

AN ONTOLOGICAL CONCEPTUALIZATION OF MULTISCALE MODELS

Aidong Yang^{*}, Wolfgang Marquardt

Lehrstuhl für Prozesstechnik, RWTH Aachen, 52064 Aachen, Germany

Abstract. Multiscale modeling has recently been widely used in various fields of science and engineering. A few research efforts have attempted to generalize existing applications mainly by means of classifying multiscale modeling paradigms, to reach a general understanding of multiscale modeling. However, these existing efforts have been largely inductive and abstracting from the applications the researchers were familiar with. This has inevitably caused inconsistencies within the results of different researchers. Further, existing work has not provided a precise definition for the fundamental concepts such as scale, which arguably has been the main source of confusion. In this work, we attempt to address this issue through developing a coherent conceptualization framework. Instead of starting with existing applications, our work is built upon a conceptualization of general systems, from which precise definitions of scale and inter-scale relations are derived. This is further followed by a precise specification of the composition of a multiscale model and subsequently a classification of multiscale models, all through rigorous definitions. This conceptualization framework is represented in form of an ontology by means of a set-theoretic language. It reuses part of an existing ontology (the BWV ontology). The key concepts in this framework are illustrated through two modeling examples judiciously chosen from the field of chemical engineering. In comparison with existing results, the key contribution of this framework is not a classification but rather a generic conceptual foundation for a better understanding of multiscale modeling, which is intended to be sharable by researchers from different backgrounds. On the other hand, it is demonstrated that a rigorously defined classification can be derived from this conceptual foundation. It is also shown that the classifications of multiscale modeling proposed in other works can be interpreted by means of the suggested conceptualization framework in a precise manner.

Keywords. Multiscale modeling, conceptualization framework, ontology, systems, classification

1. Introduction

In general, a mathematical model of a system is a collection of relations (usually in the form of mathematical equations) among variables representing certain properties of the system measured in particular domains. By

^{*} Corresponding author. Current address: Faculty of Engineering and Physical sciences (J2), University of Surrey, Guildford GU2 7XH, UK. E-mail: a.yang@surrey.ac.uk

solving a model, one can predict values of some properties of the system given the values of some other properties, or predict distributions of the values of some properties in particular domains, given certain boundary conditions. To meet the demands from science and engineering on the depth of revealing the characteristics of a system, a mathematical model may involve relations of properties at different scales of the system. For example, in order to predict certain properties of a material, both the continuum scale as well as the molecular scale might have to be considered.

In recent years, multiscale modeling has become a popular topic in different fields; the targets being modeled range from engineering systems (e.g. chemical reactors) to materials and biological systems, to name the frequently reported ones. This status is well reflected by several review papers, namely two from the chemical engineering community by Pantelides (2001) and more recently by Vlachos (2005) and one from the applied mathematics community by Givon et al. (2004). A large number of applications from various domains have been referenced in these reviews. Additionally, there have been special journal issues dedicated to the topic of multi-scale modeling, such as Issues 8 and 9 in Volume 59 of Chemical Engineering Science. The rapidly growing interests in multiscale modeling have even led to the creation of two new journals in 2003, namely Multiscale Modeling and Simulation (by SIAM) and the International Journal on Multiscale Computational Engineering (by Begell House Publishers). The former was largely initiated by the applied mathematics and the latter by engineering science communities.

Accompanying the widespread applications, there have been several generalization efforts towards a better understanding of and improvements to the multiscale modeling paradigms in general. For example, E and Engquist (2003 a, b) present, from a mathematical perspective, a so-called heterogeneous multiscale method which generalizes a variety of applications of multiscale modeling and computation. They also formulate a general procedure of applying a heterogeneous multiscale method and analyze the stability and accuracy of numerical solutions following such a method. Within the chemical engineering community, generalization is pursued through several attempts on classifying existing multiscale modeling methods. Pantelides (2001) presents a classification on scale integration strategies. The notions introduced were applied by Bezzo et al. (2004) in presenting a general formulation of hybrid multizonal/CFD models of chemical process equipment. The classification was later expanded by Ingram et al (2004) who identify a number of integrating frameworks and discuss the compatibility issue in applying them. More recent work on classification has been reported during the preparation of this manuscript, e.g. by Li et al. (2005) and by Vlachos (2005).

The contributions cited above are all generalization-oriented and have set a starting point towards a sound theory of multiscale modeling, which is important for making methodological improvements and for fostering new

applications. However, these contributions are largely result of inductive summary on the applications in one or several fields that their authors are familiar with. Consequently, it has been inevitable that these proposed frameworks and classifications appear to be overlapping and inconsistent. Further, such inductive studies can hardly touch on the fundamental questions such as what a scale exactly means. We would argue that answers to such questions are essential for resolving inconsistencies with respect to the understanding, classification, and representation of multiscale modeling paradigms.

Recognizing the limitations of the existing studies, we propose to take a different approach to improve our understanding of multiscale modeling. Instead of directly abstracting a variety of multiscale modeling applications, we firstly direct our attention onto the conceptualization of general systems. This is reasonable because the target of a multiscale modeling task, whichever application or field it is originating from, will always be some kind of system in the sense of general systems theory as introduced by Bertalanffy (1968). Secondly, we derive, from the basic concepts of systems, rigorous definitions of notions of interest in multiscale modeling, including scale and inter-scale relationships. As the result of these two steps, a generic conceptual foundation independent of concrete applications and fields is rendered to support the understanding of various multiscale modeling paradigms. Thirdly and finally, we make use of this conceptual foundation and look into the compositions of multiscale models in terms of scales and inter-scale relations. The differences in multiscale models naturally lead to different model classes.

In contrast to existing work, such an approach to the understanding of multiscale modeling appears to be more deductive than inductive. However, we have to admit that our conceptualization framework is still influenced and perhaps limited by existing multiscale modeling applications which we are familiar with, particularly those from our own field, namely Chemical Engineering. Thus, this contribution should not be regarded as one that offers a complete and universally valid classification of multiscale modeling paradigms. Instead, our goal is to provide a coherent framework for conceptualization which (1) renders a generic yet rigorous conceptual foundation for understanding multiscale modeling that hopefully can be shared by other parties, and (2) demonstrates how multiscale models, existing or new, can be precisely represented and classified on the basis of such a foundation. In other words, this framework is intended to be a basis for a common understanding by people from different communities and to be open for further extensions.

As indicated above, our approach starts with a conceptualization of general systems. In Computer Sciences, an explicit specification of a conceptualization of a domain of discourse is called an ontology (Gruber, 1993). One of the roles an ontology may assume is clarifying the meaning of concepts in the context of knowledge representation

(cf. Guarino, 1994). This role is very relevant to this work because the clarification of the meaning of a scale, among others, is the key to a precise understanding to multiscale modeling. An ontology that assumes such a role is usually composed of the definitions of a set of concepts and their relations (Uschold, Grüninger, 1996). In this work, our approach is implemented essentially by developing such an ontology which starts with a conceptualization of general systems (i.e. the first step). For this purpose we have opted to extend and adapt part of an existing ontology, which was originally formulated by Bunge (1977, 1979), later adapted by Wand and Weber (1990) and referred to as the Bunge-Wand-Weber (BWW for short) ontological model in the recent literature (e.g. Rosemann and Green, 2002). This existing ontology is selected as the basis of our own work, because it gives precise definitions of basic concepts, from which it derives very logically the concept of a system and its characterizations.

In the rest of the paper, the entire conceptualization framework for multiscale models is presented through rigorous definitions and illustrated by examples. In Section 2, two examples from chemical process modeling are briefly introduced. They will be referred to throughout the paper. In Section 3, the conceptualization framework (i.e. the ‘theory’ part of the paper) is presented following the three steps as mentioned above. In Section 4, the conceptualization framework is applied to the two chemical process modeling examples in a thorough manner. A comparison with existing work is made in Section 5, where we try to demonstrate how the conceptualization framework can be used to interpret and unify diverse terminologies and classifications by different authors. Thus, Sections 4 and 5 can be viewed as the ‘application’ part of the paper. In particular, Section 5 may hopefully convey the idea that our conceptualization framework provides a sharable foundation of understanding and is open for extensions. Finally, some concluding remarks are given, including those with regard to the potential usage of the results of this work.

2. Two illustrative examples

Two examples of multiscale modeling of chemical processes are judiciously chosen from the literature. Example 1 models a system on different scales to simplify computations, while a physical interpretation of the scales considered in modeling exists in Example 2. A more precise account on the difference between these two types of models will be given later in the paper. In the following, these two examples are briefly introduced; details of the mathematical formulation of the models can be found in Section 4.

2.1 Example 1

The first example is a so-called hybrid multizonal/Computational Fluid Dynamics (CFD) model of chemical process equipment (Bezzo et al., 2004). As schematically shown in Figure 1, a piece of equipment (e.g. a vessel) is first spatially decomposed into a number of zones. Each zone is characterized by a number of variables such as

temperature, pressure, and so on. Generally, these variables have different values in different zones, to reflect the spatial heterogeneity of the overall equipment. Furthermore, each zone is spatially decomposed into a number of cells. Each cell is again characterized by a set of variables which often complies with the one for a zone. For different cells in one zone, these variables may also have different values.

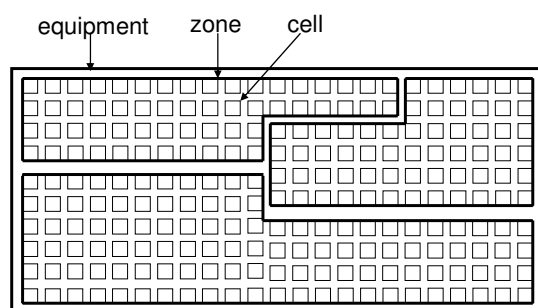


Figure 1. Modeling of a piece of chemical equipment.

The reason why the piece of equipment is decomposed on two different levels, is because some phenomena (e.g. fluid dynamics) must be modeled at the level of cells in order to ensure the precision of modeling. For other phenomena (e.g. chemical or particulate kinetics), it may be suitable to model them at the level of zones for reducing computational costs while the expected precision of modeling is still achieved. This modeling approach has been used in e.g. crystallizers modeling (Kulikov et al., 2005). It is worth noting that within this modeling paradigm, the system is of a continuum nature. The decomposition of the system into different levels is carried out only for computational reasons.

2.2 Example 2

The second example is a homogeneous-heterogeneous chemical reactor model (Vlachos, 1997). As schematically shown in Figure 2, the reactor is considered to have two parts: the (homogeneous) bulk fluid phase and the (heterogeneous) solid catalyst surface. The solid surface is the place where adsorption, reaction, and desorption of molecules occur. The entire reactor is modeled by adopting a continuum model for the bulk fluid phase and the solid surface. In the fluid phase, the reactants of the surface reaction diffuse towards the solid surface, and the products of the surface reaction diffuse from the solid surface into the bulk fluid phase. The model for the solid surface relates the mass conservation at the surface to the rates of the surface kinetics phenomena. To generate important kinetics information as required by an accurate continuum model for the surface, the surface is further decomposed through a molecular lattice which comprises a number of sites of a particular spatial layout. This molecular lattice is not a continuum model. Instead, it is treated by a Monte-Carlo molecular simulation. Note that the relationship between the continuum model of the surface and its molecular lattice model is different from that

between the model of a zone and the model of a cell in Example 1; the former involves different types of laws applied by the two models while the latter does not.

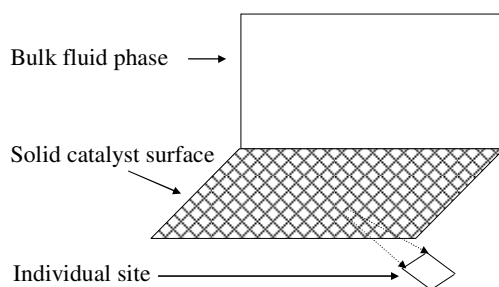


Figure 2. Modeling of a chemical reactor.

3. The conceptualization framework

As indicated earlier, the conceptualization framework is developed on the basis of the BWV ontology which has been specified using a rigorous set-theoretic language by the original authors. This specification has received some criticism on the lack of understandability (Rosemann and Green, 2002). However, such a language is able to convey the semantics of the concepts in the ontology with a high degree of precision. To preserve this advantage, such a language is still adopted in this current work. Regarding the ontology's scope, the original work by Bunge (1977, 1979) was intended to represent general systems for general purposes. The adaptation by Wand and Weber (1990) was done later for providing an ontological model of information systems. In this current work, we have so far focused on a conceptualization that allows us to formulate general multiscale system models. Accordingly, the ontology presented here only addresses this particular purpose instead of providing a complete characterization of systems.

This current work is mainly based on the original ontology by Bunge (1977, 1979). At a few occasions, the work by Wand and Weber (1990) is reused. For a definition involved in the ontology, links to the original references (if any) are always given. Further, wherever the adaptation to the original formulation is significant, the definitions are marked with “*”.

The conceptualization framework is organized according to the three steps to an understanding of multiscale modeling introduced in Section 1. Consequently, the framework is composed of three parts, namely *basic concepts of a general system*, *scales and inter-scale relations*, and *multiscale models of systems*, as schematically shown in Figure 3.

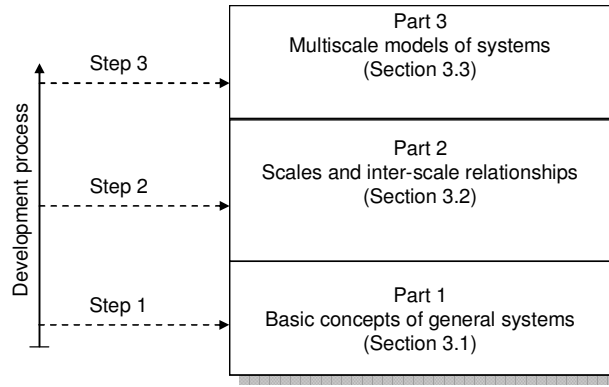


Figure 3. Three parts of the conceptualization framework. Part 1 and Part 2 (shaded) together are referred to as the “conceptual foundation” of the framework.

3.1 Basic concepts of a general system

3.1.1 Things, couplings, and laws

In this subsection, basic notions about things, their couplings and the laws applicable to them are presented. They provide the basis for defining systems in Section 3.1.2.

To get started, two fundamental concepts are taken for granted without being rigorously defined: *substantial individuals* are individuals that exist; properties of any substantial individual are called substantial properties, or simply *properties*. Based on these two concepts, the concept of a *thing* is introduced as the pair of a substantial individual and the set of its properties. For example, a chemical reactor as a thing refers to the individual of the reactor as well as its properties such as volume, conversion, yield, etc.

Definition 1 (B-I 3.1)¹: Thing. Let $s \in S$ be a substantial individual and call $p \subset P$ the collection of its (unarized) properties. Then, the individual together with its properties is called the *thing* x :

$$x := \langle s, p(s) \rangle.$$

To denote the complete set of things, we introduce

Definition 2 (B-I 3.2): Totality of things. The totality of things is called Θ :

$$\Theta := \{ \langle s, p(s) \rangle \mid s \in S \wedge p(s) \subset P \}.$$

¹ To show precise links to the references including Bunge (1977), Bunge (1979) and Wand & Weber (1990), these references are marked as “B-I”, “B-II”, and “WW”, respectively. The reference number of the referenced item as appearing in the original reference is appended after these labels. E.g. (B-I 3.1) implies a reference to Definition 3.1 given by Bunge (1977).

We now examine the *part-of* relation between two different things on the basis of the concept of a *juxtaposition of things*. A juxtaposition of things x and y is defined as the pair of the juxtaposition of the substantial individuals represented by x and y and that of the sets of properties of the two substantial individuals.

Definition 3 (B-I 3.3): Juxtaposition of things. Let $x = \langle s_1, p(s_1) \rangle$ and $y = \langle s_2, p(s_2) \rangle$ be two things. Then the juxtaposition of x and y is the third thing $z = x \dot{+} y = \langle s_1 \dot{+} s_2, p(s_1 \dot{+} s_2) \rangle$, where $\dot{+}$ represents association or juxtaposition.

Thing x is a part of thing y if and only if the juxtaposition of x and y is equal to y . This is reflected by

Definition 4 (B-I 3.4): Part-of relation. Let $x, y \in \Theta$ be things. The x is a part of y iff $x \dot{+} y = y$. Symbol for the part-whole relation is \sqsubset . Its negation is marked by $\not\sqsubset$.

This definition basically states that if no factual change or novel information arises when associating (or juxtaposing) one thing (e.g. one fluid phase of a multiphase reactor) with another thing (e.g. the reactor to which the fluid phase belongs), then the former is deemed to be part of the latter.

With this part-of relation between things, the *composition of a thing* is defined by means of

Definition 5: Composition of a thing. Let $x \in \Theta$ be a thing. The composition of x is the set of its parts, i.e.

$$C(x) = \{y \in \Theta \mid y \sqsubset x\}^2.$$

Note that, according to such a definition, some parts of a thing may appear in the composition of the thing together with parts of their own. For example, if the thing x has the structure shown in Figure 4, then $C(x) = \{a, b, c, d, e, f\}$.

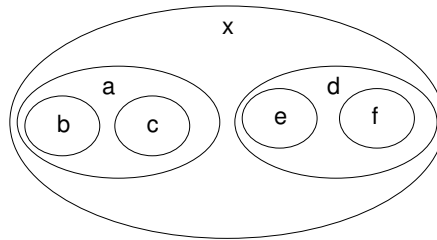


Figure 4. Composition of a thing.

Now we characterize things in greater detail by representing their properties and defining their states.

² The composition of a thing may change over time, so does the relation with other things. Considering this in this definition, the notation $C(x)$ should rather be $C(x, t)$. However, the argument “ t ” is omitted here for clarity. The same simplification is used also in the definitions of any system’s composition and structure in Section 3.1.2.

Definition 6 (B-I 3.6): Functional schema. Let $x = \langle s, p(s) \rangle$ be a thing of class $T \subseteq \Theta$. A functional schema x_m of x is a certain nonempty set M together with a finite sequence F of nonpropositional functions on M , each of which represents a property of T . Briefly,

$x_m := \langle M, F \rangle$, where

$F = \langle F_i \mid F_i \text{ is a function on } M, 1 \leq i \leq n \rangle$.

Definition 7 (B-I 3.9): State function, State. Let x be a thing modeled by a functional schema $x_m = \langle M, F \rangle$, and suppose that each component of the function

$F = \langle F_1, F_2, \dots, F_n \rangle: M \rightarrow V_1 \times V_2 \times \dots \times V_n$

represents a property of x , where $V_i, i=1, \dots, n$ represents a dimension of M . The F_i , for $1 \leq i \leq n$, is called the i th *state function* of x , F the *total state function* for x , and its value

$F(m) = \langle F_1(m), F_2(m), \dots, F_n(m) \rangle$

for $m \in M$ is said to represent the *state* of x at m .

To illustrate Definitions 6 and 7, consider a room as a thing. The room has a number of properties or formally *state functions* such as temperature (F_1), pressure (F_2), and moisture (F_3). The value of each of these state functions is determined by the specific time (as an instance of V_1) and the spatial location (as an instance of V_2) where the value is measured. The time and spatial location together form a particular m .

The above two definitions are comparable to some notions adopted by Klir in the context of general systems (Klir, 1985). In particular, a *state function* is equivalent to what is termed as “variable” by Klir for an operational representation of a property in terms of a specific measurement or observation. An element m in *domain* M is called a “support set” which represents a concrete backdrop (e.g. in terms of time, space, etc.) in which a measurement or an observation on a property is conducted.

To reflect the fact that not every arbitrary state of a thing is necessarily realistic, we further introduce the concept of law:

Definition 8 (B-I 3.10): Law³. Let $x_m = \langle M, F \rangle$ be a functional schema for a thing x . Any restriction on the possible values of the components of F and any relation among two or more such components is called a *law*, denoted by $l(x)$. The **totality of the laws** of x is defined as:

$$L(x) := \{l_i(x), 1 \leq i \leq n\}$$

An example for a law which represents a restriction on the possible value of a state function is ‘the temperature of a room can not be lower than 20 degrees’. An example for a law which represents a relation between two state function is ‘the pressure of a (closed) room is proportional to its temperature’.

Remark 8: We call a set of things as of a certain **type** if these things have the same *total state function* and the same *totality of laws*.

According to this notation, the zone and the cell described in Example 1 (Section 2) are of the same type. Although a zone and a cell are characterized by a different sets of laws in the example, this has been done due to the computational concerns. From a physical point of view, any state function applicable to a cell is always applicable to a zone, and the same is true with respect to the applicable laws. This is because both a zone and a cell are of the same continuum nature. On the contrary, a site in the molecular lattice and the solid surface as a whole, as involved in Example 2, are of different types: a site has state functions such as occupation-site function and local coordination (cf. Section 4.2.2 for a detailed explanation of the model of a site), which are not applicable to the solid surface as a whole. Similar differences exist also with respect to the set of applicable laws.

Based on the notion of the state of a thing, Bunge (1977) further introduces the concepts of *event* (as a change of states) and *history* (as a collection of events in time). These concepts in turn allow a definition of the notion of *actions* between two things – thing x *acts on* thing y iff the existence of x changes the *history* of y . Omitting these detailed definitions, here we directly come to

Definition 9: (B-II, Page 6⁴) Coupled things. Let x and y be two things. They are said to *be coupled* iff x acts on y and/or y acts on x . Such coupled things are denoted by $b(x,y)$.

For example, a fluid phase and a solid phase in a reactor are considered as two coupled things, if there exists mass transfer (as an inter-action) between these two phases. Note that two things can be coupled through multiple actions. For this consideration, we further introduce

³ Bunge used the term ‘Law Statement’, which is shortened to ‘Law’ in this paper for simplicity.

⁴ This is the page number in which the semantics of this item is discussed in the original reference.

Definition 10: (B-II, Page 6) Coupling. Let x and y be coupled things. An individual coupling relation, or simply a *coupling* between x and y is denoted by $c(x,y)$. The totality of the couplings between x and y is denoted by $C(x, y) := \{c_i(x, y), 1 \leq i \leq n\}$.

According to this definition, the totality of the couplings between two chemical plants, for example, may include all the material flows between them; each flow is regarded as one distinct coupling. In the following, we examine the influence of couplings between things on their state functions. It is recognized that some properties of a thing exist only because of the existence of certain coupling(s). For example, the amount of material or energy leaving from or entering into a plant arises due to a certain material or energy exchange with another plant which leads to a coupling between the two plants. If there is no such material or energy connection between two plants, this property will not be considered. To distinguish such properties or state functions from the rest, we introduce the following

Definition 11: Coupling-induced state function. Let x be a thing coupled with at least one other thing. Any state function of x is called a *coupling-induced state function*, denoted by F_k^c , iff it represents a property of x that occurs only due to the coupling $c(x,y)$ between x and another thing y . Such a property is denoted as p_i^c . By definition, $p_i^c \in (p(x \mid y \in \Theta \wedge b(x, y)) - p(x))$.

In general, the characteristics of a coupling, manifested via the coupling-induced state functions of two coupled things, can be influenced by the characteristics of the two things independent of the coupling between them. These characteristics are manifested via *non-coupling-induced state functions* of the two things. Such influences are expressed via a *coupling mechanism law* as defined by

Definition 12: Coupling mechanism law. Let x and y be two coupled things. Let $c(x, y) \in C(x, y)$ be an individual coupling between x and y . Then, a *coupling mechanism law*, denoted as $l^c(x,y)$ is defined as any relation that maps non-coupling-induced state functions of x and/or y to coupling-induced state functions of x or y induced by $c(x,y)$. The totality of the coupling mechanism laws related to a coupling $c(x,y)$ is defined as

$$L^c(c(x, y)) := \{l_i^c(x, y), 1 \leq i \leq n\}.$$

The term ‘mechanism’ is used here, because such a law usually explains why and how the interaction occurs between the two parties involved in a coupling. Examples of coupling mechanism laws can be easily found in the studies of transport phenomena (Bird, et al., 1960). For instance, the rate of heat transfer (as a *coupling-induced state function*) between two objects is proportional to the difference of temperatures (as *non-coupling-induced state functions*) of the objects.

While a *coupling mechanism law* relates coupling-induced state functions to the characteristics of two coupled things, a *topological connection law* on a given coupling relates a coupling-induced state function of one thing to one of another thing:

Definition 13: Topological connection law. Let x and y be two coupled things. Let $c(x, y) \in C(x, y)$ be an individual coupling between x and y . Let f_x^c be any coupling-induced state function of x induced by this coupling and f_y^c be any coupling-induced state function of y induced by this coupling. Then, a *topological connection law*, denoted as $l(x, y)$ is defined as any relation between f_x^c and f_y^c . The totality of the topological connection laws on the coupling $c(x, y)$ is defined as

$$L^c(c(x, y)) := \{l_i^c(x, y), 1 \leq i \leq n\}.$$

Typically, a topological connection law on a coupling *equates* a coupling-induced state function of one thing to that of another thing. Consider for example two chemical plants coupled with a pipe. The amount of material flowing out of plant A through this pipe is equal to the amount of the material flowing into plant B. The term *topological* refers to the relations between two connected things, none of which is part of another one. It is used here in contrast to *mereological* which refers to relations between one thing and one or more things which are part of the former. Using the term *topological* is appropriate here because a connection law as defined above is for specifying the link between two parties which are not part of each other. Otherwise the connection law would be *mereological*. The semantics of such laws will be defined later (cf. Definition 27).

3.1.2 Systems and subsystems

On the basis of the notion of a thing and related concepts, we define and characterize systems and subsystems in this subsection. A system is essentially defined as a set of things and their connections. Rigorously, we introduce

Definition 14 (WW 10): System. Let C be a set of things (i.e. $C \subseteq \Theta$), and let $B_c = \{(x, y) \mid x, y \in C \wedge b(x, y)\}$. Let $\sigma(C, B_c)$ be a graph, where C is the set of vertices and B_c the set of edges. Then $\sigma(C, B_c)$ is a system denoted by σ , iff it is a connected graph.

This definition basically states that a system is the collection of things, any of which must be connected to one or more of things of the collection. For example, a chemical reactor that has a number of phases is considered as a system, only when every phase is linked to at least one other phase in the reactor.

Corollary 14. Any system σ is a thing, i.e. $\sigma \in \Theta$.

In subsection 3.1.1, we have introduced the concept of composition of a thing through Definition 5. For convenience, we define this concept again particularly for a system and also introduce the term *component*.

Definition 15* (B-II 1.2): System's composition. Let σ be a system. The composition of σ is the set of things in σ :

$$C(\sigma) := \{x \mid x \in \Theta \wedge x \sqsubset \sigma\}.$$

Remark 15: Any element in the composition of the system σ , denoted by c , is called a **component** of σ .

Still using a multiphase reactor as an example, the composition of this system is the phases of the reactor as they are all the *things* that are *part of* the reactor.

In the original formulation of Bunge (1979), the notion of atomic composition (or A-composition) is employed, to represent a collection of parts of a thing or a system, if these parts are atomic, i.e. if they are not intended to be further decomposed according to the nature of the system studied or to the purpose of the study. According to Bunge, the atomic parts of a system are things of the same class. This is problematic, because there exist *hybrid* systems such as human-machine systems and software-hardware systems which are composed of atomic parts of different classes of things. To avoid this problem, we redefine the notion of A-composition through the following:

Definition 16: A-component. Let σ be a system, and c be a component of σ . c is an *atomic component* or an *A-component* of σ , denoted by c_A , iff c does not contain any other components of σ , i.e.

$$c_A \in C(\sigma) \ \& \ c_A \notin \{x \in C(\sigma) \mid \exists y \in C(\sigma), y \neq x, y \sqsubset x\}.$$

Definition 17*(B-II 1.2): A-composition. Let σ be a system. The composition of σ in terms of all A-components of σ is called the *A-composition* of σ , denoted by C_A :

$$C_A(\sigma) := \{c_{A,i}, 1 \leq i \leq n_A\}.$$

Definitions 16 and 17 may be illustrated by one of the previous examples. If the thing x in Figure 4 is a system σ (thus assuming connectivity between the components of x), then b , c , e , and f are the A-components of x , but b and d are not. The A-composition of x is $C_A(\sigma) = \{b, c, e, f\}$.

Employing the notion of A-component as defined above, the concepts A-environment and A-structure are defined as follows:

Definition 18 (B-II 1.2): A-environment. Let σ be a system. The A-environment of σ is the set of all things which are not components of σ , but are coupled with some A-components of σ :

$$E_A(\sigma) := \{x \mid x \notin C(\sigma) \wedge (\exists y)[y \in C_A(\sigma) \& b(x, y)]\}.$$

Definition 19 (B-II 1.2): A-structure. Let σ be a system. The *A-structure* of σ is the set of *couplings among* the A-components of σ , and among them and the things in the A-environment of σ :

$$S_A(\sigma) := B_A(\sigma) \cup \hat{B}_A(\sigma),$$

where

$$B_A(\sigma) = \{c(x, y) \mid x, y \in C_A(\sigma)\} \neq \Phi,$$

$$\hat{B}_A(\sigma) = \{c(x, y) \mid x \in C_A(\sigma) \wedge y \in E_A(\sigma)\}.$$

Hence, $B_A(\sigma)$ is the set of coupling relations among A-components of σ , and $\hat{B}_A(\sigma)$ is the set of coupling relations between σ and its A-environment.

To illustrate the definitions of A-environment and A-structure, consider the computer network system of a university, as a system composed of inter-connected PCs, and all the external devices connected to particular PCs of the system. If the PCs are not to be further decomposed within a particular context of investigation, i.e. if they are considered as A-components of the system, then these external devices are the A-environment of the system. A-structure of the system is the collection of all the connections between the PCs in the computer network system and the links between the external devices and the PCs to which these external devices are connected.

Now we are ready to define a subsystem by using the concepts of A-composition, A-environment, and A-structure:

Definition 20 (B-II 1.6, WW 12): Subsystem. Let σ be a system with A-composition $C_A(\sigma)$, A-environment $E_A(\sigma)$, and A-structure $S_A(\sigma)$. Then x is a subsystem of σ , iff x is a system, and

$$\begin{aligned} &[C_A(x) \subseteq C_A(\sigma)] \wedge \\ &[E_A(x) \subseteq \{E_A(\sigma) \cup \{C_A(\sigma) - C_A(x)\}\}] \wedge \\ &[S_A(x) \subseteq S_A(\sigma)]. \end{aligned}$$

The latter conditions state that (i) the A-composition of a subsystem x of system σ is part of (or equal to) the A-composition of σ , (ii) the A-environment of subsystem x is part of (or equal to) the summation of the A-environment of σ and the part of the A-composition of σ other than the A-composition of this subsystem, and (iii) the structure of the subsystem x is part of (or equal to) that of σ .

Still using the university computer network example, one can state that a computer cluster of the university is a subsystem of the entire network. Here, the PCs in the cluster form a subset of the entire connection of PCs in the entire network. The environment of the cluster comprises elements from the rest of the network and/or those from the environment of the entire network. The connections between the PCs in the cluster are part of the connections between all the PCs in the network.

3.2 Scales and inter-scale relations

When a system is investigated for a certain purpose, a particular decomposition of it might be taken, either following numerical principles (e.g. finite volume discretisation in CFD modeling) or accounting for the well-recognized, “natural” kinds of the components of the system (e.g. continuum, molecular, and atomic levels of material). Generally speaking, decomposition of a system can result in multiple levels of abstraction, each of which is composed of components of a particular granularity (in space, time, etc.). In the context of modeling, these levels have been conventionally called “scales”, giving rise to the term of multiscale modeling. In this subsection, we attempt to clarify the semantics of levels or scales and identify typical inter-scale relations.

We first introduce the concepts of level (or scale) and level hierarchy. Intuitively, a level of a system represents part of a certain decomposition schema. In a multilevel system, one level is said to be “lower” than another one, if any component in the former is always part of a particular component in the latter. For instance, in Example 1 introduced in Section 2, the level of cells is lower than the level of zones, because any cell is always a component of a particular zone. Rigorously, we introduce

Definition 21 (B-II 1.8): Level (or scale) and level hierarchy of a system⁵. Let σ be a system. Let $V = \{V_i \mid 1 \leq i \leq n\}$ be a family of n nonempty sets of things, $V_i \subseteq C(\sigma)$. Further, $\forall x_i, x_j \in V_i, x_i \sqsubseteq x_j$. Then,

(i) $(V_i < V_j) := (\forall x)[x \in V_j \Rightarrow (\exists y)(y \in V_i \wedge y \in C(x))]$.

(ii) $H = \langle V, < \rangle$ is called a *level hierarchy* of σ , and any $V_i \in V$ is called a *level (or scale)* of σ .

We have used the convention that V_1 is the level at the bottom of the level hierarchy, i.e. the components on this level are not further decomposed. V_2 is the level immediately above V_1 . Following this convention, for $1 \leq i \leq n$, V_n becomes the level at the top of the level hierarchy.

⁵ Bunge (1979) uses the term “level structure” instead of “level hierarchy”. We have changed the term here to avoid possible terminological confusion between this concept and “L-Structure” introduced by Definition 23.

The above definition can be illustrated with the two examples introduced in Section 2. Both of the two examples adopt a 3-level hierarchy, as shown in Figures 5 and 6.

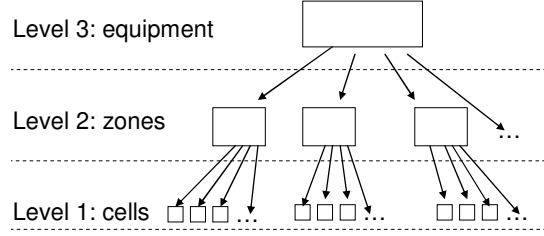


Figure 5. Level hierarchy of a chemical equipment represented by a hybrid multizonal/CFD model (Example 1).

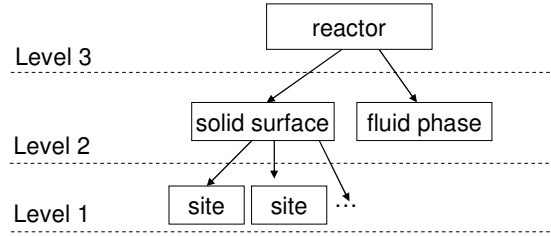


Figure 6. Level hierarchy of a chemical reactor represented by a heterogeneous-homogeneous model (Example 2).

Corollary 21a (B-II 1.8): For any $V_i, i > 1, x \in V_i \Rightarrow C(x) \subseteq \bigcup_{k=1}^{i-1} V_k$.

As an illustration to this corollary, one can see in Figure 6 that “reactor” at Level 3 has the following components: “solid surface” and “fluid phase” from Level 2, and the “sites” from Level 1. Besides, one can also see that Level 2 contains two subsystems of “reactor”, which gives an example for the following corollary.

Corollary 21b: Any level V_i of a system σ not at the bottom of the level hierarchy H must contain some subsystem(s) of σ . I.e.

$$\forall V_i \in V, i > 1 \Rightarrow \exists x \in V_i, x \prec \sigma.$$

Corollary 21c: It is possible that a level V_i of a system σ only represents part of the system. I.e. for

$$x_j \in V_i, j = 1, 2, \dots, m, \text{ it is possible that } \left(\bigcup_{j=1}^m C_A(x_j \in V_i) \right) \subset C_A(\sigma).$$

Example 2 of Section 2 provides an illustration for Corollary 21c, because in that example only the solid surface, but not the fluid phase, is decomposed on Level 1. Consequently, no representation of the fluid phase exists on Level 1 (cf. Figure 6).

Remark 21: We call two neighboring levels **homogeneous** if all the components in these levels are things of the same *type* (cf. Remark 8). Otherwise they are called being **heterogeneous**.

To illustrate these two kinds of neighboring levels, recall the explanations previously given for Remark 8 referring to the two examples of section 2. It has been made evident that the level of zones and the level of cells are *homogeneous* neighboring levels, and the level of molecular lattice sites and that of the continuum solid surface are *heterogeneous* neighboring levels.

Definition 21 clarifies what constitutes a level. In the following, we further define the environment and the structure of a level.

Definition 22: L-Environment. Let σ be a system with levels $V = \{V_i \mid 1 \leq i \leq n\}$. The environment of V_i , or an *L-environment*, is defined as the set of things which are not components in V_i but are coupled with one or more components in V_i :

$$E_L(V_i) := \{x \in \Theta \mid x \notin V_i \text{ \& } [\exists y \in V_i, b(x, y)]\}.$$

Definition 23: L-Structure. Let σ be a system with levels $V = \{V_i \mid 1 \leq i \leq n\}$. The *L-structure* of V_i is defined as the set of couplings among the components of V_i , and among them and the things in the *L-environment* of V_i :

$$S_L(V_i) := B_L(V_i) \cup \hat{B}_L(V_i),$$

where

$$B_L(V_i) = \{c(x, y) \mid x, y \in V_i\} \neq \Phi,$$

$$\hat{B}_L(V_i) = \{c(x, y) \mid x \in V_i \wedge y \in E_L(V_i)\}.$$

To illustrate the above two definitions, consider a two-level reactor system where a reactor (at Level 2) is composed of a vessel and a temperature control jacket (both at Level 1). The L-Environment for Level 1 may include the air surrounding the jacket outside the reactor (as a heat sink) to which the jacket loses heat. This heat sink should *not* be regarded as part of the L-Environment of Level 2, because it is directly coupled with the jacket at Level 1 of the system. The L-Structure of this system, according to Definition 23, is the set of couplings including

the (internal) heat transfer between the jacket and the vessel, as well as that between the jacket and the external heat sink.

In the context of multiscale modeling, it is of particular interest to investigate the relation between a component at a certain level and its *immediate composition*, as defined by

Definition 24: Immediate composition. Let σ be a system with levels $V = \{V_k \mid 1 \leq k \leq n\}$. The *immediate composition* of a component x on level V_i , $i > 1$, denoted by $C^{\text{im}}(x)$, is defined as a set of components on levels preceding V_i such that, each component in this set does not belong to the composition of any component located in a level below V_i . I.e.

$$C^{\text{im}}(x \mid x \in V_i, 1 < i \leq n) := \{y \in C(x) \wedge (\forall z \in V_k, k < i)[y \notin C(z)]\}$$

Remark 24: An element in the immediate composition of a component c is called an *immediate component* of c .

As an example, “solid surface” and “fluid phase” in Figure 6 are both immediate components of “reactor”, however the “sites” (at Level 1) are not.

A model of a system with multiple levels often takes into account the relations between a component of a system and its immediate components. In particular, many multiscale modeling applications are motivated by the possibility of better modeling properties of a component at a higher level of abstraction through relating them to some properties of its immediate components. In the following, we distinguish between two kinds of relations.

Definition 25: Aggregation law. Let σ be a system with levels $V = \{V_k \mid 1 \leq k \leq n\}$. Let x be a component on level V_i , $i > 1$ with an *immediate composition* $C^{\text{im}}(x)$. An *aggregation law* on x , denoted as $l^a(x)$, is a relation that maps one or more state functions of all the components in $C^{\text{im}}(x)$ to a state function of x . The totality of aggregation laws of x is defined as $L^a = \{l_i^a, 1 \leq i \leq n\}$.

Remark 25: We call an aggregation law an **averaging law**, if it represents the averaging (with respect to the dimension(s)⁶ along which the system is decomposed) of one state function of all components in the immediate composition of x (i.e. $C^{\text{im}}(x)$) to obtain a state function of x . An *averaging law* usually exists between two *homogeneous* neighboring levels (cf. Remark 21).

⁶ A dimension along which a system may be decomposed is represented as V_i in Definition 7.

Definition 26: Disaggregation law. Let σ be a system with levels $V = \{V_k \mid 1 \leq k \leq n\}$. Let x be a component on level V_i , $i > 1$ with an *immediate composition* $C^{\text{im}}(x)$. A *disaggregation law* on x , denoted as $l^d(x)$, is a relation that maps a state function of x to a state function of all the components in $C^{\text{im}}(x)$. The totality of disaggregation laws of x is defined as $L^d = \{l_i^d, \ 1 \leq i \leq n\}$.

To illustrate the above two definitions, the two examples of Section 2 are used again. As an aggregation law, a state function of a zone at Level 2 in Example 1 (cf. Figure 5) which represents a property such as the turbulent energy dissipation rate is determined by the same property of all CFD cells in this zone, as appearing at Level 1. In this particular case, the value of the property of the zone can be obtained by spatially averaging the values of this property of all the cells. As a disaggregation law, state functions that represent the temperature, chemical composition, etc. of a cell at Level 1 are determined by those of the entire zone at Level 2 which contains this cell. As one of the modeling alternatives, the values of these properties of each cell can simply be set equal to those of the entire zone.

In Example 2, within its three-level hierarchy, some state functions that represent the properties of the surface at Level 2, such as the desorption rate of a certain chemical species, can be determined by the micro-configuration of the lattice, i.e. by how every site that appears in Level 1 is occupied. The micro-configuration of the lattice is obtained through Monte Carlo simulation. This essentially implies an *aggregation law*. Further, the adsorption, desorption, or reaction transition probability of each site at Level 1, as an important state function of the site involved in the Monte Carlo simulation, is dependent of the temperature of the site. This temperature is set to be equal to the temperature of the entire surface which appears on Level 2. The relation between the temperature of a site and that of the surface gives another instance of *disaggregation law*.

In addition to aggregation laws and disaggregation laws which primarily govern *non-coupling induced state functions*, we still need to set up the connection between the *coupling induced state functions* of a component and some of its immediate components. For this purpose, we introduce

Definition 27: Mereological connection law. Let σ be a system with levels $V = \{V_k \mid 1 \leq k \leq n\}$. Let x be a component on level V_i , $i > 1$ with an *immediate composition* $C^{\text{im}}(x)$. A *mereological connection law* on x , denoted by $l^m(x)$, is a relation between a *coupling-induced state function* of x induced by a coupling with $y \in V_i$ or $z \in E_L(V_i)$, and one or more *coupling-induced state functions* of a set of components $U \subseteq C^{\text{im}}(x)$, each of

which is induced by a coupling between $u \in U$ and $v \in C^{im}(y)$ or $w \in C(z)$. The totality of mereological connection laws of x is defined as $L^m := \{l_i^m, 1 \leq i \leq n\}$.

To illustrate Definition 27, let us first examine the coupling between two zones in Figure 2. Figure 7 represents a (mass transfer) coupling between Zone A and Zone B. Now consider one of the two zones, e.g. Zone A. This zone has two coupling-induced state functions F_{out} and F_{in} , representing the rate of the mass flow leaving this zone and entering Zone B, and *vice versa*. In Figure 8, it is shown that at Level 1 (i.e. the level of cells), each cell at the boundary of Zone A is coupled to one cell at the corresponding boundary of Zone B. Such a coupling introduces two coupling-induced state functions of this cell, f_{out} and f_{in} . As an instance of the mereological connection law, there exists a (equality) relation between F_{out} (F_{in}) and the sum of f_{out} (f_{in}) of all cells at the boundary of Zone A which is connected to Zone B.

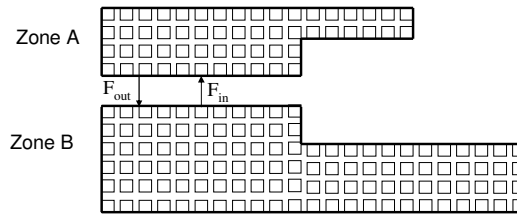


Figure 7. Coupling between two zones.

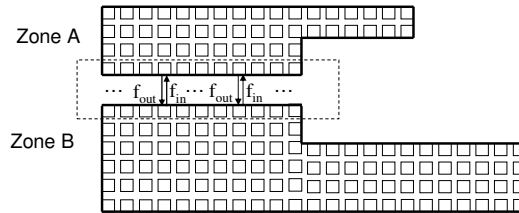


Figure 8. Couplings between cells at the boundaries.

In some simpler cases, an input or output of a component at a higher level is directly assigned to one of its immediate components. In such a case, a mereological connection law simply represents this assignment. Figure 8 illustrates an example in chemical plant simulation. The variables of a material stream (all being coupling-induced state functions), which flows from Plant A to Plant B as part of the simulated flowsheet when viewed at the *plant* level (as opposed to the *unit* level), eventually need to be assigned to those of a material stream entering a certain

process unit within the destination plant (i.e. Plant B), in order to set up the coupling between these two plants at the unit level.

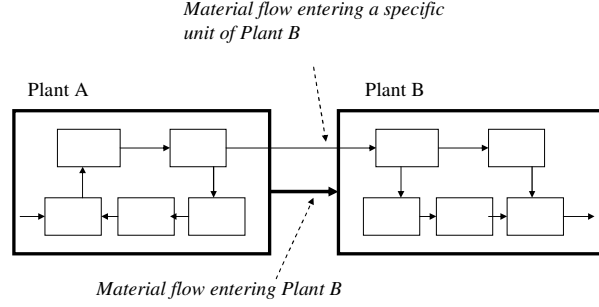


Figure 9. An example of simple mereological connection laws.

3.3 Multiscale models of systems (Part 3)

Notions introduced in Subsections 3.1 – 3.2 have paved the way for precisely characterizing a multiscale model of a system. In this subsection, we first define the concept of a single-scale system model. Thereafter, the concept of a multiscale model and its classification are introduced. Note that only definitions of these models are given here; one should refer to Section 4 for examples of these models.

A single-scale system model is a collection of the laws of all components at a certain level of observation and the laws on couplings among these components. Rigorously, we introduce

Definition 28: Single-scale model. Let σ be a system and V a level of σ . For any component x on level V , let $L(x)$ be the totality of laws on x . Let $S_L(V)$ be the L-structure of level V . Let $L^t(r)$ and $L^c(r)$ be the totality of the topological connection laws and that of coupling mechanism laws of a coupling $r \in S_L(V)$, respectively. The *single-scale model* of σ for level V , denoted as $M^s(\sigma, V)$, is defined as follows:

$$M^s(\sigma, V) := \{L(x) \mid x \in V\} \cup \{L^t(r) \mid r \in S_L(V)\} \cup \{L^c(r) \mid r \in S_L(V)\}.$$

A multiscale model is a collection of single-scale models as well as laws characterizing inter-scale relations of the system being modeled. Rigorously, we introduce

Definition 29: Multiscale model. Let σ be a system with levels $V = \{V_k \mid 1 \leq k \leq n, n > 1\}$. For any component z on level V_i , $1 < i \leq n$, let $L^a(z)$, $L^d(z)$, and $L^m(z)$ be the totality of its aggregation, disaggregation, and mereological connection laws, respectively. Thus, the *multiscale model* of σ , denoted as $M^m(\sigma)$, is defined as follows:

$$M^m(\sigma) := \{M^s(\sigma, V_k), V_k \in V\} \cup \bigcup_{k=2}^n \{L^a(z) \mid z \in V_k\} \cup \bigcup_{k=2}^n \{L^d(z) \mid z \in V_k\} \\ \cup \bigcup_{k=2}^n \{L^m(z) \mid z \in V_k\}$$

Reflecting how closely the different levels of a system are linked, multiscale models can be further classified into three types⁷. In one extreme case, a *scale-collecting model* simply collects a number of single-scale models without involving any laws on the inter-scale connections. In another extreme case, a *scale-integrating model* involves at least aggregation or disaggregation laws. In between, a *scale-connecting model* involves some mereological connection law(s) but does not have any aggregation or disaggregation laws. Rigorously, we introduce

Definition 30: Scale-collecting model. Let σ be a system with levels $V = \{V_k \mid 1 \leq k \leq n, n > 1\}$. Let $M^m(\sigma)$ be a multiscale model of σ . For any component z on level V_i , $1 < i \leq n$, let $L^a(z)$, $L^d(z)$, and $L^m(z)$ be the totality of its aggregation, disaggregation, and mereological connection laws, respectively. $M^m(\sigma)$ is a *scale-collecting model*, iff

$$\bigcup_{k=2}^n \{L^a(z) \mid z \in V_k\} \cup \bigcup_{k=2}^n \{L^d(z) \mid z \in V_k\} \cup \bigcup_{k=2}^n \{L^m(z) \mid z \in V_k\} = \emptyset.$$

This definition simply states that a scale-collecting model is one where there is neither an aggregation law, a disaggregation law, nor a mereological connection law on any level that has the possibility of possessing such a law (i.e. any level other than Level 1).

Definition 31: Scale-connecting model. Let σ be a system with levels $V = \{V_k \mid 1 \leq k \leq n, n > 1\}$. Let $M^m(\sigma)$ be a multiscale model of σ . For any component z on level V_i , $1 < i \leq n$, let $L^a(z)$, $L^d(z)$, and $L^m(z)$ be the totality of its aggregation, disaggregation, and mereological connection laws, respectively. $M^m(\sigma)$ is a *scale-connecting model*, iff

$$\bigcup_{k=2}^n \{L^m(z) \mid z \in V_k\} \neq \emptyset \quad \text{and} \quad \bigcup_{k=2}^n \{L^a(z) \mid z \in V_k\} \cup \bigcup_{k=2}^n \{L^d(z) \mid z \in V_k\} = \emptyset.$$

This definition states that a scale-connecting model is one where there are one or more mereological connection laws but no aggregation law or disaggregation law.

⁷ This classification is essentially based on the three types of inter-scale laws defined earlier, namely the *mereological connection law*, the *aggregation law*, and the *disaggregation law*. This classification would be complete, if there were no inter-scale laws other than these three. However, we consider the validity of this assumption as an open issue. See the concluding remarks in Section 6 for further discussions of this matter.

Definition 32: Scale-integrating model. Let σ be a system with levels $V = \{V_k \mid 1 \leq k \leq n, n > 1\}$. Let $M^m(\sigma)$ be a multiscale model of σ . For any component z on level V_i , $1 < i \leq n$, let $L^a(z), L^d(z)$ be the totality of its aggregation and disaggregation laws, respectively. $M^m(\sigma)$ is a *scale-integrating model*, iff

$$\bigcup_{k=2}^n \{L^a(z) \mid z \in V_k\} \cup \bigcup_{k=2}^n \{L^d(z) \mid z \in V_k\} \neq \emptyset.$$

This definition states that a scale-integrating model is one where there is at least one aggregation law or one disaggregation law.

In the practice of systems modeling, it is rare to develop scale-collecting models; this concept is introduced here mainly for the purpose of completeness. One example of scale-connecting models is the flowsheet-partition-unit schema frequently employed in chemical process simulation, where different scales are introduced and connected for carrying out a particular simulation algorithm (cf. the second example of mereological connection law, illustrated in Figure 9). The most interesting and complex multiscale models are scale-integrating models: it is the existence of the aggregation/disaggregation laws that brings the most prominent value of considering different scales.

4. Exemplary applications of the conceptualization framework

In this section, a detailed analysis of the two illustrative examples introduced in Section 2 is presented to give a more complete explanation on the composition of scale-integrating models. All the concepts defined in the preceding section will be highlighted in *italic* subsequently. When necessary, back references to the numbers of the definitions or remarks are also given. As will be shown through these two examples, adjustments to the models formulated by their original authors may be required to properly re-express them using the current conceptualization framework; these adjustments will be briefly discussed at the end of this section.

4.1 Example 1 - Hybrid multizonal/CFD model of chemical equipment

As shown earlier, the model originally formulated by Bezzo, et al. (2004) represents a piece of process equipment as three levels (cf. Figures 1 and 5). In the following, we will take a closer look at Level 2 and Level 1 in order to analyze the composition of this hybrid model according to the conceptualization framework presented in this work. Unless stated otherwise, all the modeling treatments are based on Bezzo et al. (2004), where further explanations regarding the model may be found.

4.1.1 The multizonal model of Level 2

On this level, the *system* of equipment is decomposed into a number of (internal) zones; one zone can be connected to one or more other zones. A *coupling* between two zones, standing for material transport, is represented through an interface that links two ports⁸, each of which belongs to one of the two connected zones. In addition, the connection between the equipment and the *environment* of the equipment is represented through some so-called environment zones, which essentially play the role of *couplings* between the (internal) zones and the things in its environment. In the following, the set of ports of an arbitrary zone z is denoted by P_z . The sets of internal and external zones on this level are denoted by Z^I and Z^E , respectively.

4.1.1.1 Laws of individual (internal) zones

Laws applicable to one (internal) zone typically include the conservations of mass and energy, chemical reaction kinetics, and other types of kinetics, depending on the phenomena to be captured in the zone. The collection of these *laws* can be represented by a differential-algebraic model of the following form:

$$F_z^I(y_z, dy_z / dt, \{x_p, x_p^{in}, F_p, F_p^{in}, \forall p \in P_z\}, \alpha_z, \theta_z, V_z) = 0, \forall z \in Z^I \quad (1)$$

where y_z is a set of internal zone variables (e.g. mass and energy holdups, compositions, temperature, etc.), x_p denotes the vector of intensive properties (e.g. temperature, pressure) that any material exiting zone z at port p would have, x_p^{in} denotes the vector of intensive properties of the material entering zone z at port p , F_p is the outlet mass flow at port p , F_p^{in} is the inlet mass flow at port p , α_z denotes a set of parameters that are related to fluid flow and/or mixing phenomena, and which somehow affect the behaviour of zone z (e.g. the turbulent energy dissipation rate which may affect the rate of nucleation in a crystallisation process), θ_z is a set of fluid properties (e.g. density, viscosity, etc.) and/or parameters that affect such fluid properties, and V_z is the volume of zone z which is assumed to be a given constant.

Applying our conceptualization framework, all the above variables, properties, and parameters are *state functions* (Definition 7) of this zone. Especially, variables appearing within the bracket in Eq. (1) are essentially *coupling-induced state functions* (Definition 11).

4.1.1.2 Laws of couplings on Level 2

In the original model, an environment zone model is introduced to represent certain relations between the environment and the internal zones, which is of the following form:

$$F_z^E(y_z, u, \{x_p, x_p^{in}, F_p, F_p^{in}, \forall p \in P_z\}) = 0, \forall z \in Z^E. \quad (2)$$

⁸ Applying the concepts defined in this paper, one can consider a *port* as a collection of *state functions* of a *thing* induced by a particular *coupling* between this *thing* and another *thing*.

Here the variables y_z and u represent the connections between the environment and this environment zone, and the variables in the bracket represent connections between this environment zone and some internal zones. As mentioned above, such an environment zone can be replaced by one or more *couplings* between the environment and one or more internal zones. A *topological connection law* (Definition 13) of the following form can be applied to each of such couplings:

$$l^t(v_E^c, \{x_z, x_z^{in}, F_z, F_z^{in}\}) = 0, \quad (3)$$

where v_E^c denotes all *state functions* of the environment *induced by this coupling*, and the variables in the bracket are *state functions* of this (internal) zone *induced by this coupling*. The collection of such laws of all couplings with the environment is equivalent to (2).

In addition to the couplings with the environment, there are still couplings between internal zones. For such a coupling, *topological connection laws* in the form of identity equations are applied,

$$\left. \begin{array}{l} F_p^{in} = F_{p'} \\ F_{p'}^{in} = F_p \\ x_p^{in} = x_{p'} \\ x_{p'}^{in} = x_p \end{array} \right\} \quad \forall (p, p') \in I, \quad (4)$$

where (p, p') represents one interface between ports p and p' , each belonging to one zone involved in the coupling.

4.1.2 The (discretised) CFD model of Level 1

This level is composed of a number of cells (denoted by C) and their connections resulting from the discretization of the CFD model of the equipment.

4.1.2.1 Laws of individual cells

One CFD cell is characterized by *state functions* such as volume (V_c), pressure (P_c), intensive properties (θ_c) which are determined by the Level 2 (i.e. multizonal) model (e.g. temperature and composition), fluid velocity at its centre (v_c), hydrodynamic quantities α_c (e.g. the turbulent energy dissipation rate), and finally the physical properties such as density (ρ_c) and viscosity (μ_c). The CFD model of one cell is typically composed of laws on mass/momentum conservations, as well as physical property relations for density and viscosity of the following form:

$$\rho_c = \rho(P_c, \theta_c), \quad \mu_c = \mu(P_c, \theta_c), \quad \forall c \in C. \quad (5)$$

The solution of all the laws of one cell provides values of quantities such as v_c and α_c , which are needed for setting up connections between neighboring cells (cf. Section 4.1.2.2) and between Level 1 and Level 2 (cf. Section 4.1.3.1).

4.1.2.2 Laws of couplings on Level 1

In the original model, two adjacent cells (c, c') share a face f ; the set of faces in the CFD model is denoted as \mathcal{F} . Faces at the boundaries of the domain of interest are also denoted as (c, c'). This treatment implies that a real cell c is connected to c' which can be either another real cell or a topological boundary cell. Here, a face is essentially a *coupling* between two cells, through which momentum and mass transport occurs. For such a coupling, *coupling-induced state functions* of the involved cells are the fluid velocity and the corresponding mass flow-rates at the faces; the latter can be derived from the former through a *law* of this cell. Comparable to (4), identity relations (as *topological connection laws*) should exist between the above coupling-induced state functions of two coupled cells.

Besides, the solution of the CFD model gives the fluid velocity at the centre of each cell. To obtain the fluid velocity at a face between two cells, certain interpolation can be applied. According to Definition 12, this interpolation relation can be viewed as a *coupling mechanism law*, as it relates a *coupling-induced state function* (the fluid velocity at the face) to a *non-coupling-induced state function* (the fluid velocity at the center of a cell)⁹.

4.1.3 Inter-scale laws

Now we come to the part of the original model that addresses the connection between the above two (essentially *homogeneous* neighboring) levels (Remark 21). The notation C_z is introduced to denote the set of cells on Level 1 comprising the zone z on Level 2. We will show in the following that all the three types of inter-scale laws are present in the original model.

4.1.3.1 Aggregation laws

One rather obvious *aggregation law* (Definition 25) relates the volume of one zone and to those of the cells in the zone:

$$V_z = \sum_{c \in C_z} V_c \quad (6)$$

Another *aggregation law*, which is more significant, derives a volume-averaged property of one zone, such as its energy dissipation rate, from the properties of the cells in the zone:

⁹ The law statement in this case does not really reveal the mechanism of this coupling; it is rather merely a manipulation of the result of solving the CFD model.

$$\alpha_z = \frac{\sum_{c \in C_z} \alpha_c V_c}{V_z} \quad (7)$$

According to Remark 25, Eq. (7) is an *averaging law*.

4.1.3.2 Disaggregation laws

Disaggregation laws (Definition 26) are introduced in this model to relate the properties determined by the Level 2 (the multizonal) model (i.e. θ_z in Eq. (1)) to the properties needed by the CFD model of a cell in this zone (i.e. θ_c in Eq. (5)). The original model presents a simplest (identity) relation:

$$\theta_c = \theta_z, \forall c \in C_z \quad (8)$$

4.1.3.3 Mereological connection laws

Here, *mereological connection laws* (Definition 27) are introduced to relate *coupling-induced state functions* of cells on Level 1 with those of the zones on Level 2. In particular, mass flow-rates across the faces of cells which make up an interface (p, p') between ports p and p' of two adjacent zones (this set of faces being denoted as $\mathcal{F}_{pp'}$) are related to the mass flow-rates across the interface (or *coupling*) between these two zones. These *mereological connection laws* are

$$F_{pp'} = \sum_{(c, c')} \max(F_{cc'}, 0), \quad F_{p'p} = \sum_{(c, c')} \max(-F_{cc'}, 0), \quad (9)$$

where $(c, c') \in \mathcal{F}_{pp'}$, $F_{cc'}$ is the rate of the mass flow from cell c to its adjacent cell c' , $F_{pp'} \equiv F_p^{in} = F_{p'}$, $F_{p'p} \equiv F_p = F_{p'}^{in}$, (cf. Eq. (4)). By Eq. (9), both $F_{pp'}$ and $F_{p'p}$ are non-negative. In particular $F_{pp'}$ sums up the rates of the non-negative flows from cells (denoted by c) belonging to port p to those (denoted by c') belonging to port p' , while $F_{p'p}$ sums up the rates of the non-negative flows from cells belonging to port p' to those belonging to port p .

4.2 Example 2 - Homogeneous-heterogeneous reactor model

As shown earlier in Figure 6, the example originally formulated by Vlachos (1997) involves three levels of the reactor being modeled. As in the previous example, we will take a closer look at Level 2 and Level 1, and analyze the composition of this reactor model according to the conceptualization framework presented in this work. Again, unless stated otherwise, all the modeling treatments are based on Vlachos (1997), where further explanations regarding the model may be found.

4.2.1 The continuum model of Level 2

On this level, the *system* reactor is decomposed into two *subsystems*, namely the fluid phase and the solid surface. A *coupling* exists between the two parts to represent the mass transfer between them.

4.2.1.1 Laws of the fluid phase

The model of the fluid phase is composed by applying *laws* of mass conservation and mass diffusion (Fick's first law in particular). Concretely, it includes the following equations (assuming no reaction occurs in the fluid phase):

$$\frac{\partial C_A}{\partial t} = \frac{\partial}{\partial x} \left(D \frac{\partial C_A}{\partial x} \right), \quad (10 \text{ a})$$

$$C_A = C_{AL} \quad \text{at} \quad x = L, \quad (10 \text{ b})$$

$$D \frac{\partial C_A}{\partial x} = r_A^{out} - r_A^{in} \quad \text{at} \quad x = 0. \quad (10 \text{ c})$$

Here C_A is the concentration of A , x is the spatial coordinate, L is the boundary of the fluid phase on the opposite of its interface with solid surface, C_{AL} is the concentration of A at L , D is the diffusivity, r_A^{out} is the rate of transporting A out of the fluid phase due to the adsorption at the solid surface, r_A^{in} is the rate of transporting A into the fluid phase due to the desorption at the solid surface. Clearly, r_A^{in} and r_A^{out} are two *coupling-induced state functions* (Definition 11) of the fluid phase (see also Section 4.2.1.3).

4.2.1.2 Laws of the solid surface

The model of the solid surface is composed by applying *laws* of mass conservation of the surface species AS as well as kinetics laws of adsorption, desorption, and surface reaction. Concretely, it includes

$$\frac{\partial C_{AS}}{\partial t} = r_a - r_d - r_{rs}, \quad (11 \text{ a})$$

$$r_a = k_a C_{A0} (1 - \theta_A), \quad (11 \text{ b})$$

$$r_d = k_d f(\theta_A, T, w_{AA}), \quad (11 \text{ c})$$

$$r_{rs} = k_r C_{AS}. \quad (11 \text{ d})$$

Here C_{AS} is the concentration of AS , C_{A0} is the concentration of A in the fluid phase at the boundary adjacent to the surface (which can be viewed as a *coupling-induced state function* of the surface), r_a , r_d , r_{rs} are rates of adsorption, desorption, and surface reaction respectively, k_a , k_d , k_r are temperature-dependent rate constants of adsorption,

desorption, and surface reaction respectively, w_{AA} is the energy of inter-adsorbate interactions, and θ_A is the coverage defined as

$$\theta_A = C_{AS} / C_T \quad (11 \text{ e})$$

where C_T is the total concentration of sites on the surface.

To determine the function f in Eq. (11 c), one may employ a mean-field model, which assumes the atoms are uniformly distributed on the surface and share the same number of neighbors. This treatment does not require further modeling at the non-continuum level, but it is not able to count for imperfections such as defects and impurities on the surface. It is well known that adequately addressing these factors can be very important for providing a reliable description of the surface. To tackle this problem, this function can take the following form, which employs the information available from the modeling at the non-continuum level:

$$f = f(j, w_{AA}, T, q_j, j = 0, 1, \dots, c). \quad (12)$$

Here c is the coordination of the surface (e.g. $c = 4$ for a square lattice representing the (100) surface), q_j is the fraction (over all sites) of the occupied sites with local coordination equal to j . As we will show later in Section 4.2.3.1, the value of q_j can be determined by aggregating certain *state functions* of the sites in the lattice modeled on Level 1.

4.2.1.3 Laws of the coupling

The coupling between the fluid phase and the solid surface is characterized by the following *topological connection laws* (Definition 13):

$$C_{A|x=0} = C_{A0}, \quad (13 \text{ a})$$

$$r_A^{in} = r_d, \quad (13 \text{ b})$$

$$r_A^{out} = r_a. \quad (13 \text{ c})$$

Eq. (13 a) equates the concentration of A at the boundary in the fluid phase ($C_{A|x=0}$) with the concentration of A referred to by the surface model (C_{A0}) (cf. Eq. 11b). Eq. (13 b) equates (i) the rate of transporting A into the fluid phase due to desorption (r_A^{in}) to (ii) the desorption rate referred to by the surface model (r_d) (cf. Eq. 11c). Eq. (13 c) sets up the same relation for rates pertaining to adsorption.

It is worth noting that, according to Definition 13, a *topological connection law* relates only *coupling induced state function* of two coupled things. Now consider the variables involved in Eqs. (13 a-c) which have just been termed a set of *topological connection laws*. One can see that within the set of individual variables, only r_A^{in} , r_A^{out} (cf. Section 4.2.1.1) and C_{A0} (cf. Section 4.2.1.2) have been previously identified as *coupling-induced state functions*. All the other variables are not coupling-induced because they exist regardless of whether there is a coupling between the fluid phase and the solid surface or not. This “problem” can be resolved if the following, uniformed way of formulating models is applied:

- One uses *non-coupling induced state functions* of a thing only within *laws* of the thing;
- One introduces, when needed, additional *coupling-induced state functions* which have identity relations with some of the *non-coupling-induced state functions*; and
- One uses only *coupling-induced state functions* in *topological connection laws*.

In fact, a treatment of this kind is frequently applied in chemical process modeling, where the composition of a material stream leaving a chemical process unit (as a *coupling-induced state function*) is set to be identical to the composition of the material in the unit (as a *non-coupling-induced state function*). This is exactly what the second point is about. These *non-coupling-induced state functions* are usually used for modeling the phenomena within the material body, corresponding to the first point. These *coupling-induced state functions* are usually used for setting up the connections with other units, corresponding to the third point.

Regarding the current reactor modeling example, one can introduce (corresponding to the second point) additional *coupling-induced state functions* for $C_{Alx=0}$ of the fluid phase and r_d/r_a of the surface, by

$$C'_{Alx=0} = C_{Alx=0}, \quad (13 \text{ a1})$$

$$r'_d = r_d, \quad (13 \text{ b1})$$

$$r'_a = r_a. \quad (13 \text{ c1})$$

Here $C'_{Alx=0}$, r'_d , and r'_a are simply a “coupling-induced” copy of their non coupling-induced counterpart; they should not exist if no coupling is to be modeled. This is comparable to the introduction of material stream variables which would not be needed if a chemical process unit does not have any connections to other things. After this treatment, Eqs (13 a-c) may be rewritten, following the third point, as

$$C'_{Alx=0} = C_{A0}, \quad (13 \text{ a2})$$

$$r_A^{in} = r_d', \quad (13 \text{ b2})$$

$$r_A^{out} = r_a'. \quad (13 \text{ c2})$$

This set of *topological connection laws* now contains *coupling-induced state functions* only.

4.2.2 The non-continuum model of Level 1

On this level, the solid surface is depicted by a (large) number of sites in a lattice. A site i in the lattice can be characterized by state functions including the temperature (T_i), the concentration of A at the fluid phase boundary adjacent to the site ($C_{A0,i}$), the occupation-site function (δ_i), the local coordination (lc_i), and finally the transition probabilities (Γ_{ai} , Γ_{di} , and Γ_{ri}) which are respectively of three micro-processes, namely adsorption, desorption, and reaction. A transition probability denotes the probability for a particular type of events (i.e. adsorption, desorption, or reaction) to occur at a site in the lattice, at a particular point in time. The *laws* of a site relate the transition probabilities to other state functions, which take the following forms:

$$\Gamma_{ai} = \Gamma_a(T_i, C_{A0,i}, \delta_i), \quad (14 \text{ a})$$

$$\Gamma_{di} = \Gamma_d(T_i, C_{A0,i}, \delta_i, lc_i), \quad (14 \text{ b})$$

$$\Gamma_{ri} = \Gamma_r(T_i, \delta_i). \quad (14 \text{ c})$$

Besides, the local coordination (lc_i) of a site is related to the occupation-site function of its neighboring sites δ_k :

$$lc_i = \sum_{k=1}^c \delta_k. \quad (14 \text{ d})$$

Here c is the number of its neighboring sites. $\{\delta_k\}$ can be considered as *coupling-induced state functions* of this site. Each of these *coupling-induced state functions* is, unsurprisingly, equal to the occupation-site function of its corresponding neighbor, which is a (*non-coupling-induced*) *state function* of the neighbor. This equality relation can thus be viewed as a (simple) *coupling mechanism law* (Definition 12) of the *coupling* between this site and the neighbor.

4.2.3 Inter-scale laws

Now we turn to the part of the original model that addresses the connection between the above two (*heterogeneous* neighboring) levels. We will show in the following that *aggregation* and *disaggregation laws* (Definitions 25 and 26) are present in the original model.

4.2.3.1 Aggregation laws

As mentioned earlier, the aim of the non-continuum modeling of Level 1 is to provide information needed for determining the state function q_j of the surface at Level 2, which is the fraction (of all sites) of the occupied sites with local coordination equal to j . This has been done through Monte-Carlo (MC) simulation. The MC simulation essentially solves the Master Equation (Reese, et al, 2001)

$$\frac{dP_\alpha}{dt} = \sum_\beta [W_{\alpha\beta}P_\beta - W_{\beta\alpha}P_\alpha], \quad (15)$$

where P_α is the probability of the surface being in configuration α , $W_{\alpha\beta}$ is the transition probability per unit time of the surface changing from configuration β to α . Since Eq. (15) relates state functions of the entire surface instead of individual sites, it should be regarded as a (stochastic) *law* of the surface on Level 2. However, this law is not actually included in the reactor model, because it cannot be solved directly given the large number of possible configurations. Instead, a MC simulation is employed, the execution of which essentially provides the following relation between the *state function* q_j of the surface on Level 2 and a set of *state functions* of all sites on Level 1,

$$q_j = f^{MC}(\delta_k, \Gamma_{ak}, \Gamma_{dk}, \Gamma_{rk}, \quad k = 1, 2, \dots, NC), \quad j = 0, 1, \dots, c, \quad (16)$$

where $\delta_k, \Gamma_{ak}, \Gamma_{dk}, \Gamma_{rk}$ are the same as in Eq. (14), NC is the number of the sites in the lattice, c is the coordination of the lattice same as used in Eq. (12). According to Definition 25, Eq. (16) can be considered as a set of *aggregation laws*.

4.2.3.2 Disaggregation laws

From Eqs. (14 a-c) we can see that characterizing one site requires information such as temperature and the concentration of A at the fluid phase boundary adjacent to the site. These state functions are determined by the links to the corresponding *state functions* of the entire surface. In this particular model, the *state functions* of one site are considered equal to those of the surface. This implies that a set of *disaggregation laws* are employed in form of identity equations.

4.3 Discussion

The above two subsections have shown that the conceptualization framework for multiscale modeling is applicable to two different examples. However, some adjustment to the original model was required in each case. For example, in the first model (cf. Section 4.1.1.2), the concept of “environment zone” introduced by the original author was dropped, and the part of model addressed by this concept was re-expressed in the form of *topological*

connection laws that govern the *couplings* between the environment and internal zones of the equipment. In the second model (cf. Section 4.2.1.3), several *coupling-induced state functions* added to the original set of variables, in order to retain the standard form of topological connection laws as defined in the conceptualization framework. Generally speaking, the model elements being adjusted in the two examples were not inappropriate in their original form as far as these specific modeling applications are concerned. However, these original formulations appear to have been adopted in a rather case-specific or ad-hoc manner. After the adjustments, it was demonstrated that the same elements could be formulated merely using the concepts generically defined in the conceptualization framework.

5. Interpreting and comparing with some other generalization work

The conceptualization framework developed in this work has been intended to help to clarify fundamental concepts such as scale and to present generic compositions of various kinds of multiscale models. In this section, we try to relate our conceptualization framework to the some generalization-oriented work by other researchers.

5.1 The classification of “scale integration” strategies by Pantelides (2001)

In Pantelides (2001), a distinction is made between “scale decoupling” which deals with different scales of a system separately without linking them together, and “scale integration” which refers to strategies of modeling where different scales are connected. According to the way these scales are connected, “scale integration” is further classified into the following:

Serial integration. According to Pantelides (2001), this is a strategy in which a finer scale model is employed to provide parameters or data needed by a coarser scale model. To generalize, one can say that serial integration connects models at different scales which can be solved sequentially. Applying our conceptualization framework, serial integration results in (i) *scale-connecting models* (Definition 31) or (ii) *scale-integrating models* (Definition 32) with either *aggregation law(s)* (Definition 25) or *disaggregation law(s)* (Definition 26) (but not both). More rigorously, we introduce

Definition D1: Serial-integration model. Let σ be a *system* with *levels* $V = \{V_k \mid 1 \leq k \leq n, n > 1\}$. Let $M^m(\sigma, t)$ be a *multiscale model* of σ . For any component z on level V_i , $1 < i \leq n$, let $L^a(z)$, $L^d(z)$, and $L^m(z)$ be the totality of its *aggregation*, *disaggregation*, and *mereological connection laws*, respectively. $M^m(\sigma)$ is a *serial-integration model*, iff

$$(i) \quad \bigcup_{k=2}^n \{L^a(z) \mid z \in V_k\} \cup \bigcup_{k=2}^n \{L^d(z) \mid z \in V_k\} = \emptyset \quad \text{and} \quad \bigcup_{k=2}^n \{L^m(z) \mid z \in V_k\} \neq \emptyset, \text{ or}$$

$$(ii) \quad \bigcup_{k=2}^n \{L^a(z) \mid z \in V_k\} \cup \bigcup_{k=2}^n \{L^d(z) \mid z \in V_k\} \neq \emptyset \quad \text{and} \\ \left(\bigcup_{k=2}^n \{L^a(z) \mid z \in V_k\} = \emptyset \quad \text{or} \quad \bigcup_{k=2}^n \{L^d(z) \mid z \in V_k\} = \emptyset \right).$$

The first condition points to the case of a *scale-connecting model*. The second condition refers to a *scale-integrating model* (by the first part of this condition) which however has either only *aggregation laws* or only *disaggregation laws* (indicated by the second part of this condition).

Parallel integration. According to Pantelides (2001), this is a strategy which “involves the simultaneous use of descriptions at different scales applied to the same computational domain. The results of one description form inputs to the other, and vice versa”. This is exactly the case of a *scale-integrating model* which has both *aggregation law(s)* and *disaggregation law(s)*. More rigorously, we can derive

Definition D2: Parallel-integration model. Let σ be a *system* with *levels* $V = \{V_k \mid 1 \leq k \leq n, n > 1\}$. Let $M^m(\sigma)$ be a *multiscale model* of σ . For any *component* z on *level* V_i , $1 \leq i \leq n$, let $L^a(z)$ and $L^d(z)$ be the totality of its *aggregation* and *disaggregation laws*, respectively. $M^m(\sigma)$ is a *parallel-integration model*, iff

$$\bigcup_{k=2}^n \{L^a(z) \mid z \in V_k\} \neq \emptyset \quad \text{and} \quad \bigcup_{k=2}^n \{L^d(z) \mid z \in V_k\} \neq \emptyset.$$

Hierarchical integration. According to Pantelides (2001), this is a strategy in which “a finer scale model is formally embedded within the higher-scale model to represent a set of relations among macroscopic quantities occurring in the latter”. In “serial integration” and “parallel integration” introduced earlier, an implicit assumption has been that the models of different scales are solved or evaluated individually before serial or iterative connections between them are made. In comparison, hierarchical integration stresses the direct embedding of a finer scale model into a coarser scale model; the entire system model is then solved as a whole. Thus, this integration strategy is mainly concerned with the solution (as opposed to the composition) of a multiscale model. With regard to the composition of the model particularly in terms of the relations between different modeling scales, the model solved using this strategy can be either a *serial-integration model* or a *parallel-integration model*. Thus, no additional concept is derived for this strategy.

Simultaneous integration. According to Pantelides (2001), this is a strategy in which “the higher-scale model is formed completely from finer scale descriptions”. Applying concepts previously defined in this paper, the result of simultaneous integration is essentially a *multiscale model* in which only the lowest *level* has *laws*; *state functions* of all other *levels* are determined only through inter-scale *laws*. More rigorously, we introduce

Definition D3: Simultaneous-integration model. Let σ be a system with levels $V = \{V_k \mid 1 \leq k \leq n, n > 1\}$.

For any component x on level V_k , let $L(x)$ be the totality of laws on x . Let $S_L(V_k, t)$ be the L -structure of level V_k . Let $L^t(r)$ and $L^c(r)$ be the totality of the topological connection laws and that of coupling mechanism laws of a coupling $r \in S_L(V_k)$, respectively. For any component z on level V_i , $1 < i \leq n$, let $L^a(z)$, $L^d(z)$, and $L^m(z)$ be the totality of its aggregation, disaggregation, and mereological connection laws, respectively. Let $M^m(\sigma)$ be a multiscale model of σ . $M^m(\sigma)$ is a simultaneous-integration model, iff

$$\begin{aligned} & \{L(x) \mid x \in V_1\} \cup \{L^t(r) \mid r \in S_L(V_1)\} \cup \{L^c(r) \mid r \in S_L(V_1)\} \neq \emptyset \quad \text{and} \\ & \bigcup_{k=2}^n \{L(x) \mid x \in V_k\} \cup \bigcup_{k=2}^n \{L^t(r) \mid r \in S_L(V_k, t)\} \cup \bigcup_{k=2}^n \{L^c(r) \mid r \in S_L(V_k, t)\} = \emptyset \\ & \quad \text{and} \\ & \bigcup_{k=2}^n \{L^a(z) \mid z \in V_k\} \cup \bigcup_{k=2}^n \{L^d(z) \mid z \in V_k\} \cup \bigcup_{k=2}^n \{L^m(z) \mid z \in V_k\} \neq \emptyset. \end{aligned}$$

The first condition states that intra-level laws exist at the lowest level of the system. The second condition states that the levels other than the lowest one of the system are not modeled with any intra-level laws. Finally, the third condition states that there must be some inter-level laws defined at the levels where intra-level laws do not exist.

5.2 The classification of integration frameworks by Ingram et al. (2004)

Ingram et al. (2004) further expand the classification of Pantelides (2001), mainly based on an analysis on the relations between the “regions of the system domain” addressed by models of different scales. In general, a system domain can be viewed as a dimension along which the system under investigation is decomposed, which can typically be temporal or spatial. In the work of Ingram et al. (2004), multiscale models of process systems are particularly classified according to the spatial relationship between the balance volumes that comprise the model.

Their classification includes serial, parallel, and simultaneous integrations. Serial integration is further classified into simplification, transformation, and one-way coupling. The former two treatments eventually result in a single-scale model by making a micro-scale model part of a macro-scale model after either reduction or transformation. The case of one-way coupling, where there is only one-way information flow from the micro-scale to the macro-scale, essentially refers to *serial integration model* discussed earlier (Definition D1). Simultaneous integration is equivalent to what has been defined in Definition D3. Parallel integration has a different meaning

from the term introduced by Pantelidies (2001): it now refers to the case where multiple models are applied to the same system domain; each of them can be a multiscale model of any kind. Certain links exist between the models in “parallel”. Since this parallel integration framework primarily connects models which are not of different scales of one system, it is not analyzed here.

There are other two types of integration frameworks introduced by Ingram et al. (2004):

Multi-domain integration. This is a framework where “the micro-scale and macro-scale parts of a model occupy adjacent, non-overlapping parts of the system domain”. That means, more generally, different parts of a system are modeled at different scales. Viewing it in terms of *levels* of a *system* as formulated in this current paper, it can be seen as the case where the model is composed of (sub)models of different *levels*; each *level* is only partially modeled. To explain this, the multi-domain model of the catalytic packed-bed reactor given by Ingram et al. (2004) is analyzed as follows. The entire model has been interpreted by Ingram et al. as a combination of a macro-scale model of the liquid phase and micro-scale models of catalytic pellets (cf. Figure 10a). However, it can also be viewed as a result of first decomposing the reactor into the fluid phase and the catalyst phase and then decomposing the catalyst phase into pellets (cf. Figure 10b). Various *laws* are applied to the fluid phase, to the *coupling* between the fluid phase and the catalyst phase, and to the pellets. Additionally, the *inter-scale relations* are introduced to connect the pellets with the catalyst phase.

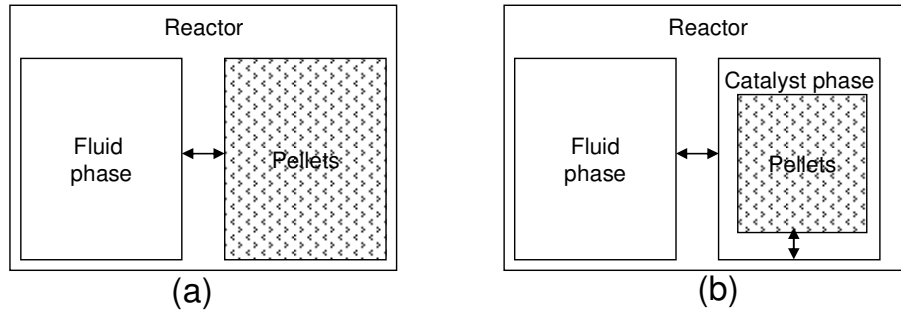


Figure 10. Decomposition of the reactor. (a) two-level decomposition. (b) three-level decomposition.

This view can be manifested to a large extent by just looking at the mass conservation equation of component A in the liquid phase, as part of the model for that phase:

$$\frac{\partial C_A}{\partial t} = -\frac{u_0}{\varepsilon_b} \frac{\partial C_A}{\partial x} - \frac{k_m a}{\varepsilon_b} (C_A - C_{AP} |_{r=R_p}) \quad (17)$$

The second term on the right hand side, if denoted by one quantity, is essentially a *coupling-induced state function*. Furthermore, this term presents a *coupling mechanism law* that characterizes the *coupling* between the

liquid phase and the catalyst phase. In addition, $C_{AP}|_{r=R_p}$ presents the link between the catalyst phase and individual pellets.¹⁰ To characterize the multi-domain integration framework generically and rigorously, we can derive:

Definition D4: Multi-domain integration model. Let σ be a system with levels $V = \{V_k | 1 \leq k \leq n, n > 1\}$.

For any component x on level V_k , let $L(x)$ be the totality of laws on x . Let $C(x)$ be the composition of x . Let $M^m(\sigma)$ be a multiscale model of σ . $M^m(\sigma)$ is a multi-domain integration model, iff all the following statements hold:

(1)

$$\exists V_s \subset V_k, k > 1, \quad \{L(x) | x \in V_s\} = \emptyset \quad \text{and} \quad \{L(x) | x \in (V_k - V_s)\} \neq \emptyset$$

(2)

$$\forall x \in V_k, k > 1, \quad \text{if} \quad L(x) = \emptyset, \quad \text{then} \\ \exists y \in C(x), L(y) \neq \emptyset$$

(3)

$$\forall x \in V_1, \quad \text{if} \quad (\forall y, x \in C(y) \quad \text{and} \quad L(y) = \emptyset), \quad \text{then} \\ L(x) \neq \emptyset.$$

The first condition states that at a certain level above the bottom, some parts are modeled with laws, while no laws are applied to the rest of this level. The second condition states, for any part at a certain level above the bottom which is not modeled, there must be some components of it (located at levels lower than this level) which are modeled with laws. The third and last condition states, if a certain part of a system has never been modeled at any level above the bottom, it must be modeled with laws at the bottom level. The combination of these three conditions ensures that the model includes the models of all of its parts which may be developed not at a single level (or scale).

If we restrict a multi-domain integration model to one in which the parts not modeled at one level are always modeled at the level immediately below it, then the condition for such a model can be simplified as follows:

$$\exists V_s \subset V_k, k > 1, \\ \{L(x) | x \in V_s\} = \emptyset \quad \text{and} \quad \{L(y) | y \in (V_k - V_s)\} \neq \emptyset \quad \text{and} \\ \forall x, z, x \in V_s, z \in C^{im}(x), \quad L(z) \neq \emptyset.$$

¹⁰ This link is essentially an *aggregation law*: the concentration of A in the catalyst phase at the interface with the fluid phase is related to the concentration of A at the surface of each pellet. In this reactor model, all pellets are geometrically identical, so that those pellets at the same axial position have the same surface concentration. Otherwise, certain averaging treatment on different surface concentrations of all the pellets at the same intersection might have to be involved in the *aggregation law*, giving rise to a typical *averaging law*.

Embedded integration. This is a framework where “the macroscale model spans the system domain while the microscale model describes only a portion of that domain”. Applying this framework to the example of catalytic packed-bed reactor yields a model where the reactor as a whole is described by averaging the fluid phase and the catalyst phase, but where the process chemistry are supplied by a detailed modeling of the pellets. Viewing this example from the perspective of system levels, one can see that the reactor system is actually decomposed by the same level hierarchy as in the multi-domain integration (cf. Figure 10b). However, the applications of various laws are different in the two integration frameworks, as revealed in Table 1, where ‘X’ indicates that there are laws actually applied.

Table 1: Applications of various laws in the reactor example

Type of integration framework	Items in the reactor system (cf. Figure 10b) to which laws may apply						
	Reactor as a whole	Fluid phase	Catalyst phase	Pellet	Coupling between fluid phase and catalyst phases (i.e. two bulk phases)	Inter-scale connection between reactor and two bulk phases	Inter-scale connection between catalyst phase and pellets
Multi-domain	—	X	—	X	X	—	X
Embedded	X	—	—	X	X	X	X

To give a generic and rigorous description of models generated by this integration framework, we introduce

Definition D5: Embedded integration model. Let σ be a system with levels $V = \{V_k \mid 1 \leq k \leq n, n > 1\}$. For any component x on level V_k , let $L(x)$ be the totality of laws on x , $C(x)$ be the composition of x . Let $M^m(\sigma)$ be a multiscale model of σ . $M^m(\sigma)$ is an embedded integration model, iff:

$$\exists x \in V_k, k > 1,$$

$$(1) \quad L(x) \neq \emptyset, \quad \text{and}$$

$$(2) \quad \exists y \in C(x), \quad L(y) \neq \emptyset, \quad \text{and} \quad \exists z \in C(x), L(z) = \emptyset.$$

5.3 The heterogeneous multiscale methods by E and Engquist (2003 a, b)

E and Engquist (2003 a, b) generalize a variety of multiscale modeling and computational methods and call them heterogeneous multiscale methods. Intuitively, a mathematical model developed using a heterogeneous multiscale method applies different physics for different scales or levels of the system considered. Using the concepts defined in our ontology, such a model is a *scale-integration model* of a system involving *heterogeneous neighboring levels*.

E and Engquist propose a general procedure of applying such a method, which comprise three steps, namely “reconstruction”, microscale model evaluation, and “compression”. In the “reconstruction” step, certain microscopic scale properties are determined using some macroscopic scale properties before the microscopic model can be evaluated. In the “compression” step, some macroscopic scale properties are obtained through processing certain microscopic scale properties after the microscopic model has been evaluated. Applying concepts defined in our ontology, the “reconstruction” step should be done essentially according to certain *disaggregation laws*, while certain *aggregation laws* are for the “compression” step.

E and Engquist make the distinction between two types of multiscale modeling problems. **Type A** refers to those in which “a macroscopic description is known but ceases to be valid in a localized region in space and/or time”. **Type B** refers to problems in which some macroscopic properties (called “missing data”) can not be made available by the macroscopic model and therefore one has to make use of microscopic models. This is typically the case where some part of the macroscopic model (e.g. a constitutive relation) is unknown or very difficult to obtain. Such a distinction can be expressed with definitions derived previously in this paper: A model of Type A is a *multi-domain integration model* (cf. Definition D4); a model of Type B is a *parallel-integration model* (cf. Definition D2). Of course, to preserve what is originally meant by E and Engquist, any model must be of a system involving *heterogeneous neighboring levels*.

6. Concluding remarks

Multiscale modeling has recently become an interesting topic in various fields. The intended contribution of this paper is the formulation of a conceptualization framework for improving the understanding of multiscale modeling paradigms. The key element in this framework is a clarification of the nature of a scale in the multiscale modeling context. This framework has been developed by following a three-step approach. The first step, i.e. the starting point of this approach, is a conceptualization of general systems. Based on this conceptualization, the second step derives rigorous definitions for precisely characterizing scales and inter-scale relationships. In the third step, the composition of a general multiscale model is introduced, from which a number of multiscale model types are derived, all on the basis of the conceptual foundation built by the two previous steps. This conceptualization

framework is presented in form of an ontology, which reuses part of the existing BWV ontology (Bunge, 1977, 1979; Wand and Weber, 1990). In comparison with the BWV ontology, the major additions introduced in the ontology include the notions of laws on intra-level couplings, the environment and structure of a level, the laws on inter-scale (or inter-level) relations, and finally the notion of a multiscale model and its classification. The key concepts in the conceptualization framework are illustrated by two rather complete multiscale modeling examples. Furthermore, the new framework is compared with the results of some existing clarification attempts in multiscale modeling. We demonstrate that the model types generalized by the previous work are all special cases of the general multiscale model defined in our conceptualization framework. Furthermore, all of the previous model types can be rigorously defined using the concepts offered by the conceptual foundation (i.e. Parts 1 and 2) of our framework.

Different from the existing efforts towards the generalization of various multiscale modeling paradigms, the approach we have taken does not directly attempt to make an abstraction of some existing applications, but rather starts with defining basic concepts of general systems. If the existing efforts are more of an inductive nature (i.e. one which infers of a generalized conclusion on multiscale modeling paradigms from particular instances of multiscale modeling applications), our approach tends to be more deductive (i.e. one which concludes about multiscale modeling paradigms by following necessarily from general premises on systems, laws, scales, etc). We argue that the advantage of this approach is to be able to provide in the first place a common conceptual ground, which can potentially be shared by multiscale modeling researchers from different backgrounds in the course of reaching a sound understanding to multiscale modeling. On the other hand, this conceptualization framework must be regarded as open. Thus, any existing and future applications from different fields may bring extensions to this framework, particularly to the types of inter-scale laws (in Part 2) and the composition and the classification of multiscale models (in Part 3). We emphasize that any extension is expected to be performed in a way similar to the current framework, i.e. every new element should have a precisely defined semantics and be logically related to the existing elements in the framework.

So far we have concentrated on the role of the conceptualization framework in contributing to the understanding to multiscale modeling. In addition, the authors also envisage the potential of this conceptualization framework in the area of computer-aided modeling of systems, particularly with phenomena-based methods. In this area, research is directed to the development of computer tools which can automatically carry out part of the tasks involved in building a mathematical model, by generating mathematical equations based on the physical description of the modeling target as supplied by the human modeller (Marquardt, 1995). One difficulty, with respect to the reusability of such a tool, is that the tool is usually capable of modeling only the type of systems for which specific

domain knowledge has been incorporated in its implementation. To address this problem, an ontology-based modeling approach has been proposed (Yang et al, 2004), which allows a tool to implement only a generic modeling logic while the domain knowledge required for a particular modeling task is supplied separately to the tool as its input. Ontologies play very important roles in this approach, because the domain knowledge is represented through a domain ontology, while a meta-ontology, as an abstraction of different domain ontologies, forms the basis for the generic modeling logic that the tool implements. It has been initially demonstrated that such a tool is capable of processing correctly the physical description of a concrete system formulated according to a domain ontology, which itself instantiates the meta ontology (Yang et al, 2004). In connection to this initiative, the conceptualization framework reported here has the potential of being applied as a meta ontology, based on which a computer-aided modeling tool may be implemented such that multiscale modeling is supported by the tool. To this end, it will become necessary to specify the ontologies (including the meta ontology) using a computer-processable language such as OWL (McGuinness and van Harmelen, 2004). This will also make the ontologies more accessible by the users through ontology browsers. This has been the case of OntoCAPE which, as a formal ontology for computer-aided process engineering, is already represented using OWL (Morbach, Yang, Marquardt, 2007). Ongoing research and development in these directions will be reported elsewhere (Yang, 2008).

Acknowledgements

This work has been supported in part by the DFG in the Collaborative Research Center IMPROVE (SFB 476) and by the EC in the COGents project (IST-2001-34431).

Nomenclature

P	the set of all <i>properties</i>
S	the set of all <i>substantial individuals</i>
Θ	the set of all <i>things</i>
\emptyset	the <i>empty set</i>
$a \in A$	individual a belongs to set A
$a \notin A$	individual a does not belong to set A
$A \times B$	the <i>Cartesian product</i> of sets A and B
$A - B$	the set of objects in set A but not in set B
$A \cap B$	the set of objects in A and in B
$A \cup B$	the set of objects in A or in B

$A \subset B$	set A is part of set B
$A \subseteq B$	set A is equal to set B or part of set B
$c = a \dot{+} b$	c is the <i>juxtaposition</i> of a and b
$x \sqsubset y$	thing x is <i>part of</i> thing y
$x \not\sqsubset y$	thing x is <i>not part of</i> thing y
$P \wedge Q$	the <i>logical “and”</i> between statements P and Q
$\langle x, y \rangle$	the <i>ordered pair</i> of x and y
$C := D$	C is <i>defined as</i> D
$f: A \rightarrow B$	<i>function</i> f map set A to set B

References

- Bertalanffy, L. v. (1968). *General System Theory: foundations, development, applications*. George Braziller Inc., New York.
- Bezzo, F., Macchietto, S., Pantelides, C. C. (2004). A general methodology for hybrid multizonal/CFD models. Part I. Theoretical framework. *Computers and Chemical Engineering* **28**, 501–511.
- Bird R. B., Stewart W. E., Lightfoot, E. N. (1960). *Transport phenomena*. New York: John Wiley and Sons.
- Bunge, M. (1977). *Treatise on Basic Philosophy, Vol. 3, Ontology I: The Furniture of the World*, Reidel, Boston.
- Bunge, M. (1979). *Treatise on Basic Philosophy, Vol. 4, Ontology II: A World of Systems*, Reidel, Boston.
- E, W., Engquist, B. (2003 a). The heterogeneous multiscale methods. *Comm. Math. Sci.*, **1**, 87-133.
- E, W., Engquist, B. (2003 b). Multiscale modeling and computation. *Notices of the AMS*, **50** (9), 1062 – 1070.
- Givon, D., Kupferman, R., Stuart, A. (2004). Extracting macroscopic dynamics: model problems and algorithms. *Nonlinearity*, **17**, R55–R127.
- Gruber (1993). A translation Approach to Portable Ontology Specifications. *Knowledge Acquisitions*, **5** (3): 199-220.
- Ingram, G.D., Cameron, I.T., Hangos, K.M. (2004). Classification and analysis of integrating frameworks in multiscale modelling. *Chemical Engineering Science* **59**, 2171 – 2187.
- Klir, G. J. (1985). *Architecture of Systems Problem Solving*. Plenum Press, London and New York.
- Kulikov, V., Briesen, H., Marquardt, W. (2005). Scale integration for the coupled simulation of crystallization and fluid dynamics. *Chemical Engineering Research and Design*, **83** (A6), 706-717.
- Li, J., Ge, W., Zhang E. J., Kwauk M. (2005). Multi-scale compromise and multi-level correlation in complex systems. *Chemical Engineering Research and Design*, **83** (A6), 574–582.
- Marquardt, W. (1995). Trends in computer-aided process modelling. *Computers and Chemical Engineering*, **20** (6/7), 591-609
- McGuinness, D. L., van Harmelen, F. (2004). OWL Web Ontology Language Overview. W3C Recommendation. Available online at <http://www.w3.org/TR/owl-features/>. Accessed 15 November 2008.

- Morbach, J., Yang, A., Marquardt, W. (2007). OntoCAPE - a large-scale ontology for chemical process engineering. *Journal of Engineering Applications of Artificial Intelligence*, 20 (2), 147-161.
- Pantelides, C. C. (2001). New challenges and opportunities for process modelling. In R. Gani, & S. Bay-Jorgensen (Eds.), *Proceedings of ESCAPE 11 conference* (pp. 15–26). Elsevier.
- Reese, J. S., Raimondeau, S., Vlachos, D. G. (2001). Monte Carlo Algorithms for Complex Surface Reaction Mechanisms: Efficiency and Accuracy. *Journal of Computational Physics* **173**, 302–321
- Rosemann, M., Green, P. (2002). Developing a meta model for the Bunge–Wand–Weber ontological constructs. *Information Systems* **27**, 75–91.
- Uschold, M., Grüninger, M. (1996). Ontologies: Principles, methods and applications. *Knowledge Engineering Review*, 11, 2, 93-155.
- Vlachos, D. G. (1997). Multiscale Integration Hybrid Algorithms for Homogeneous-Heterogeneous Reactors. *AIChE Journal*, **43** (11), 3031 – 3041.
- Vlachos, D. G. (2005). A review of multiscale analysis: Examples from systems biology, materials engineering, and other fluid-surface interacting systems. *Adv. Chem. Eng.*, **30**, 1.
- Wand, Y., Weber, R. (1990). An ontological model of an information system, *IEEE Trans. Software Eng.* **16** (11), 1281-1291.
- Yang, A. (2008). From a generic idea to a generic set of tools: towards computer-aided multiscale modeling. Abstract submitted to: 10th International Symposium on Process Systems Engineering.
- Yang, A., Morbach, J., Marquardt, W. (2004). From Conceptualization to Model Generation: the Roles of Ontologies in Process Modeling. In: C. A. Floudas, R. Agrawal (Eds.): *Proceedings of FOAPD 2004*, 591-594.