages, this flow may be expressed in terms of liquid flow, but at later stages other properties such as temperature, pressure, etc. may be included. The number

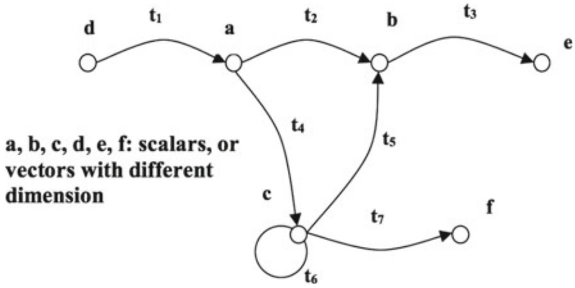

Fig. 1.6 Example of graph dimensional variability

of variables used to represent a system "flow" in natural streams may vary, and this implies a dimensional expansion of them with a subsequent expansion of the related models for the subprocess. This represents a *Dimensional Complexity Variability* from early to late design and corresponds to the case where the order of the composite system is fixed, but we have a variability in the cardinality of subsystems. Thus, although the interconnection rule (kernel graph) may remain the same, the local flows (edges in the graph) may change from scalars to vectors. This expresses an evolution from scalar to vector graphs with a respective evolution of scalar to matrix transfer functions (for the linear case). The nature of the corresponding graph will depend on the stage of the design (Early, Late), and this affects the models of local processes and the description of the physical interconnection streams. In the early stage, the graph consists of the fundamental variables linked to the physical interconnection and contains the minimal interconnection information (Fig. 1.6).

Example 1.1 Consider the directed graph where the corresponding nodes are vectors which according to degree of modelling may have an increasing, or shrinking dimension and thus the corresponding transmittances are also vector transmittances.

The kernel graph model defines a primitive form of the structure that derives from the conceptual modelling of the system, contains the basic information regarding subsystems and physical streams, and provides the minimal information about the physical interconnection topology. At later stages, as the requirements for modelling are increasing, more than one variable is associated with the physical streams and consequently the dimensionality of physical interconnection streams is changing. This defines the variability from one-dimensional to many-dimensional vertices and edges referred to as *Dimensional Complexity Variability of Graphs*.

Open Issues: Fundamental issues related to the dimensional variability of the graph related to the classification of the properties of the directed graph, which are independent of the dimensionality of nodes and those which depend on their dimensionality. Extending the Dimensional Complexity Variability by including the dynamic complexity of the subsystems raises additional issues related to structure evolution and related properties.

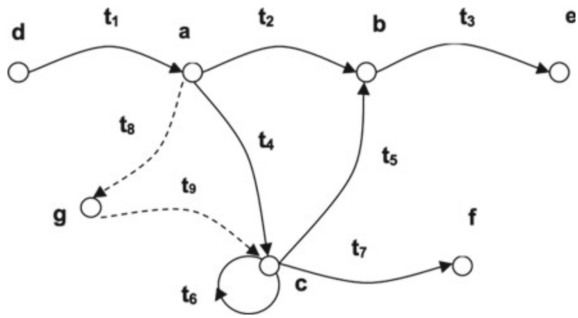

Fig. 1.7 Example of Structural Graph Growth problem. Reprinted from [32], Copyright 2008, with permission from Elsevier

1.4.3.3 Variable Order and Fixed/Variable Cardinality Systems

In engineering problems, the case of system growth or reduction of the system may occur. This involves expansion of the existing system by addition of parts (growth), or removal of parts (death) of the system. These cases lead to a variability of the original system order, emerge in problems of system redesign and are referred to as *graph structural evolution problems*. For such problems, the main interest is the study of evolution of structural and non-structural properties under such transformations. An illustration of the above is provided by the following example (Fig. 1.7).

Example 1.2 Consider the directed graph below represented by the a, b, c, d, e, f nodes and the solid edges. This is modified by adding a new node g and the new dotted line edges and produces an evolution of the previous graph.

The *Structural Graph Growth Problem* introduced here may be combined with the signal, dimensional growth, or *Dimensional Graph Evolution* discussed before. Clearly, combinations of the two may be considered and this may be referred to as the *General Graph Growth Problem*.

Open Issues: The above represent open areas for research on the fundamentals of *Graph Growth–Death.* Some important problems in this area involve the following:

(i) *The representation problem*: Define an appropriate modelling framework for describing the graph augmentation and graph reduction/death problems.

(ii) *The graph structural growth problem*: Investigate how the properties of a directed graph and algebraic invariants (based on given complexity models) are changing by addition of new nodes, or elimination of existing ones.

1.5 Cascade Design System Evolution

The technological stages of the overall system design have been described in Fig. 1.4. The Design is a cascade and complex process which is reminiscent of an evolution process that involves many different forms of system structure evolution. This evolution has many different features. A natural evolution of the system structure is that shaped through the design stages from conceptualization, to process synthesis, global instrumentation and finally control design, and this is referred to as cascade structural evolution. The system used for control design is the evolution of earlier forms shaped through the process synthesis and the systems instrumentation and its different aspects are considered here.

1.5.1 Systems Composition and Complexity

Process or System Synthesis is an act of determining the optimal interconnection of subsystems, as well as the optimal type and design of the units, subsystems within the overall system. The development of a generic synthesis framework that transcends the different application areas is a significant challenge. The modelling of composite systems using energy considerations is examined in [62, 70], and the traditional network synthesis is examined in [64]. The case of process systems is considered in [55]. Here, we consider the linear case and aim to present the evolution from the aggregate to the composite as a function of the interconnection topology. A crucial problem in system synthesis is the *Representation Problem* which is crucial for the study of structure evolution. The results here are based on the reduction of system synthesis to an equivalent feedback design problem using the standard composite system description [14, 28] and its particular characteristics based on the nature of the physical interconnection streams and the selection of the local input and output structure which leads to the notion of *completeness* [28] and providing a representation of the synthesis as generalized feedback design problem.

Let us consider a set of systems $\{S\} = \{S_j, j = 1, 2, .., \mu\}$ of order μ where every subsystem S_i has a pair of vertices $(\underline{e}_i, \underline{w}_i)$ and cardinalities $\{(d_{v,j}, d_{q,j}),$ $j = 1, ..., \mu\}$. We will assume that the subsystems have models $\{M\} = \{M_j, j = 1, 2, .., \mu\}$ of a certain type and if $\mathscr{F}$ is the interconnection rule (described by a graph and the subsystem cardinalities), then $S_a = S_1 \oplus S_2 \oplus \ldots \oplus S_\mu$ denotes the *aggregate system* with a model $\{M_a\} = block - diag\{M_j, j = 1, 2, .., \mu\}$. The *Composite System* is denoted by $S_c = \mathscr{F} * S_a$, where $*$ denotes the action of the interconnection topology $\mathscr{F}$ on S_a. The definition of Composite Systems involves the specification of the physical input and output streams and the selection of inputs and outputs at the subsystem level. Subsystems enter the composite structure, by interconnecting local variables (subsystem connecting inputs, outputs), and this affects drastically the overall properties of the composite system. A first attempt to link model composition to feedback was made in [14] and subsequently developed in [29]. The definition of the composite from the aggregate by the action of the interconnection topology raises important questions, which are linked to

(i) The representation of the composite system;

(ii) The relationships between the structure and properties of the aggregate and the composite in terms of the characteristics of the interconnection topology.

The general scheme that is considered satisfies certain assumptions which are described below [29, 32].

(a) Local Well-Connectedness Assumption (LWCA): The physical linking of a subsystem S_k to the rest of the subsystems implies that there is a connecting input vector $\underline{e}_k$ having as coordinates all variables connected directly to at least one subsystem output, or external variable (manipulated, or disturbance) and having as outputs the vector $\underline{z}_k$ of all possible measurements and connecting variables to at least another subsystem. The pair of vectors $\left(\underline{e}_k, \underline{z}_k\right)$ defines the natural inputs and outputs of the system S_k. A sub-vector of $\underline{z}_k$ is the connecting output vector $\underline{w}_k$ with coordinates all variables which feed to at least one of the subsystems or measured variables. We assume that the transfer functions $H_k(s) : \underline{e}_k \rightarrow \underline{z}_k$ are well defined and they are proper. These assumptions are referred to as *Local Well-Connectedness (LWC)* and $H_k(s)$ is the k-th connecting transfer function; furthermore, if $H_k(s)$ represents a minimal system, then the system satisfies the *Minimal LWC (MLWC)* assumption. The aggregate system S_a is represented by the transfer function matrix $H(s) = block - diag\, \{H_k(s), k = 1, 2, ..., \mu\}$.

(b) Local Well-Structured Assumption (LWSA): For every subsystem with $\underline{e}_k$, $\underline{z}_k$ physical inputs and outputs, we shall denote by $\underline{v}_k$, $\underline{y}_k$ the effective input, output vectors. We shall assume that $\underline{y}_k$ is a sub-vector of $\underline{z}_k$ in the sense that $\underline{y}_k = K_k \underline{z}_k$, $K_k \in \mathbb{R}^{p_k \times q_k}$, $p_k \leq q_k$ and that $\underline{e}_k$ is expressed as

$$\underline{e}_k = \underline{f}_k + L_k \underline{u}_k = \underline{f}_k + \underline{v}_k, \tag{1.5}$$

where $\underline{f}_k$ is some vector of variables defined by the interconnections and $\underline{v}_k = L_k \underline{u}_k$ has independently assignable (control or disturbance) variables, defined as a combination of a larger potential vector $\underline{u}_k$; thus $\underline{u}_k$, $\underline{z}_k$ emerge as potential inputs and outputs. This assumption is referred to as *Local Well-Structured (LWS)* assumption.

(c) Global Well-Formedness Assumption: [14] Let $S_a = \{S_k, k = 1, 2, .., \mu\}$ be the system aggregate under the LWC and LWS assumptions. The composite system will be called *Globally Well-Formed (GWF)*, if the interconnection rule $\mathscr{F}$: $\underline{e}_1 \times ... \times \underline{e}_\mu \rightarrow \underline{z}_1 \times ... \times \underline{z}_\mu$ represented by the diagram of Fig. 1.8 satisfies the following:

(i) Its output is $\left[\underline{z}_1^t, \ldots, \underline{z}_\mu^t\right]^t = \underline{z}$ and if $\underline{v}_k$ are external input vectors (assignable or disturbances), its inputs $\underline{e}_k$ are expressed as $\underline{e}_k = \sum_{j=1}^{\mu} F_{kj} \underline{z}_j + \underline{v}_k$, F_{kj} real.

(ii) The transfer function from $\underline{v} = [\underline{v}_1^t, \ldots, \underline{v}_\mu^t]^t \rightarrow \underline{e} = [\underline{e}_1^t, \ldots, \underline{e}_\mu^t]^t$ is defined.

If $F = [F_{k,j}]_{k,j \in \mu}$, $K = block - diag\{K_i,\ i \in \mu\}$, $L = block - diag\{L_i,\ i \in \mu\}$, $\underline{u} = [.., \underline{u}_i{}^t, ..]^t$, $\underline{y} = [.., \underline{y}_i{}^t, ..]^t$, then

$$\underline{e} = \underline{v} + F\underline{z}, \quad \underline{z} = H_a(s)\underline{e}, \quad \underline{v} = L\underline{u}, \quad \underline{y} = K\underline{z}, \tag{1.6}$$

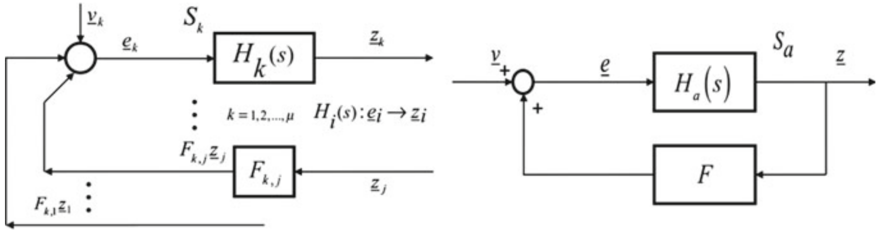

Fig. 1.8 Globally well-formed composite system

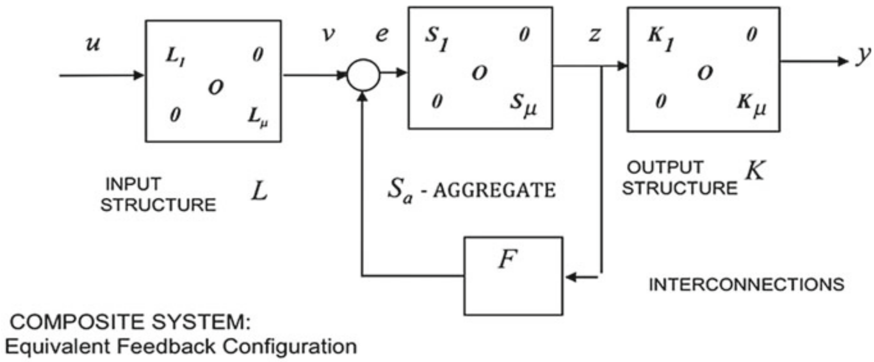

Fig. 1.9 Effective and progenitor system model

and the composite configuration is represented as the feedback configuration of Fig. 1.8.

Note that condition (c.ii) implies that $I - FH_a(s)$ is an invertible matrix. Clearly, the interconnection graph acts as feedback and the selection of effective inputs and outputs is represented as input and output constant compensators and the transfer function of the composite system S_c is

$$G(s) = K \cdot \hat{H}(s), \cdot L \text{ where } \hat{H}(s) = H_a(s) \cdot (I - F \cdot H_a(s))^{-1} \tag{1.7}$$

and it is represented in Fig. 1.9. This expresses the composite system as the action of decentralized input and output reduction (squaring down operation), represented by the input and output transformations K, L, respectively, and of an internal feedback F, representing the topology of the interconnections. The matrix $\hat{H}(s)$ is referred to as *progenitor model* of the composite system and $G(s)$ is the *effective transfer function* of the composite system. The actions of K, L are referred to as *Model Projection* (MP) operations [29, 32] and are forms of "squaring down" [35]. We shall refer to (K, L) pair as the *input–output normalizers*.

The representation of the interconnecting topology as internal feedback provides the means to link the properties of aggregate and the composite system, explains the basic form of evolution in terms of the interconnection graph and allows our intervention in the synthesis problem using results from the feedback theory. The above description of the composite system is general and can lead to family of graphs depending on the cardinality of subsystems and their respective dynamic complexity (see Sect. 1.4.3). A special case of the above general configuration is described below.

(d) Completeness Assumption: The well-formed composite system of Fig. 1.9 will be said to be *complete*, if the following two further conditions hold true:

(i) Every effective subsystem output $\underline{y}_k$ satisfies the condition $\underline{y}_k = K_k \underline{z}_k$, K_k square invertible.

(ii) Every external subsystem vector $\underline{u}_k$ has as many independent coordinates as the dimension of $\underline{e}_k$ input vector, i.e., $\underline{e}_k = L_k \underline{u}_k$ with L_k square and invertible. In this case the matrices K, L are also invertible.

Remark 1.2 The completeness implies that the composite and the aggregate are output feedback and input and output coordinate transformation equivalent, and thus they have the same structural characteristics. Guaranteeing the validity of the above assumptions is both a matter of modelling and selection of input and output schemes.

Open Issues: The representation of the composite linear system given in this section opens up a number of issues for further research. For different interconnection topologies and subsystem cardinalities investigate,

(i) Extending the variable complexity modelling from a single process to a composite system.

(ii) Define the stability and structural properties (McMillan degree, Kronecker invariants, etc.) of the composite system as a function of the corresponding subsystem properties.

1.5.2 Systems Instrumentation and Forms of Evolution

The problem of selection of inputs and outputs sets has a systems dimension, and it is referred to as systems or *global instrumentation* [27, 29]. This problem is different from traditional instrumentation dealing with measuring physical variables, or ways of acting on physical variables. The systems instrumentation involves a number of structure evolution processes linked to the study of four fundamental problems which are

(i) Model Orientation Problems;

(ii) Model Projection Problems;

(iii) Model Expansion Problems;

(iv) Local–Global Structure Evolution Problems.

These problems have a clear model shaping role; each one of them expresses a form of system evolution and their study is reduced to problems of Control Theory and Design. The distinguishing feature of systems instrumentation as far as model shaping is that it acts on the shaping of the input–output structure of the system, rather than the interconnection graph, as described previously. An overview of the overall instrumentation that includes traditional (macro) and systems aspects is given in [27].

1.5.2.1 Model Orientation Problem

A natural system description that makes no distinction as far as the role of process variables and their dependence, or independence is for the linear case the matrix pencil model (first-order differential descriptions) [60], or the general polynomial, or autoregressive model and these characterize the behaviour of an *implicit vector* the coordinates of which are not necessarily independent [7, 36, 69]. For control purposes, there is a need to classify the coordinates of the implicit vector into inputs, outputs and internal variables. This is referred to as *Model Orientation Problem* (MOP) [41]. In many systems, the orientation is not known or, depending on the use of the system, the orientation changes. Questions, such as when is a set of variables implied, or not anticipated by another, or when is it free, have to be answered. The solutions are systems of the standard state-space type and polynomial matrix case. The partitioning of the implicit vector results in a form of evolution of the resulted system from the generator matrix pencil, or autoregressive model with a corresponding evolution of the algebraic structure [41]. If all important variables are included in the physical modelling without a classification into inputs, outputs and internal variables is made, the emerging descriptions based on the implicit vector $\underline{\xi}$ are referred to as *implicit* and in the case of first-order differential descriptions they correspond to the matrix pencil, or generalized autonomous description [36]:

$$S(F,G) : Fp\,\underline{\xi} = G\,\underline{\xi}, \quad F, G \in \mathbb{R}^{\tau\times\upsilon}, \quad p \overset{\Delta}{=} d/dt. \tag{1.8}$$

The natural operator associated with such descriptions is the matrix pencil $sF - G$, and the study of such descriptions relies on the structure of $sF - G$ [19]. The classification of the variables in $\underline{\xi}$ into internal variables, or states $\underline{x}$, assignable, or control variables $\underline{u}$, and measurement, or dependent variables $\underline{y}$ is expressed in terms of transformation $\underline{\xi} = Q\,\underline{\tilde{\xi}}$, where $\underline{\tilde{\xi}} = \left[\underline{x}^t, \underline{u}^t, \underline{y}^t\right]^t$ and $Q \in \mathbb{R}^{\upsilon\times\upsilon}$, $|Q| \neq 0$. Q is the *orientation transformation* (OT) and if the original variables in $\underline{\xi}$ are physical and it is desired to preserve them, then Q has to be of the permutation type and it is a *physical* OT. For first-order linear descriptions, the most general form of oriented models is the *general singular* (GS) description [46]:

$$\begin{aligned} &S(E,A,B,C,D) : E\,p\,\underline{x} = A\,\underline{x} + B\,\underline{u}, \;\; \underline{y} = C\,\underline{x} + D\,\underline{u} \\ &E, A \in \mathbb{R}^{\sigma\times n}, \;\; B \in \mathbb{R}^{\sigma\times p}, \;\; C \in \mathbb{R}^{m\times n}, \;\; D \in \mathbb{R}^{m\times p}, \end{aligned} \tag{1.9}$$

where $\tau = m + \sigma$, $\upsilon = n + p + m$ and in general $\sigma \geq n$. In the case where $\sigma = n$, S is called *singular* and if $\sigma = n$ and $|E| \neq 0$, then the description will be called *regular* and it is equivalent to the standard state-space description $S(A,B,C,D) : p\,\underline{x} = A\,\underline{x}$

$+B\,\underline{u}$, $\underline{y} = C\,\underline{x} + D\,\underline{u}$. The *model orientation* problem (MOP) is then expressed as defining a transformation Q (free, or constrained by physical considerations) such that $S(F, G)$ is reduced to $S(E, A, B, C, D)$ or $S(A, B, C, D)$. The study of MOP for matrix pencil models in the unconstrained case has been considered in [41], and it is equivalent to a partitioning of the Kronecker invariants of $sF - G$. The nature of Kronecker invariants determines the type of the resulting oriented system.

A more general implicit description is the polynomial, or the *autoregressive representation* (AR) [69]. It is a more general implicit description, defined by a polynomial matrix $R(p)$, associated with the implicit vector $\underline{w}$ and represents the behaviour of all external variable trajectories $\underline{w}$ satisfying

$$R(p)\underline{w} = 0. \tag{1.10}$$

We may introduce orientation for such descriptions by introducing some internal variables, expressed by a vector $\underline{\xi}$ and this leads to the AR/MA representation which is specified by two polynomial matrices $H(p)$ and $Q(p)$. The external behaviour consisting of all trajectories $\underline{w}$ of the external variables is related to the trajectory $\underline{\xi}$ of the internal variables by

$$H(p)\underline{\xi} = 0, \underline{w} = Q(p)\underline{\xi}. \tag{1.11}$$

For systems with an explicit input/output structure (splitting of $\underline{w}$ into $\underline{u}$ and $\underline{y}$), the Rosenbrock's system matrix [60] in the s-domain provides a natural model, i.e.,

$$T(s)\underline{\xi} = U(s)\underline{u}, \underline{y} = V(s)\underline{\xi} + W(s)\underline{u}, \tag{1.12}$$

where all matrices are polynomial, with $T(s)$ square and invertible. The corresponding transfer function matrix $G(s)$ is represented as $V(s)T^{-1}(s)U(s) + W(s)$. The orientation problem may now be addressed in a more general setup where first we model the system behaviour, in terms of outputs, then we introduce the internal variables and then we consider the orientation. This procedure involves a realization of $R(p)$, which may be in a matrix pencil form that retains $\underline{w}$ as an output vector and includes physical variables that may act as inputs. In this context, it leads to a family of transfer functions $G(s)$ with properties that evolve from those of $R(p)$. This research is linked to the theory of strict equivalence [26].

Open Issues:There are a number of open research issues for the problem of model orientation. These involve

- **(i)** Development of solutions for the case of matrix pencil descriptions when there are physical constraints on the partitioning of the implicit vector.
- **(ii)** Development of solutions to model orientation for autoregressive models and its link to the theory of strict equivalence.

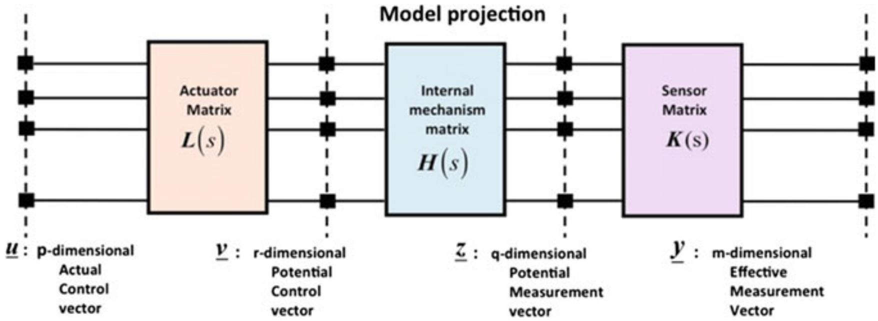

Fig. 1.10 Model projection problems

1.5.2.2 Model Projection Problems

The number of potential control and measurement variables for systems, which may be used, can become very large. For the purpose of control design, we are frequently forced to select a subset of the potential inputs and outputs as effective, operational inputs and outputs. Such a requirement implies reductions in the input and output system maps, variations in the cardinality of the system and results in an evolution process of the resulting system structure from the original one. Developing criteria and techniques for selection of an effective input–output scheme, as projections of the extended input and output vectors, respectively, is what we call *Model Projection Problems* (MPP) described in Fig. 1.10. For linear systems, where orientation has already been decided, and represented by a qxr *progenitor* rational matrix $H(s)$, the MPP is equivalent to selecting the sensor, actuator maps $\mathscr{K}, \mathscr{L}, m \leq q, p \leq r$, with representation the rational matrices $K(s), L(s)$ such that the transfer function $W(s) = K(s)H(s)L(s)$ has certain desirable properties. Clearly, the problem as stated above is in the form of a generalized two-parameter *Model Matching*. The designs of the matrices $K(s), L(s)$ are the instruments defining the evolution and may be assumed in the first instance to be constant. Note that the $\mathscr{K}, \mathscr{L}$ maps are not completely free, but they are constrained by the nature of the specific problem and the need to use certain physical variables. The evolution defined by the MPP family is linked to the process of obtaining new models by reducing the original larger input or output sets. In this sense, projection tends to aggregate and reduce an original model to a smaller dimension with desirable properties. A special problem in this area is the zero assignment by squaring down [35, 43, 59, 61]. A number of key problems related to this form of structure evolution are considered next.

Desired Generic Dimensions Problem: Defining desirable general characteristics, such as number of inputs and outputs on a system model, with some assumed internal structure, is referred to as the *generic dimensionality problem* [32]. We can use conditions, for generic solvability, or generic system properties (such as the Segre index [30]) to define the least required numbers of effective inputs and outputs needed for certain structural properties such as controllability and observability.

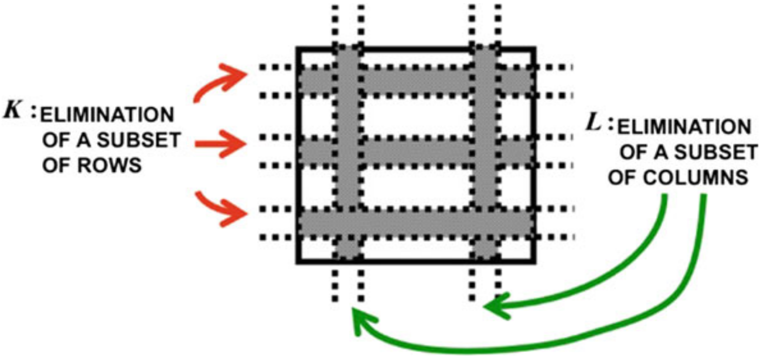

Fig. 1.11 Input–output problems reduction

Generic solvability conditions of control problems, such as pole and zero assignment under different compensation schemes, may lead to constrained integer optimization problems aiming to define generic families of systems for which a range of problems may be solved. The solvability of control problems involves the McMillan degree and/or the generic infinite zero structure of the progenitor transfer function $H(s)$. Use of such invariants requires their robust computation and this is what leads to the study of *structural identification* [42] on early models.

Input–Output Reduction and Well Conditioning of Progenitor Model Problems: The progenitor model in many applications has a large number of physical input and output variables. It is desirable to preserve, such physical variables, but their number may be too large and the progenitor model may not be well conditioned as far as its properties. A special form of model projection may be defined, where only an α subset of inputs and a β subset of outputs is used, leading to an $S_{\alpha,\beta} = S\left(A, B_\alpha, C_\beta, D_{\alpha,\beta}\right)$ subsystem with transfer function $H_{\alpha,\beta}(s)$. The objective is to select the α and β sets such that the resulting $S_{\alpha,\beta}$, $H_{\alpha,\beta}(s)$ is well structured as far as certain properties, which may include input and output regularity, non-degeneracy, minimality, output function controllability, etc. This problem is referred to as *well conditioning by input–output reduction* and it is illustrated in Fig. 1.11. Note that in a transfer function matrix setup, this problem is equivalent to defining sub-matrices of $H(s)$ by eliminating certain columns and rows leading to $W(s) = KH(s)L$ and which have desirable properties [34, 40].

Invariant Structure Transformation/Assignment Problems: The selection of given dimension and structure constant matrices K, L lead to a transfer function $W(s) = KH(s)L$ where the invariant structure of $W(s)$ is obtained by transformation of the set of invariants of $H(s)$. Apart from the study of generic properties and their link to discrete types of invariants, there is also the need to investigate the effects of input and output reduction transformation on well-defined models with fixed parameters. The transformation of one set of invariants to another is a problem not fully understood; certain results in relationship to decoupling have appeared in [13]. The special case of the general model projection is the zero assignment by squaring down leading to a square transfer function [35, 43, 44, 59, 61]. This problem involves transformation of discrete invariants (Forney indices [18]); it is well developed and belongs to the family Determinantal Assignment Problems (DAP) [34] which are studied using exterior algebra and algebraic geometry methods. The two-parameter version of squaring down aims for a transfer function $W(s)$ which is square and has a given zero structure; however, now we can also use the resulting cardinality $(m = p)$ as a design parameter. For all such problems, the overall philosophy is to design K, L such that the resulting model has a given desirable invariant structure or avoids having undesirable structural characteristics. The study of the Morgan's problem may be seen within this class as a transformation of the input structure.

Open Issues: There is a number of open research issues in the area of MPPs which involve

(i) Use of criteria for the selection of numbers of inputs p and outputs m using the McMillan degree n of $H(s)$ based on generic solvability conditions of control problems and conditions for preservation of system properties.

(ii) Characterization of evolution of the structural invariants under the action of constant pre- and post-compensation (L, K) and various input and output dimensions (p, m).

(iii) Selection of suitable (p, m) pair such that zero assignment under (L, K) pair can be achieved.

1.5.2.3 Model Expansion Problems

Defining input–output schemes with the aim to identify (or improve) a system model, or reconstruct an unmeasured internal variable, characterizes the family of *Model Expansion Problems* (MEP). This problem may be seen as prediction of an expanded model from which the current model has evolved. Questions related to the nature of test signals or properties of the measured signals are also important, on top of the more general questions related to the structure of the i/o scheme; the latter gives a distinct signal processing flavour to MEP. Model expansion expresses a form of model structure evolution, where additional inputs and outputs enable a system model to grow to a more full representation of the existing system. This expresses an alternative form of evolution of structure of the model by manipulation of the input–output, external structure. Some distinct problem areas are as follows.

Additional Measurements for Estimation of Variables: In systems, some important variables are not available for measurement. It is then that secondary measurements have to be selected and used in conjunction with estimators to infer the value of unmeasurable variables. The selection of secondary measurements is important for the synthesis of control schemes. The various aspects of the problem are discussed within the well-developed area of state estimation [45].

Input and Output Schemes for System Identification: The selection of input test signals and output measurements is an integral part of the setting up of model identification experiments [17]. In fact, the identified model is always a function of the way the system is excited and observed, i.e., of the way the system is embedded in its experimental environmental. The study of effect of location of the excitation signals and corresponding group of extracted measurements on the identification problem has received less attention. Issues such as how and whether additional excitation signals and extracted measurements may enhance the scope and accuracy of identifiable models are important problems. This area is closely related to the problem of identifiability of models [20, 47].

Model Completion Problems: This class of problems deals with the problem of augmentation of a system operator, like the matrix pencil and has a dual nature to that of model projection, since now we deal with dimensional expansion of the relevant operator. Let $sE - H$ be an $r \times q$ pencil, which is a sub-pencil of the $(r + t) \times (q + v)$ pencil $sE' - H'$, where

$$sE' - H' = \begin{bmatrix} sE - H & X \\ X & X \end{bmatrix}, \tag{1.13}$$

and the X's stand for unspecified pencils of compatible dimensions. Studying the relationships between the sets of invariants of $sE - H$ and $sE' - H'$ pencils and in particular examining the conditions under which we may assign arbitrarily the structure of $sE' - H'$ are known as *Matrix Pencil Completion Problem* [48]. The above formulation may be also extended to that of expansion of polynomial or rational models. It is worth pointing out that such formulations make sense as long as the implicit vector corresponding to the expanded system has new variables which makes sense.

Open Issues: Model expansion requires additional research in areas such as follows:

(i) Detailed system modelling to define family of models which contain the given model as a projection.
(ii) Development of matrix pencil completion problem to the case of polynomial models and rational transfer functions.

1.5.2.4 Local–Global Structure Evolution

When we consider composite systems, then all issues and problems of model orientation, model projection and model expansion may be transferred to the composite system. Note that now the subsystem cardinality may be varying with respective implications on the graph-dependent structure and properties. Guaranteeing structural properties such as controllability, observability, rejection of disturbances, etc. requires the definition of subsets of inputs and outputs at the local, or global level, or appropriate structural combinations of them. In terms of the two-parameter scheme associated with MPP, there is need to define the required Boolean structure of K, L transformations on an internal model, or the modification to the internal graph that can guarantee such properties. This involves examining the Kronecker structure of graph-structured pencil models, which are linked to the presence or the absence of certain system properties.

Open Issues: The role of the interconnection graph is now crucial in defining structure evolution problems which include

(i) Study of Kronecker invariants for structured matrix pencils.
(ii) Development of non-structural (stability, minimum phase) and structural system properties for alternative interconnection graphs and possible variable subsystem cardinality.
(iii) Investigate the role of interconnection graph on system properties, when subprocesses are described by variable complexity models.

1.6 Integrated Operations and Emergent Properties

Figures 1.3 and 1.4 describe the integrated system represented by both operations and designs. Production-level activities take place on a given system; they are mostly organized in a hierarchical manner and they realize the higher level strategies decided at the business level. Vertical activities are issues going through the Business–Operations–Design hierarchy, and they have different interpretations at the corresponding level. The Physical Process Dimension deals with issues of design–redesign of the Engineering Process, and here the issues are those related to integrated design [28]. The Signals, Operations Dimension is concerned with the study of the different operations, functions based on the Physical Process and it is thus closely related to operations for production [58]. In this area, signals and information extracted from the process are the fundamentals and the problem of integration is concerned with understanding the connectivities between the alternative operations and functionalities and having some means to regulate the overall behaviour. Both design, operations and business generate and rely on data and deploy software tools and such issues are considered as vertical activities.

1.6.1 The Multi-modelling and Hierarchical Structure of Integrated Operations

The study of Industrial Processes requires models of different types. The borderlines between the families of Operational Models (OM) and Design Models (DM) are not always very clear. Models linked to design are "off-line", whereas those used for operations are either off-line or "on-line" [58]. For process-type applications, models are classified into two families referred to as "line" and "support" models [58]. Line models are used for determining desired process conditions for the immediate future, whereas support models provide information to control models, or they are used for simulation purposes. A major classification of models is into those referred to as "black" and "white" models [58]. White models are based on understanding the system (physics, chemistry, etc.), and their development requires a lot of process insight and knowledge of physical/chemical relationships. Such models can be applied to a wide range of conditions, contain a small number of parameters and are especially useful in the process design, when experimental data are not available. Black models are of the input–output type and contain many parameters, but require little knowledge of the process and are easy to formulate; such models require appropriate process data and they are only valid for the range, where data are available.

Handling the high complexity of the overall system is through aggregation, modularization and hierarchization [8], and this is what characterizes the overall structure described in Fig. 1.3. To be able to lump a set of subsystems together and treat the composite structure as a single object with a specific function, the subsystems must effectively interact. Modularization refers to the composition of specific function units to achieve a composite function task. Aggregation and modularization refer to physical composition of subsystems through coupling, and it is motivated by the needs of design of systems. Hierarchization is related to the stratification of alternative behavioural aspects of the entire system and it is motivated by the need to manage the overall information complexity. The production system may be viewed as an information system, and thus notions of complexity are naturally associated with it [49].

Hierarchization has to do with identification of design and operational tasks, as well as reduction of externally perceived complexity to manageable levels of the higher layers. At the top of the hierarchy, we perceive and describe the overall production process as an economic activity; at this level we have the lowest complexity, as far as description of the process behaviour. At the next level down, we perceive the process as a set of interacting plant sections, each performing production functions interacting to produce the economic activity of the higher level. At the next level down in the hierarchy, we are concerned with specification of desired operational functions for each unit in a plant section and so on, and we can move down to operation of units with quality, safety, etc., criteria and further down to dynamic performance, etc. In an effectively functioning hierarchy, the interaction between subsystems at lower level is such as to create a reduced level of complexity at the level perceived above [49]. The hierarchization implies a reduction of externally perceived complexity successfully, as we proceed up the hierarchy till the top level. A simpler representation of the overall operational hierarchy of Fig. 1.3 is as shown in Fig. 1.12 [22] having blocks with the following modelling requirements:

0-level: (Signals, Data Level). Physical variables, Instrumentation, Signal processing, Data Structures.

1, 2-levels: (Primary Process Control). Time responses, Simple linear SISO / MIMO models.

3-level: (Supervisory Control Level). Process Optimization Models, Statistical Quality Models (SPC, Multivariate, Filtering, Estimation), Fault Diagnosis, Overall Process State Assessment Models.

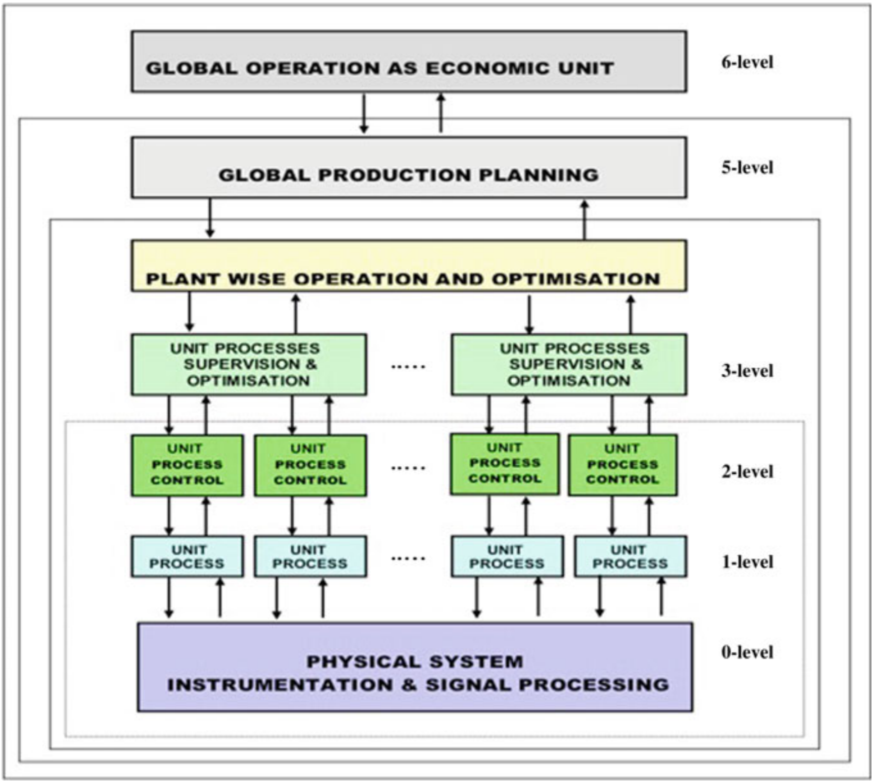

Fig. 1.12 Hierarchy of system operations

4-level: (Plant Operation and Logistics). Nonlinear Static or Dynamic Models for Overall Plant, Operational Research Models, Discrete Event Models (Petri Nets, Languages, Automata).

5-level: (Global Production Planning Level). Production Models, Planning, Forecasting, Economic Models, Operational Research, Game Theory Models.

6-level: (Business Level). Enterprise, Business Modelling, System Dynamics, Forecasting, Graph Models, Economic Models, etc.

A functional representation of the overall system represented by families of relevant models which provide an alternative description of Fig. 1.3 is given in Fig. 1.13. The different types of models in the above groupings are interrelated. Each of the model families on the unit level is simplified and aggregated to models on the plant level and then on the production site, business unit and possibly the enterprise level. Model composition is accompanied by simplification. The latter classification is of functional type, and the Process Control Hierarchy implies a nesting of models to a layered hierarchy with variable complexity as shown in Fig. 1.13. This diagram indicates that at the level of the process we have the richest possible model in terms of signals, data, full dynamic models. Then, as we move up in the hierarchy, the corresponding models become simpler, but also more general since they then refer not to a unit but to a section of the plant. The use of functional-type models for the Process Control Hierarchy implies a nesting of models to a layered hierarchy with variable complexity as shown in Fig. 1.13. The diagram of Fig. 1.13 indicates that at the level of the process we have the richest possible model in terms of signals, data and full dynamic models. Then, as we move up in the hierarchy, the corresponding models become simpler, but also more general since they then refer not to a unit but to a section of the plant. The operation of extraction of the simpler models is some form of projection, whereas wider scale models are obtained by using plant topology and aggregations. These models, although of different nature and scope, are related, since they describe aspects of the same process. Dynamic properties of subsystems are reflected on simpler, but wider area models, although this is what we may refer to as Embedding of Function Models [22].

1.6.1.1 System and Emergent Properties

The notion of emergence is intimately linked with complex systems and has its origins in philosophy [38]. With complex processes such as the Integrated Design and Operations, there is a number of emergent properties appearing which require a systems-based characterization. Emergence refers to understanding how collective properties arise from the properties of parts. More generally, it refers to how behaviour at the collective level of the system arises from the detailed structure, behaviour and relationships on a finer scale. System properties may be classified as intrinsic and extrinsic. An intrinsic property relates to the class of features and characteristics which is inherent and contained wholly within a physical or virtual object. The extrinsic properties are those which are not part of the essential nature of things and have their origin outside the object under scrutiny. Within the context of a system, an emergent property is an extrinsic one that is not an intrinsic property of any constituent of that system, but is manifested by the system as a whole.

A description of system properties in the operational system hierarchy of Fig. 1.3 is given by the diagram in Fig. 1.14. The above diagram provides an overview of the integrated system and the system properties associated with it. The family of system properties is classified into the family of *intrinsic system properties* and the *emergent properties*. Intrinsic system properties such as systems safety, reliability, quality, etc. have the common feature that can be directly assessed from signals and data associated with the physical system (physical, communications, operations layers). Emergent properties on the other hand, such as risk, assurance and sustainability, can be assessed using the inputs provided from indicators associated with the intrinsic system properties. We may define:

Definition 1.5 An *emergent property* is an extrinsic system property, which is assessed using an *emergent property* function with a domain, the set of values of intrinsic system properties.

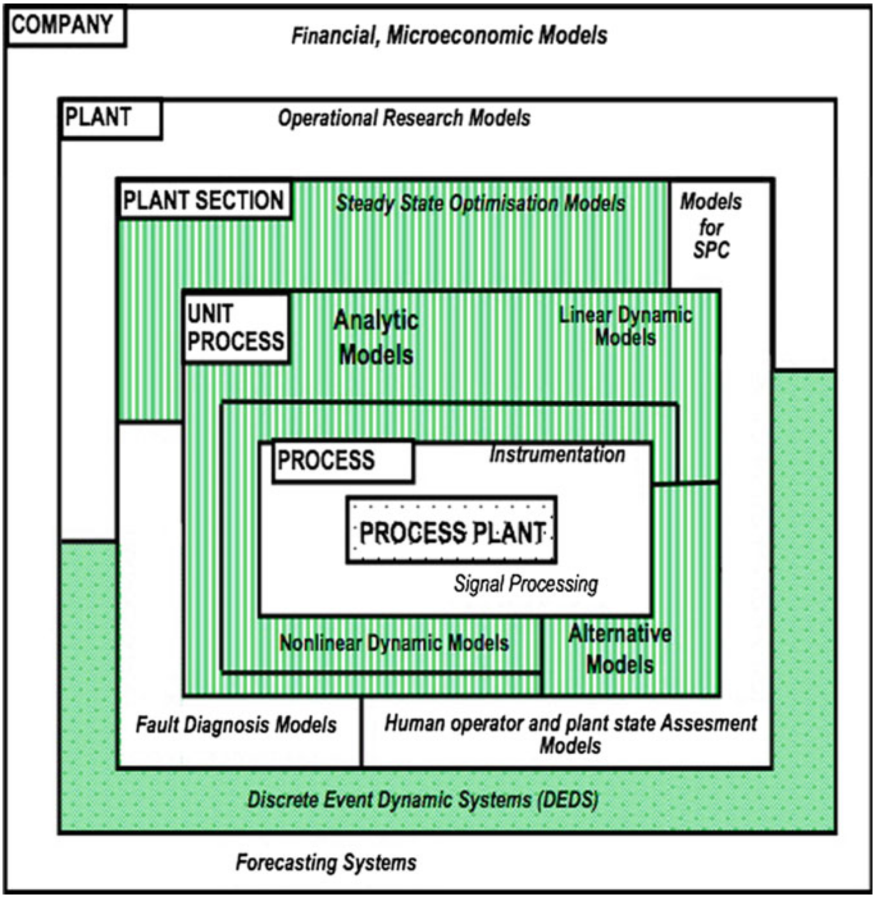

Fig. 1.13 Functional representation of the integrated system

The development of the emergent property function is a study subject in the area of overall assessment of industrial/manufacturing systems.

1.6.1.2 Control in the Hierarchical Structure

The hierarchical model of the Overall Process Operations involves processes of different nature expressing functionalities of the problem. Such processes are interlinked, and each one of them is characterized by a different nature model. We can use input–output descriptions for each of the subprocesses, with an internal state expressing the variables involved in the particular process and inputs and outputs expressing the linking with other processes [22]. Such a model is generic and can be used for all functionalities described in Fig. 1.3. We may adopt a generic description

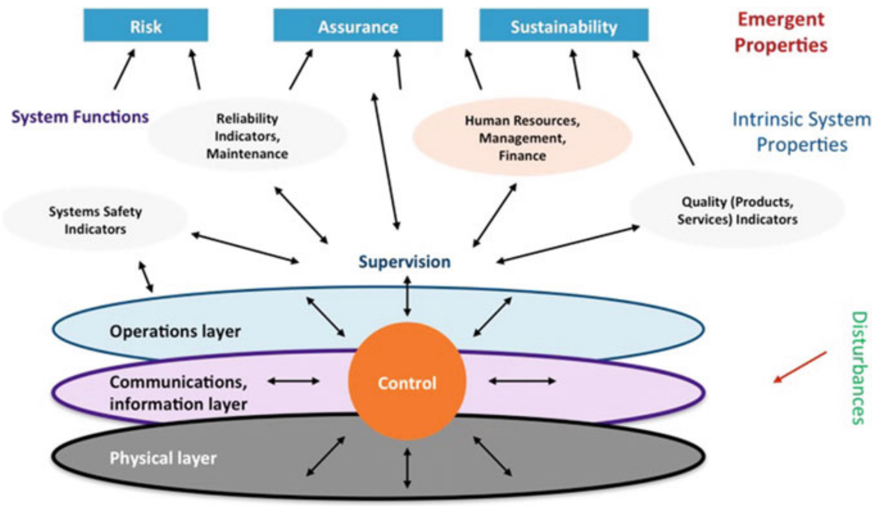

Fig. 1.14 System and emergent properties

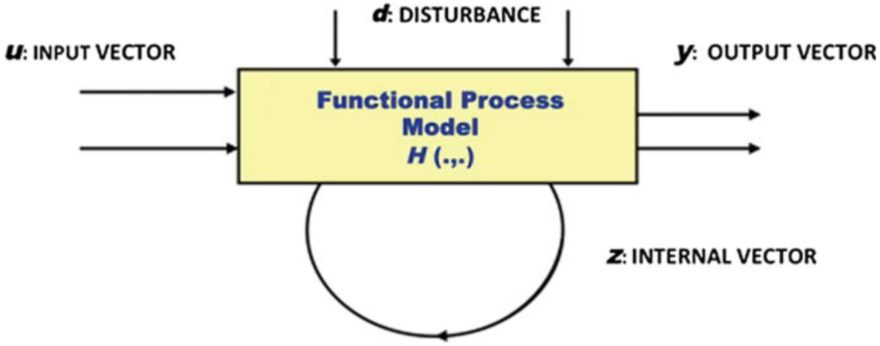

Fig. 1.15 A functional model for a general process

for the various functions as shown in Fig. 1.15, where u_i denote independent manipulated variables of the function model, called system inputs; y_j are the independent controlled variables that can be measured and they are called the system outputs; d_k are the exogenous variables, the disturbances. A description for a functional model of a general process expresses the relationships between the vectors $\underline{u}$, $\underline{d}$, $\underline{y}$ defined by $\underline{y} = H(\underline{u}; \underline{d})$, where H expresses relationships between the relevant variables, and described in Fig. 1.15.

$$M\left(\underline{u},\ \underline{y},\ \underline{d};\ \underline{z}\right) \begin{cases} F\left(\underline{z},\ \underline{u},\ \underline{d}\right) = 0 \\ \underline{y} = G\left(\underline{z},\ \underline{u},\ \underline{d}.\right) \end{cases}$$

The development of such models involves the following:

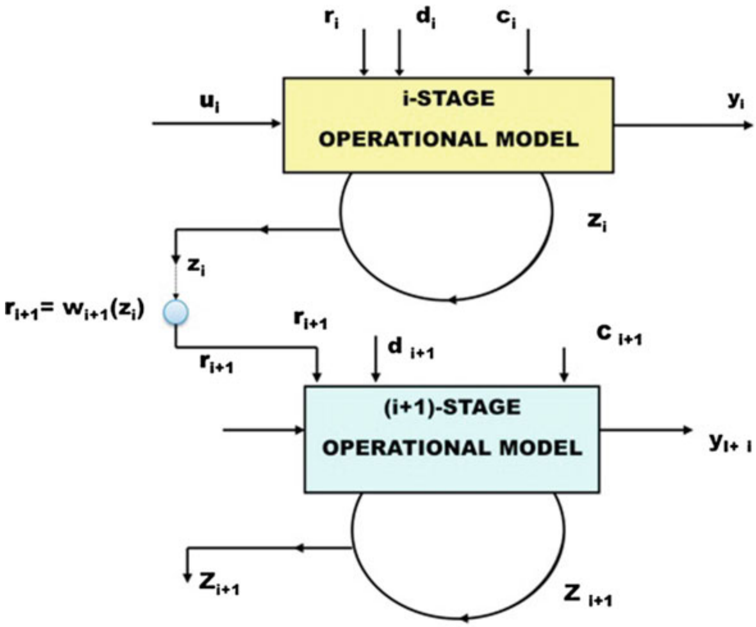

Fig. 1.16 Nesting of models in the hierarchy

(i) For the given function establish a conceptual model based on its role in the operational hierarchy.
(ii) Define the vector of internal variables z and determine its relationships to input and output vectors using any physical insight that we may possess about the functioning of the internal mechanism.
(iii) Establish the relationships between the alternative vectors z associated with problems of the operational hierarchy.
(iv) Define the appropriate formal model to provide an adequate description for the H functional model.

These generic steps provide an approach, which involves many detailed modelling tasks. The above generic steps are providing an approach, which, however, involves many detailed modelling tasks involving issues such as classification of variables to inputs, outputs, disturbances, internal variables, specification of formal description for H, definition of performance indices, etc. The nature of variables and the type of problem under consideration determines the nature of the F, G functions. The implicit model $M(\underline{u}, \underline{y}, \underline{d}, \underline{z})$ is referred to as a $\underline{z}$-stage model.

The selection of the operational stage determines the nature of the internal vector $\underline{z}$ and thus also the corresponding $\underline{z}$-stage. Describing the relationship between different stages, internal vectors are closely related to the problem that is referred to as *Hierarchical Nesting* or *Embedding of Function Models*. The fundamental shell of this hierarchical nesting architecture is described in Fig. 1.16. The reference vector

$\underline{r}_{i+1}$ of operational objectives of the $(i+1)$-stage is defined as a function of the i-th-stage internal vector $\underline{z}_i$, that is, $\underline{r}_{i+1} = w_i(\underline{z}_i)$. A scheme such as the one described above is general and can be used to describe the essence of the hierarchical nesting. This scheme can be extended to describe relations between models associated with functions at the same level of the hierarchy, extend upwards to business-level activities and downwards to the area of the physical process. It is worth pointing out that as going down the hierarchy the complexity and granularity of the subprocess models increase, whereas their nature changes. Note that all subsystems are linked and they express an alternative form of nesting of subsystems of variable complexity and nature, which may be referred to as hierarchical nesting of the operationally integrated system.

The hierarchical nesting introduces new control and measurement issues for the operationally integrated system. If the vector of internal variables in the j-th subsystem in the hierarchy is a state vector x_j, then its state space $\mathscr{X}_j$ is linked to the final subsystem state space $\mathscr{X}_k$ (Physical subsystem state space) in terms of projections/aggregation. The final subsystem state space $\mathscr{X}_k$ of the system corresponds to the physical subsystem of the integrated system. The nesting of state spaces implied by the hierarchical structure is described in Fig. 1.17, and hierarchy-depended system issues related to the implied coupling of subsystems and the different nature of their models. The fact that each stage model in the hierarchy is of different nature than the others makes the overall system of hybrid nature [5].

The nesting of systems implies a multilayer hybrid structure and some new system issues related to the notions of

(i) Global Controllability;
(ii) Global Observability.

Global Controllability Problem: This refers to the crucial issue of whether a high-level objective (possibly generated as the solution of a decision problem at a high level) can be realized within the existing constraints at each of the levels in the hierarchy and finally at lowest level, where we have the physical process (production stage). This is a problem of *Global Controllability*, which may be seen as a problem of *Realization of High-Level Objectives* throughout subprocess in the hierarchy. This new problem requires development of a multilevel hybrid theory, and it can take different forms, according to the nature of the particular stage model. The Global Controllability problem is central to the development of top-down approaches in the study of hierarchical organizations.

Global Observability Problem: This is of dual nature to global controllability and refers to the property of being able to observe aspects of behaviour at the different layers in the hierarchy by appropriate measurements, or estimation processes which are built in the overall scheme. *Global observability* expresses the ability to define *model-based diagnostics* that can predict and evaluate certain aspects of the overall behaviour. It is assumed that the observer has access to the information contained at all stages of the hierarchy, where only external measurement provides the available information. This problem is linked to the development of *bottom-up* approach in the study of hierarchical organizations. The measurements and diagnostics defined

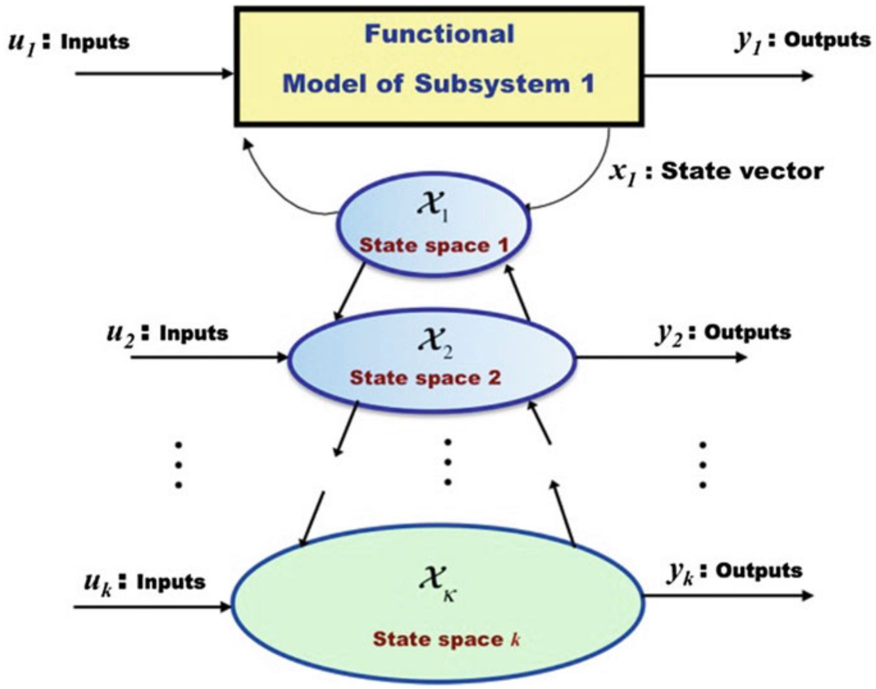

Fig. 1.17 Dynamical nesting in the hierarchy

on the physical process are used to construct the specific property functional models, and thus global observability indicates the quality of the respective functional model.

Open Issues: Integration of Operations requires study of fundamental problems and development of new research areas which include the following:

(i) Understanding the derivation of the different functional models and how they are interfaced, referred to as f*unctional model derivation and interfacing*.
(ii) Defining models for the characterization of emergent properties.
(iii) Development of multilevel hybrid system theory.
(iv) Understanding the different aspects of global controllability and global observability and defining criteria for their characterization.

1.7 The Notion of System of Systems

The concept of "System of Systems" (*SoS*) has emerged in many and diverse fields of applications and describes the integration of many independent, autonomous systems, frequently of large dimensions, which are brought together in order to satisfy a global goal and under certain rules of engagement [23, 50]. These complex multisystems exhibit features well beyond the standard notion of system composition, and represent a synthesis of systems which themselves have a degree of autonomy, but this composition is subject to a central task and related rules. So far, the term *SoS* has been used in a very loose way, by different communities and defined in terms of their characteristics with no special effort to give it a precise definition based on rigorous methodologies and concepts of the Mathematical System Theory [52]. Establishing the links with the traditional system theory approaches is essential, if we are to transfer and appropriately develop powerful and established analytical tools required for their design/redesign. Within this new challenging paradigm, the notion of emergence is also frequently used in a rather loose way. The need for a structured definition of the *SoS* notion has been raised in [33, 37, 52] and will be further developed in this section. A central part of our effort is to explain the difference of *SoS* from that of *Composite Systems* (CoS) which leads to the generalization of the standard notion of *interconnection topology* (linked to composite systems) to the new notion of "*systems play*" [52].

1.7.1 The Empirical Definition of System of Systems

An aggregate of systems leads to the creation of new forms of systems which may be either described within the framework of composite systems, or demonstrate additional features which add complexity to the description and may be referred to as system of systems. The term system of systems (*SoS*) has been used in the literature in different ways and a good treatment of the topic is given in [23]. Most definitions (see references in [23]) describe features or properties of complex systems linked to specific examples. The class of systems exhibiting behaviour of *SoS* typically exhibits aspects of the behaviour met in complex systems; however, not all complex problems fall in the realm of *SoS*. Problem areas characterized as *SoS* exhibit features such as follows [51]:

System of Systems Features: Operational independence of elements; Managerial independence of elements; Evolutionary development; Emergent behaviour; Geographical distribution of elements; Interdisciplinary study; Heterogeneity of systems; Network of systems.

The definitions that have been given so far contain elements of what the abstract notion should have, but they are more linked to specific features linked to areas of applications. A summary of different definitions is given in [37] (Part 1) where the different sources are also listed.

Summary of descriptive definitions for *SoS*

(i) Systems of systems exist when there is a presence of a majority of the following five characteristics: operational and managerial independence, geographic distribution, emergent behaviour and evolutionary development.

(ii) Systems of systems are large-scale concurrent and distributed systems that are comprised of complex systems.

(iii) Enterprise Systems of Systems Engineering is focused on coupling traditional systems engineering activities with enterprise activities of strategic planning and investment analysis.

(iv) System of Systems Integration is a method to pursue development, integration, interoperability and optimization of systems to enhance performance in future battlefield scenarios.

(v) In relation to joint war-fighting, system of systems is concerned with interoperability and synergism of Command, Control, Computers, Communications, and Information and Intelligence, Surveillance, and Reconnaissance Systems.

(vi) System of Systems is a collection of task-oriented or dedicated systems that pool their resources and capabilities together to obtain a new, more complex, "meta-system" which offers more functionality and performance than simply the sum of the constituent systems.

(vii) Systems of Systems are large-scale integrated systems which are heterogeneous and independently operable on their own, but are networked together for a common goal. The goal, as mentioned before, may be cost, performance, robustness, etc.

(viii) A System of Systems is a "super system" comprised of other elements which themselves are independent complex operational systems and interact among themselves to achieve a common goal. Each element of an *SoS* achieves well-substantiated goals even if they are detached from the rest of the *SoS*.

The above definitions are mostly descriptive, but they capture crucial features of what a generic definition should involve; however, they do not answer the question, why is this new notion different than that of composite systems. The last two definitions [23] are more generic and capture the key features of the notion, but they still do not provide a systems working tool for design and redesign of *SoS*. A major task in providing a systems definition for *SoS* is to demonstrate the differences between *SoS* and *Composite Systems* (CoS) and explain why *SoS* is an evolution of *CoS*.

1.7.2 Composite Systems and SoS: The Integrated Autonomous and Intelligent System

Developing the transition from CoS to *SoS*, we need to identify the commonalities and differences between the two notions. We note [37]:

(a) Both CoS and *SoS* are compositions of subsystems and they are embedded in the environment of a larger system.

(b) The subsystems in CoS do not have their independent goal; they are not autonomous and their behaviour is subject to the rules of the interconnection topology.

(c) The interconnection rule in CoS is expressed as a graph topology.

Fig. 1.18 Simplified description of the system

(d) The subsystems in *SoS* may have their own goals, and some of them may be autonomous, semi-autonomous, or organized as autonomous groupings of composite systems.

(e) There may be a connection topology rule expressed as graph topology for the information structures of the subsystems in an *SoS*.

(f) The interconnection rule is described in a more general form than that of the graph topology, named as "*systems play*" and every subsystem enters as an agent with their individual operational sets and goals.

The development of an abstract, systems-based characterization of *SoS* requires the following:

(i) Consider the notion of the system in a more general setup suitable for *SoS*.

(ii) Specify the special features which define the notion of "intelligent autonomous agent".

(iii) Provide a characterization of the generalized notion of relationships defined as "systems play".

First, we revisit the definition of a system as given in Sect. 1.2 and illustrated in Fig. 1.2 and define the notion of autonomy.

Definition 1.6 A simple or composite system is referred to as autonomous, if its relations with other systems in the environment are expressed only through the operational goals.

The notion of autonomy implies that as far as the other systems in the environment are related only through its goals and not through some interconnection topology. This is described in Fig. 1.18.

We are also referring to the notion of intelligent system, and this requires some appropriate interpretation in terms of control capabilities. We may define this notion as follows.

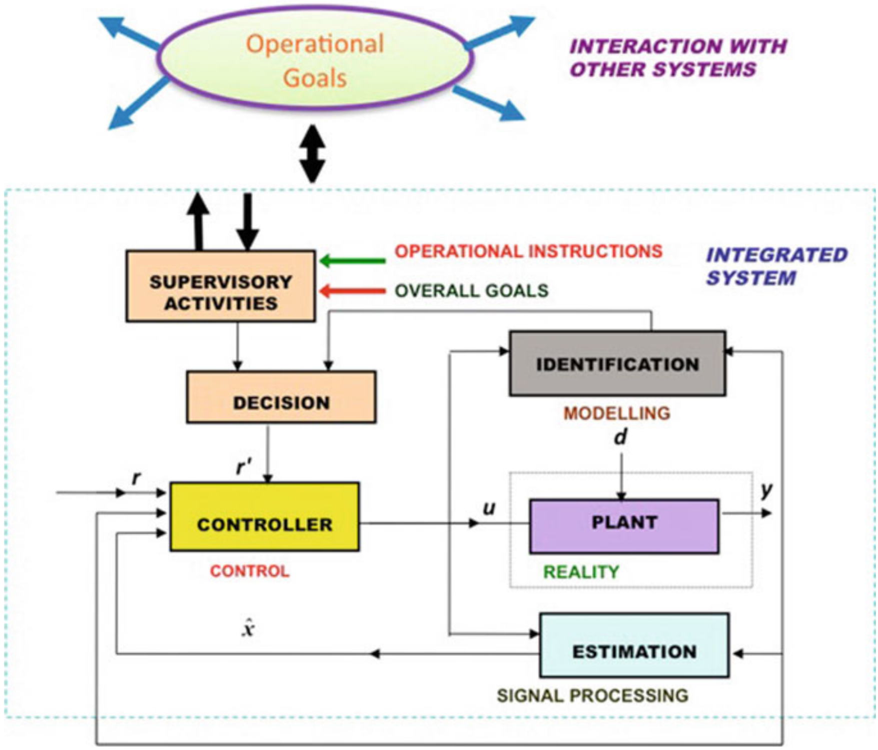

Fig. 1.19 Integrated and autonomous system © [2013] IEEE. Reprinted, with permission, from [33]

Definition 1.7 A system that has the ability to develop measurements for its behaviour, define models for different operational conditions, self-adapt itself to changes in the environment and control its behaviour subject to a set of objectives and constraints will be called *intelligent* or *intelligent agent*.

The intelligent system has the ability to interpret the high-level operational goals to appropriate measurement and control strategies which will enable to implement their realization at all aspects of its behaviour. A scheme that will enable the functioning of a system as an intelligent agent is represented in Fig. 1.19.

In the system representation of Fig. 1.19, the system appears as an autonomous agent (internal system structure together with its inputs and outputs), where its kernel is the physical system (simple or composite), having its operational instructions, and provided with control, modelling, measurement and supervisory capabilities; the supervisory activities allow the realization of the higher level operational goals.

Such a system is an intelligent agent and will be referred to as *integrated system* [33]. As far as its behaviour within the general systems' environment, this type of system is engaged only in terms of its operational goals. The *integrated system* term is used to distinguish it from systems which have no integrated control and information processing capabilities and which may be referred to as basic systems. The integrated structure implies that such goals may be realized and produce the desirable behaviours. Interactions with other systems take place only at the operational goals level.

1.7.3 The Systemic Definition of System of Systems

The distinguishing feature of *SoS* is that the subsystems participate in the composition as intelligent agents with a relative autonomy and behave as actors in a play. The latter property requires that the systems entering the composition, expressed as rules, are of the integrated type, since this requires capabilities for control, estimation, modelling and supervisory capabilities. Features, such as large dimensionality, heterogeneity, network structure, Operational, Adaptability, Emergent Behaviour, etc. may be also present in *SoS* as well as in the case of CoS. We define:

Definition 1.8 Consider a set of systems $\{S\} = \{S_i : i = 1, 2, \ldots, \mu\}$ and let $\mathscr{F}$ be an interconnection rule defined on the information structures of S_i systems. The action of $\mathscr{F}$ on $\{S\}$ defines a *Composite System*, $S_{cos} = \mathscr{F} * \{S_i\}$ or the composition of $\{S\}$ under $\mathscr{F}$.

In the above definition, the subsystems S_i can be basic or integrated. The information structure of each system is defined by the pair of the input and output influence vectors, and the interconnection rule may be represented by a graph topology [32]. The resulted system is embedded in a larger system and it is treated as new system with its own system boundary. This definition may now be extended as follows.

Definition 1.9 Consider a set of integrated and autonomous systems $\{S\} = \{S_i, P_i : i = 1, 2, \ldots, \mu\}$, where P_i are the operational goals of S_i. If $\mathscr{G}$ is a general rule defined on the operational goals P_i, of S_i systems such that $\mathscr{G} : P_1 \times ... \times P_\mu \rightarrow \mathscr{G}(P_1 \times ... \times P_\mu) = \mathscr{G}(P_i)_1^\mu$, then the system

$$S_{SoS} = \mathscr{G} * \{S_i\} = \{S_i, P_i : \mathrm{i} = 1, ..., \mu; \mathscr{G}(P_i)_1^\mu\} \tag{1.14}$$

will be called a *System of Systems* associated with a systems play $\mathscr{G}$.

In the above definition, the notion of *SoS* emerges as an evolution of CoS since the systems are assumed to be integrated and autonomous. The notion of the interconnection topology $\mathscr{F}$ is now extended to that of the *systems play* $\mathscr{G}$. The subsystems in *SoS* now act as autonomous intelligent agents which can enter into relations with other systems as defined by the systems play $\mathscr{G}$. The integrated nature of the subsystems implies that the results of the system play can be realized within each subsystem. The nature of the applications defines the *systems play*, which frequently may be expressed as a game defined on intelligent agents.

An *SoS* emerges as a multi-agent system composed of multiple interacting intelligent agents (the subsystems). This multi-agent systems view allows *SoS* to act as vehicle to solve problems which are difficult or impossible for an individual agent. The multi-agent dimension of *SoS* has important characteristics such as follows [1]:

(i) Autonomy: the agents are at least partially autonomous;
(ii) Local views: no agent has a full global view of the system, or the system is too complex for an agent to make practical use of such knowledge;
(iii) Decentralization: there is no designated single controlling agent, but decision and information gathering is distributed.

It is these properties that allow *SoS* to develop "self-organization" capabilities and find the best solution to the problems defined on them.

A major challenge in the development of a unifying approach to the study of *SoS* is the quantitative characterization of the new notion of the *systems play*. Taking into account that *SoS* problems emerge in many and diverse domains, it is clear that some classification of the general *SoS* family into subfamilies with common characteristics is essential before we embark to the characterization of notions such as *systems play*. There is frequently the use of the term *SoS* for *Physical or Natural* systems. Such systems are related to the natural world and social–economic phenomena and are the results of evolution of physical, or socio-economic processes and typical examples are those of the "ecosystem" of a geographical region, or issues related to "social phenomena". Our current approach based on *Artificial or Engineered SoS* requires further development to handle issues of lack of autonomy, or uncertainty in the links between subsystems. Of course, there are grey areas between the two classes of *Artificial and Engineered SoS (E-SoS)* and some further classifications are given in [33].

Note that in *E-SoS* the "goal" is linked to some coordination effort. This leads to another way of classifying *SoS* based on structural and operational characteristics. This classification refers to the mechanisms defining the relations between the subsystems. We may distinguish [33]

- Goal Driven and Unstructured (GU-*SoS*)
- Goal Driven with Central Coordination (GC-*SoS*)

In *GU-SoS* class, the central goal for the system operation is set, as well as the environment within which the system operations will take place. In this case, the nature of the system play is entirely defined by the set goal, which may be in the form of a game where the intelligent agents may participate. A further classification for this class is into

- Pure Goal Driven (P-GU-*SoS*)
- Goal and Scenario Driven (S-GU-*SoS*)

In the *P-GU-SoS* class, the subsystems, as intelligent agents, interpret the central goal, may assign to themselves sub-goals and they then develop actions and self-organization to achieve the central goal, which may be expressed as optimization of a performance index, subject to satisfaction of their individual goals. In *S-GU-SoS*, a scenario linked to the goal is given, the subsystems as intelligent agents undertake roles which aim to optimize a central performance index and satisfy their own particular goals. The *GC-SoS* class on the other hand has the same features as the P-GU-*SoS* and similar subclasses with the additional feature of the existence of coordination. The existence of coordination introduces a structure to the interpretation of the goal by the subsystem and the development of appropriate scenarios to achieve the central goal and partial goal. Coordination is common to E-*SoS* and may be viewed as an interpreter for the development of operational activities. The nature of coordination also introduces special features to *SoS* characterization since it introduces a structure to the resulted *systems play*. Coordination is a form of organization, and there may be different types such as "Hierarchical", "Heterarchical" and "Holonic" [67]. Such forms of organization structure affect the systems play and the development of scenarios. Types of *SoS* where the subsystems are of the engineering type without human action involvement are referred to as "*hard*". Systems involving human presence and behaviour will be referred to as "*soft*" and those involving a mixture of the two types will be called "*hybrid*".

1.7.4 Methods for the Characterization of Systems Play

The development of a description for the systems play depends on the nature of the particular *SoS*. An effort to review the relevant methodologies from system and control which may be used in describing the systems play has been given in [33] and is outlined next. These different methodologies may provide the required framework for the characterization of the systems play and include methods such as Cooperative Control, Market-Based Coordination Techniques, Population Control methodologies and Coalition Games. Each of these methodologies provides formal descriptions of the notion of *systems play*, and they are outlined next.

Cooperative Control: A typical case describing a class of *SoS* is the Vehicle Formation Problem [16] defined as the control of the formation of v vehicles that are performing a shared task; the task depends on the relationship between the locations of the individual vehicles and the task defines the scenario that has to be realized. It is assumed that the vehicles are able to communicate with the other vehicles in carrying out the task and they have capabilities to control their position in the effort to perform the task. Each vehicle is described as a rigid body moving in space and a state vector x_i may be associated with each one; by $x = (x_1, .., x_v)$, we may represent the complete state for the set of v vehicles. The collection of all individual states defines the state of the system, and the execution of assigned task requires the assignment of additional states that can make the system an *SoS*. The development of the scenario and task is handled by introducing for each vehicle an additional discrete state, α_i, which defines the role of the vehicle in the task and which is represented as an element of a discrete set A, the nature of which depends on the specific cooperative control problem. Such problems may be formulated as constrained optimization problem. For *SoS*, the problems of interest are those involving cooperative tasks that can be solved using a decentralized strategy.

Market-Economics Based Coordination Techniques: The distinguishing feature of *SoS* is that there are autonomous units with their own management and control functions that are coupled by resource flows which need to be balanced, over appropriate periods of time depending on local or global storage capacities. The performance of the subsystem consumption and production is influenced by availability of these resources. To perform an arbitration of these flows requires economic balancing mechanisms [12, 68]. The management of the resource flows may be expressed as a network management problem, given that the resource flows define some generic network structure within which we define the flows. Clearly, the overall system performance and behaviour is influenced by discrete decisions taken. Two different approaches that can be used for the management of such flow-coupled *SoS* are *economics-driven coordination* and *market-based mechanisms*. In both cases, the coordinator has only limited information about the behaviour and the constraints of the local units which perform a local optimization of their operational policies. In the economics-driven coordination, it is assumed that the control of *SoS* involves the setting of production / consumption constraints or references between the global *SoS* coordinator and the controllers of individual systems. The *SoS* coordinator utilizes simplified models of the subsystems, and a model of the connecting networks to compute references or constraints on the exchanged flows. The resulting optimization is based on the dynamic price profiles for the resources that are consumed or produced by the subsystems over the planning horizon. An alternative approach is to use mechanisms employing the concepts of *economic markets* to distribute limited resources between subsystems. The market is defined as a population of agents consisting of producers selling goods and consumers buying these goods [12], where the consumers' demand depends on the usefulness or *utility* of a good for the completion of its task. Market-based mechanisms are inherently decentralized and can thus be mapped directly to systems with autonomous subsystems.

Population Control Methods: Population control refers to systems that comprise a large number of semi-independent subsystems, which macroscopically are viewed in terms of their emergent behaviour. Such systems are used in ecology to capture the fluctuations in the populations of interacting species and the relevant models use continuous variables to capture populations and differential equations to capture their evolution. Of special interest is the class of mixed-effect models [54], which address the evolution of a heterogeneous population of individuals, which deploy ordinary differential equations, but with parameters linked to appropriate probability distributions. Population systems dynamics are gaining in importance, as man-made systems become increasingly complex and larger scale and control of the emergent behaviour of large collections of semi-autonomous subsystems becomes an issue. Such methods are primarily motivated by biological applications but have potential for the engineering field of *SoS*.

Coalition Games: The basic idea of *SoS* is to consider the overall system as a set of subsystems that are controlled by local controllers or agents which may exchange information and cooperate. This feature demonstrates the link of *SoS* to distributed and decentralized control schemes with the additional property that the interaction between the subsystems may indicate a time-varying coupling. It is this special feature that indicates the links to a rather new category of management and control schemes referred to as *coalitional management schemes* [54]. In this paradigm, different agents cooperate when there is enough interaction between the controlled systems and they work in a decentralized fashion when there is little interaction. A coalition is a temporary alliance or partnering of groups in order to achieve a common purpose or to engage in a joint activity [56]. A coalition of systems is a temporary system of systems built to achieve a common objective. Forming coalitions requires that the groups have similar values, interests and goals which may allow members to combine their resources and become more powerful than when they each acted alone.

Open Issues: The development of a systems theoretic approach for *SoS* is still in its early stages of development. Broad areas where development is required involve

(i) Characterization of the family of *SoS* according to their origin (engineered, natural, social systems);

(ii) Identifying the methodologies that can contribute to the characterization of *systems play*.

1.8 Conclusions and Future Research

Control Theory and Design have developed around the classical servomechanism paradigm. The area of Systems Integration for large Complex Systems, involving both design and operations, introduces many new challenges and a number of new paradigms generating new requirements and needs for future developments of the Systems and Control Theory beyond the classical paradigm. The identified new paradigms of *Structure Evolving Systems (SES)* and *System of Systems (SoS)* are new areas of complex systems relevant to integrated design and operations for the family of *Engineered Systems*. For the design problem, the challenges in the *SES* area come from the cascade and design-time-dependent evolutionary nature of the process, whereas for system operations the challenges come from the *SoS* type of complexity and specifically the need to characterize the notion of the *systems play*. Additional issues that introduce new dimensions of complexity come from the large dimensions of the processes which have not being considered here. The paper outlines the problem areas and the new challenges posed by *SES* and *SoS* paradigms and identifies some relevant methodologies for their development. The long-term objectives of the proposed research have been the management of complexity in engineering systems and the areas considered here were

(i) Explaining aspects of structure evolution in design and redesign of systems and developing methodologies for controlling the development of the evolutionary design processes.
(ii) Understanding complexity of integrated operations, addressing issues stemming from their organization and characterizing the nature of emergent properties.
(iii) Developing a systems-based characterization of the *SoS* family, characterize its essential features and develop links with concepts and tools of Control Theory.

In the first area, the dominant notions have been time evolution and the cascade design evolution. The overall philosophy for the time evolution has been that there is an evolution from early to late design which is accompanied by an evolution of model structure and associated system properties. The approach followed in cascade design is that each particular design stage shapes a local model; the structure of this local model has important implications on what can be achieved at the next design stage, and it thus determines overall cost, operability, safety and performance of the final process. Structural properties and thus performance, operability, etc. characteristics evolve, but not in a simple manner. This evolution of structure and related potential for delivering certain level of performance is only partially understood for the linear case. We would like to drive the model evolution along paths avoiding the formation of undesirable structural features and where possible to assign desirable characteristics and values. In the effort to formulate a generic system/control-based framework for both aspects of structure evolution, it is essential to address issues such as follows:

General Control Problems

(P.1) Characterization of desirable/undesirable performance characteristics and the limits of what can be achieved.
(P.2) Relate the best achievable performance characteristics to desirable system model structure.
(P.3) Define structure design/redesign problems for the different aspects of system structure.

These are traditional Control Theory tasks which are essential for intervening in the evolution problems related to design. Further aspects related to control of the design evolution process are as follows:

Nests of Variable Complexity Models and Design

(P.4) Development and parametrization of the family of minimal partial realizations (MPRs) according to complexity (MacMillan degree). Extension to nonlinear input–output descriptions.

(P.5) Investigate whether key system properties of the original system (stability–instability, minimum–non-minimum phase) are preserved within the chain and determine the degree of complexity required to preserve key system properties.
(P.6) Characterize the process of evolution of the Kronecker invariants in the MPR chain of models.
(P.7) Development of the structure and properties of the family of numerically nested models.

These problems are linked to the design time evolution family. For cascade design evolution, the emphasis is on the development of composite systems and the evolution of structure under systems instrumentation. Important issues are as follows:

Interconnected Systems and Complexity

(P.8) Explore the role of interconnection graph on structural and non-structural properties of composite systems. Specifically, examine the role of the "completeness" and "lack of completeness" assumptions on the composite system properties.
(P.9) Investigate the structural and non-structural system properties when the cardinality (input and output dimensions) of subsystems changes.
(P.10) Examine interconnected system properties when subprocesses are described by variable complexity models.
(P.11) Study interconnected system properties when the interconnection graph is augmented or loses part.

Structure Evolution in Systems Instrumentation

(P.12) Develop solutions to model orientation for matrix pencil and autoregressive models and study the structure evolution.
(P.13) Examine the structural properties under general input and output projection compensation.
(P.14) Evolution of Structure and Properties under Model Expansion.

The first of the above families of problems refer to generalized process synthesis, and the interconnection graph is central in the characterization of structure and related properties. The second deals with *systems instrumentation* and refers to the model shaping role of the selection of inputs and outputs and the shaping of the evolved system structure, as this is expressed in terms of structural invariants. Each one of the above areas has also a design dimension linked to the shaping of the system model structure. Ideally, we would like to assign desirable properties, but in reality it would be more relevant to avoid the formation of undesirable characteristics.

The area of Integration of Operations requires study of fundamental problems and development of new research areas which include study of emergent properties, system organization, multilevel modelling and control problems in complex hierarchies. Specific areas of interest involve the following:

Complexity in Integrated System Operations

(P.15) Understanding the derivation of the different functional models and how they are interfaced within the framework of different forms of system organization (hierarchical, holonic, etc.).
(P.16) Defining metrics and models for the characterization of emergent properties.
(P.17) Development of multilevel hybrid system theory by exploring the notions of global controllability and observability.

The new definition for the *SoS* is the starting point for the development of methodology that may lead to systematic design. Examining the rules of composition of the subsystems and their coordination as agents in a larger system defines a challenging new area for research and requires links across many disciplines. Examining in detail the special features of the different classes of *SoS* is crucial in the effort to provide a quantitative formulation of the notion of "*systems play*" which may take different forms in the different classes. This is also crucial in quantifying the notion of emergence in the *SoS* context. Key problems in the development of this field are as follows:

System of Systems

(P.18) Characterization of the family of *SoS* according to their origin (engineered, natural, social systems).
(P.19) Identifying the methodologies that can contribute to the characterization of *systems play*.
(P.20) Study the special aspects of emergence in the context of *SoS*.

The chapter has provided an identification of the challenges emerging within the two new classes of complex systems, the *SES* and *SoS*. By introducing the additional dimension of large dimensionality, the above classes take new dimensions. It is then that issues of organization, problem decomposition, decentralization and computational aspects take an additional significance. Design has been central to our study, but for many systems already designed in the past, redesign becomes a crucial issue. So far little effort has been spent in addressing this problem. We note, however, that versions of the above families of problems may be formulated when the redesign problem is considered, where either we want to modify the graph, the input, output structure, or the controller to achieve new requirements and objectives.

References

1. Alonso, E.: Actions and agents. In: Frankish, K., Ramsey, W.M. (eds.) The Cambridge Handbook of Artificial Intelligence, chap 5. Cambridge University Press (2013)
2. Antoulas, A.: On recursiveness and related topics in linear systems. IEEE Trans. Autom. Control **31**(12), 1121–1135 (1986)
3. Antoulas, A.C., Bishop, R.H.: Continued-fraction decomposition of linear systems in the state spaces. Syst. Control Lett. **9**(1), 43–53 (1987)
4. Antoulas, A.C., Matsūo, T., Yamamoto, Y.: Linear deterministic realization theory. In: Mathematical System Theory, pp. 191–212. Springer (1991)

5. Antsaklis, P., Kohn, W., Nerode, A., Sastry, S.: Hybrid Systems II, vol. 999. Springer Science & Business Media (1995)
6. Antsaklis, P.J., Michel, A.N.: Linear Systems. McGraw-Hill, New York (1997)
7. Aplevich, J.D.: Implicit Linear Systems. Lecture Notes in Control and Information Science, vol. 152. Springer, Berlin (1991)
8. Backx, A.: Engineering aspects of industrial applications of model-based control techniques and system theory. In: Essays on Control. Springer, pp 79–109 (1993)
9. Bar-Yam, Y.: Unifying Themes In Complex Systems, Volume 1: Proceedings of the First International Conference On Complex Systems. CRC Press (1997)
10. Bosgra, O.: On parametrizations for the minimal partial realization problem. Syst. Control Lett. **3**(4), 181–187 (1983)
11. Byrnes, C.I., Lindquist, A.: The stability and instability of partial realizations. Syst. Control Lett. **2**(2), 99–105 (1982)
12. De Vries, S., Vohra, R.V.: Combinatorial auctions: A survey. INFORMS J. Comput. **15**(3), 284–309 (2003)
13. Descusse, J., Lafay, J., Malabre, M.: Solution to Morgan's problem. IEEE Trans. Autom. Control **33**(8), 732–739 (1988)
14. Desoer, C.A., Callier, F.M.: Multivariable Feedback Systems. Springer, New York (1982)
15. Douglas, J.M.: Conceptual Design of Chemical Processes, vol. 1110. McGraw-Hill, New York (1988)
16. Dunbar, W.B., Murray, R.M.: Distributed receding horizon control for multi-vehicle formation stabilization. Automatica **42**(4), 549–558 (2006)
17. Eykhoff, P.: Finkelstein, L. (ed.) Concise Encyclopedia of Measurement & Instrumentation, vol. 7, pp. 137–142. Pergamon Press (1994)
18. Forney Jr., G.D.: Minimal bases of rational vector spaces, with applications to multivariable linear systems. SIAM J. Control **13**(3), 493–520 (1975)
19. Gantmacher, F.R.: The Theory of Matrices, vol. 2. Chelsea, New York (1959)
20. Grewal, M., Glover, K.: Identifiability of linear and nonlinear dynamical systems. IEEE Trans. Autom. Control **21**(6), 833–837 (1976)
21. Grigoriou, G.: Structure evolving systems: model structure evolution and system properties. Ph.D. thesis, City University (2012)
22. Hessami, A.G., Karcanias, N.: Integration of operations in process systems: complexity and emergent properties. In: 2011 IEEE International Systems Conference (SysCon), IEEE, pp. 466–471 (2011)
23. Jamshidi, M.: System of Systems Engineering: Innovations for the 21st Century, vol. 58. Wiley (2008)
24. Kaila, V.R., Annila, A.: Natural selection for least action. Proc. R. Soc. Lond. A Math. Phys. Eng. Sci. R. Soc. **464**, 3055–3070 (2008)
25. Kalman, R.: On partial realizations, transfer functions and canonical forms. Acta Polytech. Scand. **31**, 9–32 (1979)
26. Karampetakis, N., Pugh, A., Vardulakis, A.: Equivalence transformations of rational matrices and applications. Int. J. Control **59**(4), 1001–1020 (1994)
27. Karcanias, N.: Global process instrumentation-Issues and problems of a system and control theory framework. Measurement **14**(1), 103–113 (1994)
28. Karcanias, N.: Integrated process design: a generic control theory/design based framework. Comput. Ind. **26**(3), 291–301 (1995)
29. Karcanias, N.: Control problems in global process instrumentation: a structural approach. Comput. Chem. Eng. **20**, S1101–S1106 (1996)
30. Karcanias, N.: Multivariable poles and zeros. Control Syst. Robot. Autom. (2002)
31. Karcanias, N.: System concepts for general processes: specification of a new framework. Systems and Control Centre Research Report (2004)
32. Karcanias, N.: Structure evolving systems and control in integrated design. Ann. Rev. Control **32**(2), 161–182 (2008)

33. Karcanias, N.: Systems of systems: a control theoretic view. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics (SMC), IEEE, pp. 1732–1737 (2013)
34. Karcanias, N., Giannakopoulos, C.: Grassmann invariants, almost zeros and the determinantal zero, pole assignment problems of linear multivariable systems. Int. J. Control **40**(4), 673–698 (1984)
35. Karcanias, N., Giannakopoulos, C.: Necessary and sufficient conditions for zero assignment by constant squaring down. Linear Algebr. Appl. **122**, 415–446 (1989)
36. Karcanias, N., Hayton, G.: Generalised autonomous dynamical systems, algebraic duality and geometric theory. IFAC Proc. Vol. **14**(2), 289–294 (1981)
37. Karcanias, N., Hessami, A.G.: Complexity and the notion of system of systems: part (i): general systems and complexity. In: World Automation Congress (WAC), 2010, IEEE, pp. 1–7 (2010)
38. Karcanias, N., Hessami, A.G.: System of systems and emergence part 2: synergetic effects and emergence. In: 2011 Fourth International Conference on Emerging Trends in Engineering & Technology, IEEE, pp. 33–38 (2011)
39. Karcanias, N., Sagianos, E.: Small numbers, induced model nesting and system properties. Systems and Control Research Report, School of Engineering and Mathematics, City University (2008) (Sy&Co SEMS res rep 06-08/01)
40. Karcanias, N., Vafiadis, K.: Effective transfer function models by input, output variables reduction. IFAC Proc. Vol. **34**(13), 59–64 (2001)
41. Karcanias, N., Vafiadis, K.: Model orientation and well conditioning of system models: system and control issues. IFAC Proc. Vol. **35**(1), 315–320 (2002)
42. Karcanias, N., Sagianos, E., Milonidis, E.: Structured transfer function matrices and integer matrices: the computation of the generic McMillan degree and infinite zero structure. Int. J. Control **80**(9), 1404–1420 (2007)
43. Kouvaritakis, B., MacFarlane, A.: Geometric approach to analysis and synthesis of system zeros. Part 1. Square systems. Int. J. Control **23**(2), 149–166
44. Leventides, J., Karcanias, N.: Structured squaring down and zero assignment. Int. J. Control **81**(2), 294–306 (2008)
45. Lewis, F.L.: Optimal Estimation: with an Introduction to Stochastic Control Theory. Wiley, New York (1986)
46. Lewis, F.L.: A tutorial on the geometric analysis of linear time-invariant implicit systems. Automatica **28**(1), 119–137 (1992)
47. Ljung, L.: System Identification: Theory for the User. Prentice-Hall (1987)
48. Loiseau, J.J., Mondié, S., Zaballa, I., Zagalak, P.: Assigning the Kronecker invariants of a matrix pencil by row or column completions. Linear Algebr. Appl. **278**(1–3), 327–336 (1998)
49. MacFarlane, A.G.: Information, knowledge and control. In: Essays on Control; Perspectives in Theory and its Applications, Birkhauser, Berlin (1993)
50. Maier, M.W.: Architecting principles for systems-of-systems. Syst. Eng. J. Int. Counc. Syst. Eng. **1**(4), 267–284 (1998)
51. Manthorpe, W.H.: The emerging joint system of systems: a systems engineering challenge and opportunity for APL. Johns Hopkins APL Tech. Dig. **17**(3), 305 (1996)
52. Mesarovic, M.D., Takahara, Y.: General Systems Theory: Mathematical Foundations, vol. 113. Academic Press (1974)
53. Mesarovic, M.D., Macko, D., Takahara, Y.: Theory of Hierarchical, Multilevel, Systems, vol. 68. Elsevier (1970)
54. Milias-Argeitis, A., Summers, S., Stewart-Ornstein, J., Zuleta, I., Pincus, D., El-Samad, H., Khammash, M., Lygeros, J.: In silico feedback for in vivo regulation of a gene expression circuit. Nat. Biotechn. **29**(12), 1114 (2011)
55. Morari, M., Arkun, Y., Stephanopoulos, G.: Studies in the synthesis of control structures for chemical processes: part I: formulation of the problem. Process decomposition and the classification of the control tasks. Analysis of the optimizing control structures. AIChE J. **26**(2), 220–232 (1980)
56. Nourian, M., Caines, P.E., Malhame, R.P., Huang, M.: Nash, social and centralized solutions to consensus problems via mean field control theory. IEEE Trans. Autom. Control **58**(3), 639–653 (2012)

57. Perkins, J.D.: Interactions Between Process Design and Process Control. Elsevier (2014)
58. Rijnsdorp, J.E.: Integrated Process Control and Automation. Elsevier Science Publishers (1991)
59. Rosenbrock, H., Rowe, A.: Allocation of poles and zeros. Proc. Inst. Electr. Eng. IET **117**, 1879–1886 (1970)
60. Rosenbrock, H.H.: State-Space and Multivariable Theory. Nelson (1970)
61. Saberi, A., Sannuti, P.: Squaring down of non-strictly proper systems. Int. J. Control **51**(3), 621–629 (1990)
62. van der Schaft, A., Jeltsema, D., et al.: Port-Hamiltonian systems theory: an introductory overview. In: Foundations and Trends® in Systems and Control **1**(2–3), 173–378 (2014)
63. SESDIP: Structural evaluation of synthesis of distributed industrial processes, p. 8924. Technical report, ESPRIT III Basic research Project (1996)
64. Smith, M.C.: Synthesis of mechanical networks: the inerter. In: Proceedings of the 41st IEEE Conference on Decision and Control, 2002, IEEE, vol. 2, pp. 1657–1662 (2002)
65. Tenner, A.R., DeToro, I.J.: Process Redesign: The Implementation Guide for Managers. Addison-Wesley Publishing (1996)
66. Tether, A.: Construction of minimal linear state-variable models from finite input-output data. IEEE Trans. Autom. Control **15**(4), 427–436 (1970)
67. Van Brussel, H., Bongaerts, L., Wyns, J., Valckenaers, P., Van Ginderachter, T.: A conceptual framework for holonic manufacturing: identification of manufacturing holons. J. Manuf. Syst. **18**(1), 35–52 (1999)
68. Voos, H.: Resource allocation in continuous production using market-based multi-agent systems. In: 2007 5th IEEE International Conference on Industrial Informatics, IEEE, vol. 2, pp. 1085–1090 (2007)
69. Willems, J.C.: Models for dynamics. In: Dynamics Reported, pp. 171–269. Springer (1989)
70. Willems, J.C.: On interconnections, control and feedback. IEEE Trans. Autom.control **42**(3), 326–339 (1997)
71. Wooldridge, M.: An Introduction to Multiagent Systems. Wiley (2002)