Integrating Model to Support Decision Making

Guido Guizzi*, Teresa Murino*, Stefania Santini**, Manuela Tufo***, Elpidio Romano*

*Federico II University/D.I.C.M.A.P.I, Naples, Italy  
**Federico II University/ D.I.E.T.I., Naples, Italy  
***University of Sannio/ Engineering Department, Benevento, Italy  
g.guizzi@unina.it, murino@unina.it, elromano@unina.it, stsantin@unina.it, manuela.tufo@unisannio.it

Abstract - The aim of the paper is to investigate the integration of different analytical and simulations tools to support people to make decisions, and to show how the relationships among the different methods can be advantageous to solve specific problems. ORM (object – role modeling), PN (Petri Nets) and SD (System Dynamics) have been combined to capture the static and dynamic aspects of system.

I. INTRODUCTION

Decision support systems (DSS) are necessary tools to improve both the efficiency with which people make a decision and the effectiveness of the decision itself. DSS helps to select the best decision-making strategy to achieve a prefixed objective in particular and complex contests where it is difficult for human to establish what is the right choice to take.

A fundamental step when dealing with DSS is the design of the system conceptual model based on mathematical relationships to represent the complexity of reality. For these complex models it is impossible to manage the mathematic solution and the numerical simulation is the only possible approach. To give strength to the modeling often the Design of Experiment (DOE) technique is used since it allows describing a system, whose behavior is unknown, on the basis of a discrete number of measured data.

In this paper we integrate DOE with other different methods and tools to design and exploit models for DSS. Specifically, we merge Object-Role Modeling (ORM) as a graphical representation for the conceptual model structure, with Petri Nets (PN) and System Dynamics (SD) for the representation of discrete flow and a quantitative flow, respectively. In this article, the first section provides the state of the art and the literature review illustrating the basic concepts behind the above cited methods. Section 3 show how integrate SD with PN, while in Section 4 focus on the dynamic simulation model. The use and the integration of the DOE is discussed at the paper end.

II. LITERATURE REVIEW

Given a certain process the behavior of the system can be described by using different tools. Among them the most successful are the Object-Role Modeling (ORM) to capture the static behaviour of the system, while System Dynamics (SD) and Petri Nets (PN) can be used to mimic its dynamics. SD is a well-known methodology to assist businesses and government organizations in strategy development, analysis of policy options, and analysis of all those dynamic processes where capturing the information flow and describe feedback mechanism are fundamental. A SD model highlights all those feature and factors affecting the behaviour of the system in a causal-loop diagram. This diagram clearly depicts the linkages and feedback loops among the fundamental elements in the system, as well as all pertinent linkages between the system and its operating environment (Andersen Consulting). On the other hand, PNs have been found to be a very strong and heavy formalism to describe a discrete event system (DES). A DES is a discrete-state, event-driven system, where the state evolution depends entirely on the occurrence of asynchronous discrete events over time. Note that DES are useful to describe individual entities, attributes decisions and events, while more classical continuous time SD modelling approaches concentrates on homogenised entities, continuous policy pressures and emergent behaviour. A further useful distinction is that while SD focuses on strategic problems, usually a DES approach is more targeted to describe and solve operational problems. (Lane, 2000). Sometimes in the DSS paradigm to add high quality to Object-Role Model (ORM) is used to integrate a formal conceptualization approach to SD modelling. ORM aims at modelling static objects, and depicts the world in terms of objects that play roles. It includes procedures for mapping conceptual and logical levels and it was originally conceived for data modelling purposes (Halpin, 1998). ORM allows to design and query database models at the conceptual level, where the application is described in terms that can be readily understood by users, rather than being recast in terms of implementation data structures. Sometimes it is called a fact-based modelling tool because ORM verbalizes the relevant data as elementary fact (GiaSurguladze et al.2009). Although its positive features, the fundamental limit of the ORM approach can be summarized according to Halpin and Wagner as: “although ORM support modeling of business term fact, and many static integrity constraints and derivation rules, it cannot model the reactive behavior of systems which can be described using dynamic integrity constraints”. For this reason, to capture the dynamic aspects of a given system the use of SD and PNs is required. In this paper, we show how to integrate these three methods for capturing static and dynamic aspects of a system in a DSS context. Integration of methods is an approach that is often applied to complex software development since it aims to
Is it possible to have which way the object type participate in that fact type. It relationships between object types and they represent sense. Fact types \( R', U' \) are used as predicate-like to denote entities and are a collection of ORM use lines with arrows and Exogenous variables as circles. Information links and Exogenous variables can be defined as links that relay information from an exogenous variable and stock into or out of flows. Information links and Exogenous variables measure the quantities in levels and, through various calculations, control the rates. Information links appear as lines with arrows and Exogenous variables as circles. ORM use object types and fact types. Object types \( X' \) are utilized to denote entities and are a collection of objects with similar properties, in the set-theoretical sense. Fact types \( R', U' \) are used as predicate-like relationships between object types and they represent associations consisting of a number of roles that denote in which way the object type participate in that fact type. It is possible to have unary fact types \( U' \) and binary fact types \( R' \). PNs use places, transitions and arcs. Places \( Y' \), depicted as circles, are inactive concepts. Tokens exist within places and represent the current state of a process. Each PN has one start place, one end place and a number of intermediate places. Transitions \( T' \), depicted as rectangles, are active and represent tasks to be performed. Arcs \( C' \) are shown as connecting lines which can go from a Transition to a Place and vice versa. (Tulinayo, 2008).

To compare these three modelling approaches and to study their features you need to know the key concepts of each of them. To this aim here Figure 1 summarize the key concepts of System Dynamics \( A \), Object Role Modeling \( B \) and Petri Nets \( C \). SD use Stocks (Stock A and Stock B), Information links, Exogenous variables, and Flow rates (Inflow into stock A and Outflow from Stock A into stock B). Stocks can be considered reservoirs containing quantities describing the state of the system. Flows (inflow to and outflow from the various levels) can be seen as pipelines with a valve that controls the rate of accumulation to and from the stocks. The Exogenous variables include information in the form of equations or values that can be applied to stocks, flows, and other exogenous variables in the model. The Information links can be defined as links that relay information from an exogenous variable and stock into or out of flows. Information links and Exogenous variables measure the quantities in levels and, through various calculations, control the rates. Information links appear as lines with arrows and Exogenous variables as circles. ORM use object types and fact types. Object types \( X' \) are utilized to denote entities and are a collection of objects with similar properties, in the set-theoretical sense. Fact types \( R', U' \) are used as predicate-like relationships between object types and they represent associations consisting of a number of roles that denote in which way the object type participate in that fact type. It is possible to have unary fact types \( U' \) and binary fact types \( R' \). PNs use places, transitions and arcs. Places \( Y' \), depicted as circles, are inactive concepts. Tokens exist within places and represent the current state of a process. Each PN has one start place, one end place and a number of intermediate places. Transitions \( T' \), depicted as rectangles, are active and represent tasks to be performed. Arcs \( C' \) are shown as connecting lines which can go from a Transition to a Place and vice versa. (Tulinayo, 2008).

The relationships among these methods can be derived looking to how concepts interact amongst themselves, or the roles they play in the process of modelling a systems. Stocks in SD are similar (though not identical) to unary fact types in ORM and places in PNs since they contain “things”, (all act as containers of quantities (SD), objects (ORM) and tokens (PNs)). Quantity in SD are similar to counting Objects in ORM and Tokens in PNs. They can be seen as quantities that flow within the system or process. Flows in SD are linked to Object types in ORM and to Transition in PNs. They connect different stocks (SD), Unary fact (ORM) and Places (PNs) and transfer objects (ORM), tokens (PNs) and quantity (SD). Finally Information links are linked to fact types (ORM) and to Arcs (PNs) since they are active and have activities that cause a change to the recipient/destination (Tulinayo et al., 2008). By combining SD concepts with the ORM and Petri Nets modelling approach, it is possible to better capture the static part of a process, and to link it with the dynamic aspects (Tulinayo et al., 2009). Once models are created, one of the most relevant problems is the dependence of the simulations or test results on factors that are not explicitly consider in the experiment. So, it is useful to use a DOE (Design of Experiment) approach to identify if and where it is possible to operate to modify or achieve specific targets. The main objective of experimental design is to construct an empirical model to approximate an unknown relationship (in a complex phenomena) up to a given level of accuracy. These empirical models are often algorithmically non expressible, algebraically difficult to model and ill structured. Fundamentally, since an unknown relationship can be denoted as function of appropriate input and output spaces, the modelling objective is often translated into finding a mapping relation between the input space and the output space, which can approximate the desired function within a given accuracy. Experimental design and decision support problems can be therefore conceptualized by mapping the performance requirements of a product or process onto suitable values of the decision input parameters. These are the parameters that describe the physical design or the kind of operation of

![Figure 1. SD, ORM e PNs concepts](image-url)
the manufacturing system. This idea led to a new concept for experimental design based on the function approximation concept, as shown in Figure 2.

**Figure 2. Concepts for experimental design and decision support (Tay Kiang Meng, 2002)**

Hence DOE is a comparison tool that allows the understanding of different solution (factors) effects. To plan a DOE the following steps are needed:

- To evaluate if performances are coherent with the problem
- To create a list of all the factors
- To extrapolate from the list a subset of factors to be investigated
- To select the levels for each factor
- To design a DOE matrix

On the basis of the number of chosen factors it is possible to identify two different methods:

- Full Factorial Experiment
- Fractional Factorial Experiment

Full Factorial Experiment is a method that is independent on number of factors and allows to have a balance plan in which each factor is predictable, independently from the other, while it is possible to evaluate all the interactions (regardless the order). Fractional Factorial Experiment instead allows considering a subset of Full Factorial Experiment, thus reducing the number of test and consequently the cost.

While usually in a classical modeling approach only one single modeling approach (and its simulation tools) is used to deal with a single system or problem, here the idea is to merge and ménage together the fundamental features of different modeling approaches (SD and PNs). To this aim is thus necessary to use a DOE approach to identify the aspects of the system that can interfere to reach or modify the objectives of the analysis in a DSS paradigm. Such kind of experimentation is fundamentally carried out to reply to the following question: "Is this solution better than the other one?" so to understand the relations between causes and effects ($Y=f(X)+\text{NOISE}$) and to make the best decisions on the process under investigation. Note that in our context the noise represents all that influences the process performances that is not explicitly modeled.

### III. INTEGRATING SYSTEM DYNAMICS AND PETRI NETS.

A process is a sequence of activity performed to obtain a specific purpose. The process map is a graphic representation of the flow of activities and it is often used to investigate the system and to understand how it evolves. The tracking of the operations map is a key point since it helps to have a clear view of all the actions to be carried out to obtain the prescribed output and to identify the critical operations to be further improved. An example of a process map is shown in Figure 3:

**Figure 3. An example of Process Map**

The most common tools to map a process are the Flow Charts, that describe the logic links and information flow among process activities. The Flow Charts gives a schematic representation of the process and allows to facilitate analysis of the contents and to support the decisional processes.

The phases of process map consist the following steps: identification of the processes to be analyzed, information gathering to describe the processes and process modeling. The main phase is the process modeling. The concept of “model” is more than a geometric description: it is a simplified representation of a system in a particular state. Among the modeling techniques we can distinguish between analytic and simulation techniques. In an analytic model, system components and weight are represented by variables and parameter, while the interactions between components are represented by the mathematical relationships among these quantities. The system evaluation, made by exploiting the analytic model, requires the calculation of the solution using both analytic method and numerical solutions. Model can be also used to tame the dynamic of the system under investigation that can be continuous, or even hybrid and discontinuous (Di Bernardo, 2013a), (Di Bernardo 2013b), (Di Bernardo 2008).

Mathematical models represent the system behavior that can be static or dynamic, continuous or discrete. SD, with the definition of stocks and flows suggests is one of the possible techniques to solve an analytical model based on continuous or discrete time equations. PNs are instead one of the many modeling languages able to describing discrete events systems.
IV. A DYNAMIC SIMULATION MODEL

Production efficiency, cost and time reduction have become very important factors especially with respect to the serious economic crisis of the last years. Today the concept of Lean Manufacturing, defined as the “the identification and application of best manufacturing practice to eliminate waste and variation” (Bhuiyan, 2005), has become synonym of smart production, technique and instrument to be used into the company processes to optimize time, human resources, activities, and to improve product/service quality level. Thanks to numeric simulations techniques the applicability of the lean production paradigm has been shown. Numeric simulation is the current technique used to lead experiments, involving some mathematic and logic models, describing the economic and industrial system behaviour. Compared with experimentation (expensive and, sometimes, not applicable) the simulation is versatile and can be run quickly with low costs. A simulation is “the imitation of the operation of a real-world process or system over time,” and involves “the generation of an artificial history of a system, and the observation of that artificial history to draw inferences concerning the operating characteristics of the real system” (Bhuiyan, 2005). Simulation is used to assist the decision makers to cope with real-world complexity, and to inform design decisions in advance of implementations that involve significant commitments of resources. Dynamic complexity is present in many natural and human systems and it is determined by causal relationships, involving positive and negative feedbacks, and time delays, while it can give rise to unexpected behaviours (Osorio, 2008).

Historically, the discrete-event simulation (DES) and system dynamics (SD) approaches have developed as two separate and distinct fields. The main differences highlighted include the observation that for DES, the objects in the system are distinct entities, whereas for SD such entities are aggregated into a population and treated like continuous quantities. In addition, the DES system evolves through a sequence of temporal points at which occurs an event that modify the system status. Some events are programmed at beginning of simulation; some others are generated during the execution. Analysis and interpretations of systems behavior developed in SD are based on two concepts: division of state variables and their relations, and feedback loop. Since a complex system has particular characteristic that the emerging behavior is not obtainable through a simple sum of their parts, the entities relationship network produce nonlinear effects that are not explainable studying each component separately. The presence of nonlinearities and complex systems characteristic induce the use of simulation as a valid alternative to analytic models. Among all simulation technique SD is the one closer to mathematic formalism, since it is based on differential equations. Although a differential pattern is based on equations that treat the time in the continuous, its informatics simulation involves a discrete process. Compared to the other simulation technique SD needs a lower use of programming language, thus allowing a very quick pattern design. So far it has been analyzed how to use single methods, comparing characteristic, and integrating them. It has been possible to understand how this is important in a simulation contest to give information on the future development of the model, reproducing patterns with the maximum detail. To this aim a simulation model has been applied to the problem of routing pallet in Whirlpool Emea to identify the best path. To examine this problem the Powersim Studio software has been used. This software allows to graphically representing every complex system to be analyzed through SD methodology. Using different levels, flows, constants, auxiliary variables and information connector it is possible to represent graphically even a complex system and to study its behavior. In Figure 4 an example of logic flow is represented, consisting of the entire production process.

This physical flow, representing the process in Whirlpool plant, is activated by events, similarly to the Petri Network logic, and it covers the whole work areas, starting from the order generation to the finished product. These were simulated in three chains of events, each representative of the work areas sited on the production process place.

In Figure 5, an example of chain of event is represented. The chain is composed of token, that represents a place card moving into the process, occupying tail or unit services. Besides the token flow there is also the information flow. To each token a state variable, usually managed by activities, may be associated. The unit that takes the routing decisions to define the token route may need these specific attributes.

V. SIMULATION MODEL THROUGH DOE FOR SCENARIO ANALYSIS

Once simulation model has been designed, interconnected and at least beta tested, before using it as an investigation tool, some verification is to be carried out. First of all, our process model should be representative at the best of the current situation. This means that if our purpose is to gain useful knowledge, we need to check if any random variation of the outputs under investigation is well represented.

Firstly, we will understand if we really inject noise in our simulation model; secondly to evaluate simulations will be run as real manufacturing processes were available for experimentation. Finally we can have a “quantification” of the noise level in order to evaluate the “significance” of the model. To do this evaluation we have to introduce a new approach for process simulation that is the Design of Experiment (DOE).
Experimentation intent is to force significant events to occur in front of a perceptive observer, able to understand that it contains information that could be even generalized. From a design standpoint, experimentation scope is to create defects in order to understand how to avoid them. The most important standpoint is to experiment in a structured way in order to plan, run and analyze, leveraging on resources, with a pretty straightforward analysis. The main advantages of a DOE could be quickly summarized as follows:

- To maximize the chance of having an informative event.
- Active, not passive strategy, increasing learning opportunities.
- To confirm theories or discover new paths of investigation.

To establish a cause and effect relationship:

\[ Y = f(x) + \text{Noise}. \]  

Planned to balance factor significance it puts us in condition to investigate interactions. In manufacturing and design, there are several available possibilities. As a first step we have to give the following statement.

DOE is defined with the following conceptual formula:

\[ \text{DOE=DS + US } + \text{LOR} \]  

The Design Structure (DS) accounts for manipulated knobs (called factors) used inside the experiment to “force” an event to happen. The Unit Structure (US) defines the experimental noise that we capture inside the experimentation (like units submitted to a treatment, set ups, measurement variation,). The Line of Restriction(s) (LOR) highlights the fact that for any reason treatments cannot be randomized. There are mainly two different types of DOE: Full Factorial (FF) and Fractional Factorial (FFF). With FF we mean that, given a number of factors \( n \) and a number of levels assigned to each factors \( n \), the experiment tests every possible combination of factors at a given number of levels. FF’s are very easy to be both generated and analyzed, but they are very “inefficient” when the number of factors increases. An example of Factor Relationship Diagram is shown in Figure 6 that is a tool effective to assist in the planning of DOEs, the evaluation of alternative design plans, and in the interpretation of DOE results. In industrial world, fractionation, using FFF is adopted. There are also some graphical analysis options like the Normal Probability Plot (NPP), as displayed in Figure 7 and Figure 8 to analyze factorial experiment. Once the model has been considered satisfactory from an engineering and representativeness standpoint, we are now ready to use it for investigation purposes.
From the analysis perspective, the same approach already shown in the Components of Variation Study: Practical, Graphical and Quantitative it has been followed.

First of all we tried to capture how much variation we created manipulating knobs in the DOE. In order to show this a Time Series Plot, sorting outputs from the minimum value to the maximum one (Fig. 9) has been used.

![Figure 9. A generic Time Series Plot](image)

Next step is the Graphical one: we verified the Statistical Significance (SS) of the effects, comparing them with the right level of noise, as:

$$\hat{E}(\text{factor}) = E(\text{factor}) + \text{any confounding} + \text{noise} \quad (3)$$

Involved tools are NPP and Pareto analysis, using Lenth’s PSE (Figure 11)

![Figure 11. A generic NPP & Pareto plot](image)

The following step has been to investigate the Main Effects and the Interactions found out to be Statistically Significant (all the effects above the red line, called sometimes water line). In Figure 11 is shown a generic Main Effect Plot. In the Interaction Plot (Figure 12) optimum and robust working conditions are shown with colors: red circle is the optimal setting (the same for Main effect); purple circle represent a robust setting. This is the phase in which we will compare results with predictions, to understand if we discovered new things or we confirmed the current theories. Only active factors will be considered in the next steps of investigations. Good practice will require also including a final Graphical Summary to highlight the relevant work directions (Figure 13).

![Figure 12. A generic Interaction Plot](image)

![Figure 13. A generic Graphical Summary using an Annotated Value Plot](image)

Finally, a Quantitative step has been implemented. A quick and good calculation tool could be derived using the following generalized linear model GLM:

$$Y = \beta_0 + \beta_1x_1 + \beta_2x_2 + \beta_{12}x_1x_2 + \ldots + \epsilon \quad (4)$$

Where Y is the performance under investigation, x₁, x₂, x₃, etc., are the manipulated knobs and the relevant interactions, while β coefficients are the half value of the effects (for 2 level design only) of each factor and interaction.

In the model we have included only active factors, defined to the factors whose effects are statistically significant (using Lenth’s PSE criterion), have a practical meaning and match predictions.

**VI. CONCLUSIONS**

In this paper the creation of a model integrating two different methods is highlighted with SD macroscopic (but not linear) aspects and PNs microscopic (but linear) aspects.
Moreover, all different method properties have been underlined, showing the relationships linking the fundamental aspect of each of them.

It has been shown why these models are useful for simulation, in particular, simulation allows to obtain effective results, more versatile and quicker than the results that can be obtained with a direct experimentation.

This model, developed according the Petri networks logic, is divided into a phase of physical flow construction, that is the synthetic representation of the whole production process, and then we detailed showing the chain of events. This integration method is useful for many simulation approach in order to obtain a great support to make strategic decisions. Finally we have defined by DOE what variables are “principal” in order to explain the generic phenomenon.

REFERENCES


[4] D. Pfahl and K. Lebsanft Integration of system dynamics modeling with descriptive process modeling and goal-oriented measurement, April 1999


[9] Lane, You Just Don’t Understand Me: Modes of Failure and Success in the Discourse between System Dynamics and Discrete Event Simulation, 2000

[10] Andersen Consulting, A Comparison of System Dynamics (SD) and Discrete Event Simulation (DES), 1999


