Measurement-Driven Simulation of Complex Engineering Systems

Arnoud Visser

Universiteit van Amsterdam

Measurement-Driven Simulation of Complex Engineering Systems

PhD Thesis
Universiteit van Amsterdam

Arnoud Visser

23 October 2007

Lay out: typeset by the author using $\LaTeX 2_\varepsilon$.
Cover design: René Staelenberg.
Cover illustration: Jeroen Bons.

ISBN: 978-90-5629-500-4
NUR: 980

Measurement-Driven Simulation of Complex Engineering Systems

Academisch Proefschrift

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. dr. D.C. van den Boom
ten overstaan van een door het college voor promoties
ingestelde commissie,
in het openbaar te verdedigen in de Aula der Universiteit
op dinsdag 23 oktober 2007, te 10:00 uur

door

Arnoud Visser

geboren te Castricum.

Promotiecommissie

Promotor: Prof. dr. L.O. Hertzberger

Overige leden: Prof. dr. ir. F. C. A. Groen
Prof. dr. P. W. Adriaans
Prof. drs. M. Boasson
Prof. dr. ir. B. van Arem
Dr. G. M. Meijer
Dr. ir. S. J. Gelsema

Faculteit der Natuurwetenschappen, Wiskunde en Informatica

This research was carried out in the context of the Virtual Laboratory for e-Science project (www.vl-e.nl). This project is supported by a BSIK grant from the Dutch Ministry of Education, Culture and Science (OC&W) and is part of the ICT innovation program of the Ministry of Economic Affairs (EZ).

The work reported here is part of the research performed for the Interactive Collaborative Information Systems (ICIS) project, supported by the Dutch Ministry of Economic Affairs, grant nr: BSIK03024.

Contents

Chapter 1

Introduction

1.1 Context

While most sciences concentrate on how they can decompose a system into separate measurable elements upon which they can focus their research, computer science often concentrates on combining elements into versatile computing environments. The last century science has progressed fast, and scientists had to specialize on more and more extreme elements. The extremity can be in small sizes (atoms visible by a tunnelling microscope), large sizes (old galaxies visible by the Hubble telescope), high energies (Higgs-boson visible by the Large Hadron Collider) or in complexity (cell parts studied in the system-biology).

In the 'The end of science revisited' [47], John Horgan claims that, due to this progress, fundamental discoveries are becoming increasingly difficult. Science will only progress in small steps, in contrast to the developments in the previous centuries. Yet, this doesn't mean that scientists can end their work: the only way to truly know the limits of science is to keep trying to overcome them.

For computer scientists it is difficult to accept a limit on the progress, having grown up in a period of explosive scientific and technological progress, reflected by such measures as Moore's law. The American historian Henry Adams already observed a century ago that science & technology were accelerating at that age with an unprecedented momentum, and pointed out that at the underlying force was that more and more scientists dare to trust instruments which superseded their senses [2]. Knowledge gives better techniques; better techniques increase knowledge. Technological advances often enable researchers to overcome seemingly insurmountable obstacles. Computers in particular have vastly increased scientists' capacity for data acquisition, analysis, storage, and communication. Innovations such as optical and quantum computing may extend the reign of Moore's law indefinitely, but this still makes computers no magic wands that will simply solve the toughest puzzles. Computers remain instruments that have to be used wisely. Finding the methodologies to use the computer wisely is task of the computional scientist, and the subject of this thesis for a specific domain.

Our domain is very complex dynamical systems; systems with a very large number of interacting "objects" exhibiting many complicated phenomena. It is intriguing to trace these complicated phenomena back to their source, because chaos theory has shown that interaction of many automata with simple behaviors can already generate very complicated phenomena [31]. When the interacting "objects" are more than automata, but rather entities that can adjust their behavior dynamically (think of the entities that are part of a stock market, a neuron system, an ecosystem, or a traffic system), we are sure that we can never predict their behavior individually, and can only estimate their behavior with a coarse model

Simulation can be used to gain insight into the emergent behavior of the combination of many interacting "objects". Simulation models are capable of mimicking the dynamics at any required level of detail, although this mimicking remains an approximation of reality. Further, there is a trade-off in selecting the appropiate level of detail. First, the lower the level of detail, the higher the computational demands of the simulation. Second, the lower the level of detail, the higher the number of parameters that have to be validated. Yet, there are also benefits to be gained. First, the level of control the scientist has over the circumstances that are evaluated. The scientist can design clean experiments to find dependencies between parameters by scanning over complex surfaces in parameter space. Second, a large variety of phenomena can be tracked. With experiments in the real world, one has to wait till a phenomenon occurs, if one can detect it with the sensors available. In the simulated world a domain scientist has full control and can create infrequent phenomena on request, and observe these phenomena with virtual sensors which are designed to follow precisely those features that are of interest at that moment. So, simulation is a powerful instrument for computer experiments, when used properly, and in combination with experiments in the real world.

Computer and real world experiments are two important methodologies to gain insight in complex systems, when applied individually or iteratively. But when both methodologies are combined more intimately [25], even more insight can be gained, as demonstrated in this thesis. An application simulation can guide the measurement process, creating more effective data acquisition. Vise versa, the measuring process can support the simulation process. New data sources can enhance or refine the simulation by counterbalancing incompleteness in the model. New data sources can be on-line measurements, or archival data that is collected by making the appropriate query. We will call this paradigm MDS-SDM. This acronym stands for Measurement Driven Simulations – Simulation Driven Measurements. We will look what generic methodology can be derived from the experiments we have performed during the evaluation of the "Rekening Rijden" system.

1.2 Challenges

To enable the Measurement Driven Simulations – Simulation Driven Measurements paradigm (MDS-SDM), the following challenges have to be addressed:

Multi-level modeling

Measurements can be made of a large variety of features of the system. To be able to incorporate this large variety of measurements, application models are needed that describe the application systems at different levels of detail. Simulators have to be able to invoke those models dynamically, to be able to make transformations between different levels of detail.

Data fusion and uncertainty propagation

Major problems occur when data (measured or computed) has to be combined that is collected on different spatial or temporal scales. Further precautions have to be taken to prevent small sample sizes and extreme events. With data generated dynamically, methods are needed that estimate the quality and propagation of errors and uncertainty.

Measurement representation / transformation

There is a need developing globally accessible interfaces to complex measurement systems. This can for instance mean that other researchers can have a view on current and archived measurements. This requires new approaches for information management systems, allowing different naming schemas and complex joins of information to support different scientific views on the same 'raw' measurements.

Integration environment

To couple the variety of measurements and models, an environment is needed that supports dynamic selection and coupling of application components. This means new interfaces to measurements systems and models simulators, interfaces that guarantee stable data streams by dynamically match computational and data requirements to the appropriate resources.

These are ambitious challenges. Still, it is worth taking on those challenges and creating environments with some of the features introduced here. As stated by researchers of the RAND institute [26]; it is very useful to already have a part of these features available. Different components, models and information systems with each their own detail, grow slowly together into a family. This integration process is difficult [71], because it requires translations between those components that are not trivial.

Essential is not to have a hierarchical view on the components; to classify one model as high level, and the other one as low level (see figure 1.1) and only to look at the aggregation transformation. Both the high- and low-level models are good ways to describe the same recorded process of going from an initial state (i/I) to a final state (f/F). The process is typically a function over time, and the time is typically represented with a higher resolution in the low-level simulation/measurements. The transformation between the low-level and high-level is seen as an aggregation, a process where information is lost. The aggregation is typically a statistical operation, which can be a simple average, or the fitting of a distribution function. When enough information of the distribution is known, the process can be reversed, a process that is called *disaggregation*. The information that is discarded during the aggregation process is recreated based on domain knowledge (as for instance the shape of the distribution function). This transformation is as fundamental as its reverse. With this transformation it is possible to couple models at multiple levels, to provide multiple interfaces (both to high and low resolution models) and maintain the consistency internally.

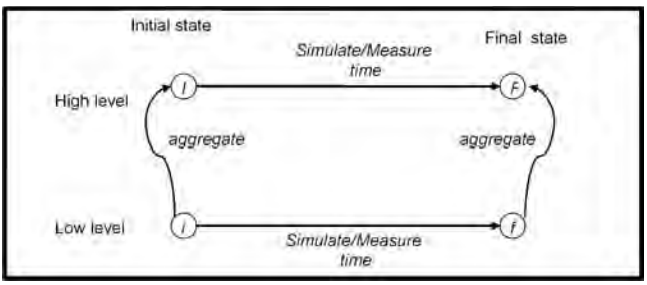

Figure 1.1: Hierarchical view on high-level and low-level application models

This hierarchical view of models is misleading. Each model is build for a certain purpose, which means that some aspects are worked out in detail, while other aspects are represented in an abstract way. Typically, for certain applications, the complexity of high- and low-level models is equivalent, where the complexity of low-level models is concentrated in grasping the dynamics of the physical world, while the complexity of high-level models can be found in strategies and policies. This means that low-level models can benefit from aggregating complex rules into typical usage, while high-level models can aggregate dependencies into typical performance. I.e., transformations are not top-down or bottom-up, but peer-to-peer transformations with at both sides aggregation of unnecessary details and disaggregation of essential distributions. This is illustrated in figure 1.2.

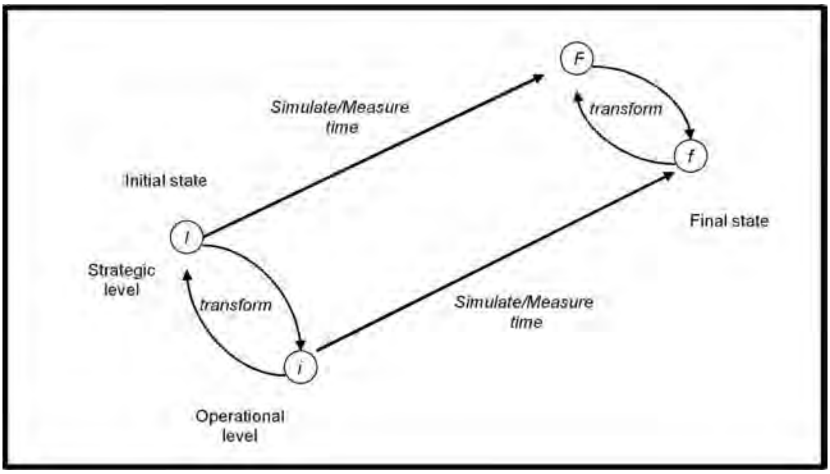

Figure 1.2: Balanced view on high- (strategic) level and low- (operational) level application models

Essential for the transformation, both the aggregation and disaggregation, is that statistics can only be used with care. When for instance a phase-transition can be expected during the simulated/measured time, there exist two independent models (before and after the phase-transition). The parameters of both models have to be estimated separately. This doubles the validation effort, but prevents the distributions of the parameters from being smeared out over an unrealistically large domain.

Further it is important not to forget that both models are approximations of the real world. It is unfair to compare the equivalence of one model with another model after transformation in the context of the former model, because such a transformation is always crude. If one has to compare the applicability of two models, one has to do that based on their predictions of the performance of the system that is of interest of the users. So, both final states have to be projected on users quality measures, and on that level both models have to come with equivalent answers.

Central in this study is the evaluation of the reliability of the "Rekening Rijden" system[1] for the Dutch government. The quality measures imposed by the Dutch government focused on a number of failure events, which should have a very low probability of occurring (as low as 10^{-6}). The evaluation of the "Rekening Rijden" system was performed in several phases. During the first phase of the evaluation it became clear that the requirements could be met during 'normal' circumstances, but that study was needed to characterize the circumstances under which failures were made. This resulted in more detailed models of the "Rekening Rijden" system, which gave equivalent answers for the quality measures for 'normal' Dutch traffic, but also incorporated different answers when the system was tested under 'specific' circumstances. This level of detail required a tight coupling between the modeling & simulation and the measurements & analysis. Special care had to be taken, to ensure that 'specific' circumstances occurred enough in 'normal' traffic to contribute to the failure events reported in the first phase of the project. For a balanced view two-way transformations between the different variables of the models are needed, as we show in this thesis.

1.3 Structure of the thesis

Chapter 1 has introduced the context of this study in a wide perspective. Chapter 2 will survey the literature about complex systems, concentrating on multi-level modeling. The details of the application studied in this thesis are explained more elaborate in chapter 3.

In chapter 4 the methodology that we applied to the model and simulation the system is introduced. Chapter 5 highlights the effort to create realistic stimuli for our system. Multi-level simulation is worked out for one of our subsystems in chapter 6. The environment combine simulation and measurement analysis is described in chapter 7. Our progress is compared with comparable initiatives in chapter 8. The thesis is concluded in chapter 9.

[1] "Rekening Rijden" is the name of a specific Road Pricing variant which was extensively studied from 1995 until 2001 for the Dutch government

Chapter 2

Complex Systems

2.1 Introduction

The domain of this thesis is the study of a complex dynamic system; a system consisting of many subsystems, with complicated interaction between those subsystems. Each subsystem is a macroscopic system in its own right, which can be described by many variables and controlled by many parameters. The subsystems are responding to the stimuli from the environment, which are in the case of the "Rekening Rijden" application the passage of vehicles. The vehicles are steered by their drivers, living beings from which the behavior is not simple to predict.

Yet, biologists have already modeled the behavior of living beings for a long time with success, which evolved into contributions to statistics [72], self-organizing systems [9] and cybernetics [10]. These contributions had indirectly a great impact on the engineering science. The contribution of biology can be explained [69] by the fact that dissipative systems, systems that show irreversible processes, can be associated in biology with progression (e.g. evolution). For physics, dissipative systems are associated with degradation of the system (e.g. friction). Cybernetics evolved later into general system theory [53], where a scientific discipline such as biology is seen as just a particular class of systems. In system theory the informational, relational and structural aspects of a model are dominant, the domain knowledge only a component. Examples of such domain knowledge would be biology, sociology, medicine, physics or chemistry.

The Modeling and Simulation field has its own ambitions. Taking advantage of improvements in software (object-oriented programming) and hardware (faster processors), the creation and execution of impressive simulation models is now possible [98]. The fundamental issues currently under study are model credibility (e.g. validation, verification, model family consistency) and interoperation (e.g. repositories, reuse of components, resolution matching) [24]. This thesis concentrates more on the model credibility [45]. This thesis depends heavily on the complexity concept as introduced in all three fields in the past century, especially on the relationships of multiple models inside a family.

2.2 Cybernetics

Cybernetics was a field introduced at the Second World War by Wiener [94]. In Cybernetics [10] the relationship with complexity is two-fold. On the one hand it is important, because only when systems become complex the methods to cybernetics do show their power. On the other hand the complexity of a system can be hidden, because the focus is on the emergent behavior of the system and the way to steer that behavior. The system is seen as a black box; a central concept in cybernetics.

The black box can be used as model for a system in the real world, which can have many aspects. Only some aspects of the real world will be of interest of the investigator and are represented in the black box. In that sense complexity is not something intrinsic of the system, but related to the model of the system. Complexity is expressed by the interaction between the investigator and the object of investigation. Complexity in that sense is the investigator's view on the system. This point is well-characterized by Ross Ashby [11]:

> I shall measure the degree of "complexity" by the quantity of information required to describe the vital system. To the neurophysiologist the brain, as a feltwork of fibres and a soup of enzymes, is certainly complex; and equally the transmission of a detailed description of it would require much time. To a butcher the brain is simple, for he has to distinguish it from only about thirty other "meats", so not more than $log^2 30$, i.e. about 5 bits, are involved. This method admittedly makes a system's complexity purely relative to a given observer; it rejects the attempt to measure an absolute, or intrinsic, complexity; but this acceptance of complexity as something in the eye of the beholder is, in my opinion, the only workable way of measuring complexity.

So, in Ashby's eyes, no assumptions are made about the nature of the black box and its contents. However, the investigator should have certain given resources for acting on it and certain given resources for observing its behavior. By thus acting on the box, and by allowing the box to affect him and his recording apparatus, the investigator is coupled to the box, so that the two together form an experiment; a system with feedback, as illustrated in figure 2.1.

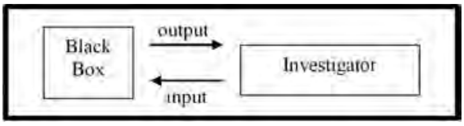

Figure 2.1: An experiment defines the relation between an investigator and a black box

To make the experiment well defined and reproducible, the box's 'input' and 'output' have to be specified. Every real system has an indefinite number of possible inputs (ways to exert some action on the box) and outputs (means to observe and record some behavior of the box). For an orderly experiment the inputs and outputs to be used have to be selected, which means that all other possible inputs and outputs are left alone.

At the moment that the experimenter starts to alter the inputs of the box, some of the outputs will start to change (otherwise the inputs are incorrectly chosen [63]). At a certain moment the experimenter recognizes repeatable patterns in the behavior, and defines states and transitions between those states. States can be described by the instantiation of multiple variables. Some variables show correlations, and the scientist starts to deduce parts and connections within the black box. Yet, Shannon has shown that such behavior can be produced by an indefinitely number of possible networks of parts and connections inside the black box. Multiple system models can show the same behavior. A behavior doesn't specify the parts and the connections uniquely. Occam's razor says that for models with equal predictive powers, we should choose the simplest model. For models with a different number of parts and connections this seems decidable, but shifts at the end the question to what the predictive power of one connection relative to another connection is. In System Theory this is called the Reconstruction Problem.

Because of this ambiguity, whole families of system models can describe the box's behavior, and it is important to be able distinguish if two system models are equivalent (isomorphism) or that one system is a simplification of the other system (homomorphism).

Two systems are *isomorphic* when a one-to-one transformation can be found for each state and variable.

A system is a *homomorphism* of another system when multiple states or variables transform into a single state or variable.

The distinction between homomorphism and isomorphism makes it possible to specify the granularity of the knowledge of a system and to create multi-level models. Yet, for real systems this knowledge is only a small selection of what could be known from the system. Worse, at the moment that the experimenter has the impression that a consistent view of the behavior of the system is acquired, the tendency is to make this model a component in a larger system. A compound system is created by combining a model of a 'known' black box with another 'known' black box. For a combination of a limited number of boxes this approach can work, and the behavior of the overall system can be predicted from the behavior of its parts. Yet, as the compound systems grows, the moment will come that it is realized that an essential property of the overall system is missed, that there is a correlation between states of components not modeled before.

Another issue is the relation between the dynamics of the system versus the dynamics of the environment [69]. On the one hand the behavior of the system has to show long-range order, to be able to identify it as a system with certain macroscopic properties. On the other hand the behavior has to show short-scale randomness; otherwise it would be a deterministic system. The natural spread of responses allows a scientist to explore the state-space of the system. So, variance in its behavior is a natural prerequisite for a complex system, and finding the invariants in the underlying dynamics is the core of Cybernetics.

2.3 System Theory

System Theory [54] distinguishes two generic measures for complexity:

Descriptive complexity is to measure the number of entities involved in the system (variables, states, components) and the variety of interdependence among the entities, as observed by the investigator.

Uncertainty complexity is to measure uncertainty in the prediction of the behavior of the system, when the model is compared to measurements.

When we simplify a system, we want to reduce both the complexity based on descriptive information and the complexity based on the uncertainty information. Unfortunately, these two complexities conflict with each other. Often, when we reduce one, the other increases or, at best, remains unchanged. In System Theory, a set is created of models that can be used to describe a system. The descriptive complexity can be used to select a number of simple models from the set. Each simple model is represented with system behavior function $f(c)$, is compared with the measurements $f_h(c)$, to check if the system is still well enough represented. The measure for this check is the normalized information divergence [23] between the model and the measurements. The information divergence is defined to indicate how well the system behavior reproduces the measured behavior; the information divergence is used as measure for the uncertainty complexity. The information divergence is defined as the loss of information when the behavior of a real system is replaced by the behavior of model system. The loss of information can be expressed by

$$D(f, f^h) = \frac{1}{log_2\|C\|} \sum_{c \epsilon C} f(c) log_2 \frac{f(c)}{f^h(c)} \tag{2.1}$$

for a probabilistic system.

The variable f represents the behavior function of the real system, the variable f^h represents the behavior function of a hypothetical model system, C is the set of behavior states c. A behavior function $f(c)$ defines if a state c occurs ($f(c) = 1$) or not ($f(c) = 0$). The behavior function $f(c)$ can be learned from a set of observations of the real system.

Decomposing a system into subsystems is a common methodology in System Theory, a process that creates Structured Systems. Decomposing in System Theory is actually only a partial solution to battle the complexity. To be able to define a whole-part relation in System Theory, first a common set of the states and variables has to be defined. So the model of a subsystem is only a subset of the set of its supersystem: both systems remain compatible, only supersystem is larger than the other. The benefit of decomposition of a system is that the number of connections between states and variables can be reduced, by ignoring weak and indirect correlations. So decomposition in System Theory is not as powerful as decomposition in Software Engineering, where also states and variables can be made private to the subsystem.

The power of System Theory is that it generalizes systems to such a high abstraction level, that a family of structured system models could be identified that could represent the real system (a set of reconstruction hypotheses). The selection of the best structured system is often impossible, because there are many comparable systems. It is possible to define structures that have equivalence classes (same number of connections), and to define an ordering between those classes (refinement ordering). Refinement is seen as making the system more general, to reduce the number of assumptions, to eliminate as many couplings between subsystems as possible. The ultimate goal of refinement is to reduce the system into independent parts, with no connections between them.

Structures with a certain equivalence class can be refined or coarsened. Unfortunately, in many cases several refinements and coarsenings are possible, each leading to structures in different equivalences classes. When one is looking for the simplest model with the maximum descriptive power, the family of structured systems can grow very rapidly. Yet, the family relation can be used to limit the number of structures that have to be evaluated for their descriptive power. For a number of structures in an equivalence class the descriptive power can be calculated by a measure like the information divergence. The