MODELING ISSUES WHEN USING SIMULATION TO TEST THE PERFORMANCE OF MATHEMATICAL PROGRAMMING MODELS UNDER STOCHASTIC CONDITIONS

Anne-Laure Ladier
Université de Lyon, INSA Lyon,
DISP laboratory EA 4570
Bât. L. de Vinci, 21 av. J. Capelle
69621 Villeurbanne, France
anne-laure.ladier@insa-lyon.fr

Allen G. Greenwood
Poznan Univ. of Technology
Faculty of Engineering Mgmt.
Strzelecka 11, 60-965
Poznan, Poland
allen.greenwood@put.poznan.pl

Gülgün Alpan
Univ. Grenoble Alpes, G-SCOP,
F-38000 Grenoble, France
CNRS, G-SCOP, F-38000
Grenoble, France
gulgun.alpan@g-scop.fr

KEYWORDS

Discrete-event simulation, discrete mathematical-programming optimization, validation, verification.

ABSTRACT

Discrete-event simulation (DES) models and discrete mathematical-programming optimization (DMPO) models are often used together in a variety of ways. This paper discusses the issues that modelers must address when using DES models to test the performance of DMPO models in a stochastic environment. The issues arise during validation of the simulation models – comparing the simulation results under deterministic conditions with results from deterministic optimization models. In our case, the issues are derived from validating simulation models that are used to test the performance of scheduling and resource allocation models (integer and mixed-integer programming optimization models) under various types of uncertainty. The models are from our work in crossdocking operations; however, we believe they are relevant to a wide variety of problem domains. In addition to describing the issues, we offer suggestions on how modelers might address the concerns.

INTRODUCTION

Modelers often employ both simulation and optimization models, combined or related in various ways, to address a particular problem. This paper identifies four relationships between simulation and optimization models that allow the two disparate modeling types to be combined to address a specific problem. These relationships, as illustrated in Figure 1, are defined as to whether they are recursive or not and as to which model type utilizes (or is supported by) the other.

The relationship in Panel (a) of Figure 1 is recursive with a simulation model utilizing an optimization model. An optimal decision is made within a simulation model and thus optimization supports simulation. For instance, Clausen et al. (2012) simulate the operations within a logistic network using optimization (multi-stage mixed-integer program, solved with a modified tabu search) to make decisions regarding the routing between the different terminals. In order to realize this relationship, the optimization is typically embedded within the simulation model.

Panel (b) of Figure 1 also illustrates a relationship where a simulation model utilizes an optimization model, but the relationship is non-recursive. In this case, a simulation model is used to test the results of an optimization model, e.g., a schedule. Wang and Regan (2008) propose two time-based algorithms for the inbound truck scheduling problem in a crossdock, evaluated with a detailed simulation model. Liu and Takakuwa (2010) test the inbound truck schedule and the employees' schedule in a fresh-food crossdock operation using a simulation model. Deshpande et al. (2007) use discrete-event simulation to evaluate the performances of various heuristics for the problem of assigning trucks to the different doors of a crossdocking platform.

The relationship in Panel (c) of Figure 1 is recursive with an optimization model utilizing a simulation model. In this case, a simulation model is typically embedded within an optimization model and the simulation is used to evaluate the objective function associated with a solution obtained from the optimization model. Olafsson and Kim (2002) provide tutorials for this technique, which they refer to as "simulation optimization." Greenwood et al. (2005) describe embedding simulation and optimization models in a decision support system to improve shipbuilding operations. In the logistics field, Aickelin and Adewunmi (2006) use simulation as a black box to evaluate the objective function within a meta-heuristic for the cross dock truck-to-door assignment problem. In a different approach, Almeder and al. (2009) translate the solution of the optimization model into decision rules for the discrete-event simulation, and apply the procedure iteratively until a stable point is reached.

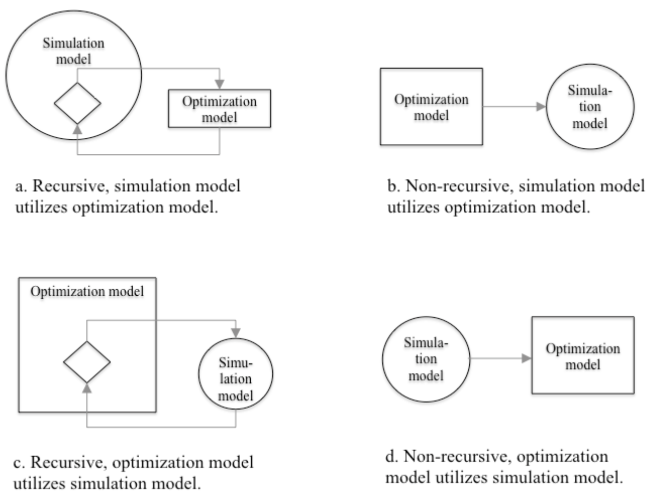

Figure 1: Complementary uses of simulation and optimization models

In Panel (d) of Figure 1, the relationship is non-recursive : simulation models generate data that are then used in an optimization model. For example, to address a personnel planning problem at a crossdocking center, Liu and Takakuwa (2009) use a simulation model to determine the workload needed. These data are inputs for an integer programming model which produces an optimal schedule for the operators, taking their skills into account. In another example, Hauser (2002) uses a simulation model to provide data on alternative layouts in a manufacturing plant.

The focus of this paper is on the non-recursive relationship where a simulation model utilizes an optimization model, as illustrated in Panel (b) of Figure 1. Based on our experience with developing and testing simulation and optimization models that interact in this manner, we detail and explain the modeling issues raised by such a relationship. We explain how those issues can be solved or circumvented. The goal is to provide the modeling community with useful insights on this application of simulation and optimization and to encourage and further enable the use of discrete-event simulation models as a means to assess the performance of optimization models.

Discrete mathematical programming optimization (DMPO) models can represent systems in a very realistic way, taking into account as many details as the simulation does; however, adding too many details makes the solution non-computable. Assumptions are often made in order to simplify the optimization model and focus on the most salient aspects. A discrete-event simulation (DES) model can be used to validate those assumptions and to determine their validity range. On the other hand, some simplifications can be made in the DES model in order to closely follow the assumptions made in the DMPO model. This is important in order to validate those assumptions.

To validate a model is to determine whether or not it is a meaningful and “accurate” representation of the real system, and contains sufficient accuracy to meet its intended use. It is about “building the right model.” Verification is the process of determining whether a model is working as intended. It is about “building the model right.”

In order to validate and verify the DES model, one expects it to behave similar to the DMPO model under deterministic conditions. In a second step, the DES model will be used under realistic, stochastic conditions in order to assess the performance and robustness of the DMPO schedules. This

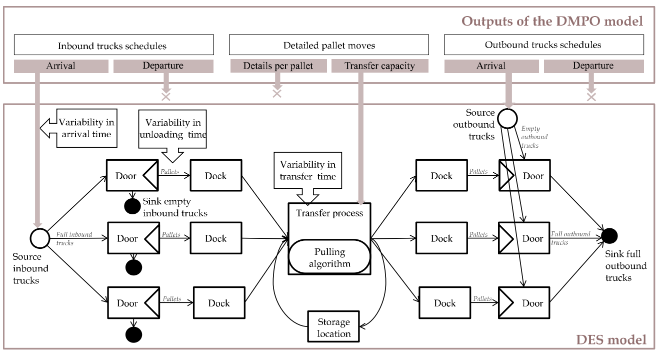

Figure 2: Case 1 − DES flow diagram, links with the DMPO model

paper describes how, due to differences in the modeling approaches, disparities can occur even in the first step (validation and verification), when the models are developed to represent the same system in the same operating environment. The examples on which our observations are based come from the logistics domain (crossdocking operations), but we believe they can be relevant to a wide variety of problem domains. In addition to describing the issues, we offer suggestions on how modelers might address and solve them. Therefore, this article seeks to help modelers in the use of discrete-event simulation to assess the performance of mathematical optimization models.

BASES FOR IDENTIFYING MODELING ISSUES

The modeling issues defined in this paper are the result of testing, using discrete-event simulation, two optimization models for robustness under operational conditions that differ from those explicitly considered in the mathematical formulation, e.g., operating in a stochastic environment.

In the first case (referred later as “Case 1”), test schedules are obtained using the DMPO program described in Ladier and Alpan (2014). A schedule is generated for inbound and outbound trucks to a crossdocking facility that maximizes transportation providers' satisfaction (in terms of the closeness to their desired arrival and departure times) and minimizes total quantity of items placed in temporary storage (rather than being directly loaded onto an outbound truck). The obtained schedule gives the exact arrival and departure time of the inbound and outbound trucks, as well as the detailed pallet moves inside the platform. A key assumption in the optimization model is that unloading, scanning, transfer, loading and departure operations can all be done within the same time period (e.g., 60 minutes) if the inbound truck and the outbound trucks are both present. That is, the time period is long enough to ensure pallets can be transferred to storage, or to their outbound truck, in masked time. The transfer capacity inside the platform (i.e., the quantity of pallets that can be moved at each time period) is limited. Also, the distance of the transfer (thus the location of the doors) is not taken into account. We refer the interested reader to Ladier and Alpan (2014) for more details about the DMPO model and assumptions.

A DES model is used to test the schedules’ robustness when subjected to various levels of randomness, e.g., early or late truck arrivals (modeled with exponential distributions), variations in process times (unloading and transfer, modeled using triangular distributions). Figure 2 shows a simplified flow diagram of the DES model. The diagram identifies the

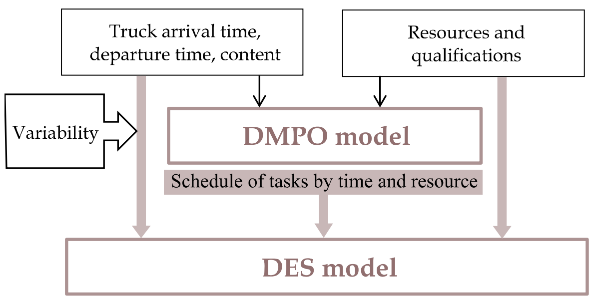

Figure 3: Case 2 – Links between DMPO and DES models

sources of information from the DMPO model that are used by the DES and the sources of variability which provide the stochastic environment for the test.

Case 2 takes place in the same platform, but the focus in on the employees rather than the trucks. The truck schedule and truck door assignments are inputs of the problem. Test schedules are generated using the DMPO model described in Ladier et al. (2013), which uses three mixed-integer linear programs solved in sequence. The sequential solution processes results in detailed timetables (with 15-minute precision) for the employees of the logistics facility. The task assignments have to cover all of the workload for one day, while taking into account the employees' competencies by assigning each of them to tasks for which they are most proficient. More details on the assumptions and the solution methods can be found in Ladier et al. (2013).

A DES model is used to test the robustness of the timetables generated by the DMPO, when subjected to randomness in the amount of workload. In this DES model, the workers are therefore explicitly represented, with their own capacities and their respective competencies. Figure 3 shows the links between the DMPO and the DES model in Case 2.

Both simulation models were developed using the simulation software FlexSim© (www.flexsim.com). Although both models address crossdocking scheduling problems, we believe that the issues raised in the next sections are not unique to logistics or crossdocking, and can be encountered in other modelers application domains.

FOUNDATIONAL DIFFERENCES

The first set of dissimilarities between DMPO and DES models include foundational differences in the ways the two modeling approaches represent the underlying system. These differences are described in terms of time representation, spatial representation, model structure, and model size.

Time representation

How the passage of time is represented in models constitutes a major difference between DMPO and DES. Temporal DMPO uses discrete time intervals where events and resulting activities occur within a time period. For example, within a one-hour time interval a truck arrives for unloading or a task is assigned to, and completed by, an employee. All that is considered is that these events/activities occur somewhere within the interval; the exact time is not important to the model. However, DES has a much finer granularity, events occur at precise instances of time; e.g., a truck arrives 27.1752 minutes after the arrival of the previous truck. Also, in simulation, events trigger, and are triggered by, other events; therefore, timing is an important element.

Because of these key differences, the behavior of a DMPO model using discrete time intervals and the behavior of a DES model will rarely be matched exactly. In Case 1, the DMPO model only allows a truck to leave at a multiple of the time interval considered, e.g. 60 minutes, while the trucks in the DES model leave when a specified condition is met, e.g., when a truck is empty (inbound) or full (outbound). Therefore, if we compare the truck departure times as calculated by the DMPO model and as observed in the DES model, we incur time differences as large as 59 minutes even though both models behave as expected. Those differences can be reduced by shortening the time intervals used in the optimization model; however, that makes the optimization model more complex (and possibly incomputable) and some differences will always be observed. One way to circumvent this issue is to measure performance in terms of intervals. For example, assuming the masked time is 60 minutes in the optimization model, then if a departure is planned at 17:00 in the optimization model and if the truck departs at 17:11 in the simulation model, then the truck departure is considered "on time" and there is no difference in the model results.

Modelers should therefore be aware of the differences in granularity of the modeling approaches and they can circumvent this issue by using time intervals rather than absolute time for their simulation measures.

Spatial representation

DES models not only consider events in time, they often consider spatial relationships among modeling elements and the effects these relationships have on system behavior and performance. Most simulation software integrate and enable the use of locational data to determine activity times; e.g., each travel time in a DES model may be based on the current location of a transporting resource, its destination(s), speeds and possibly acceleration, etc. This granularity is not always considered in optimization formulations.

DMPO models can take into account speed and acceleration – but this adds considerably to model complexity. Therefore, there is a tradeoff between fidelity in the optimization model (zero travel times) and closeness to realistic operations. As a result, spatial effects are taken into account in DMPO models only if they significantly impact the key performance measures that are used in decision making. For example, the selection of an alternative may be heavily influenced by the distance walked by employees in a crossdock facility. If spatial considerations are not at the core of the problem, then mathematical programming modelers tend to ignore travel time or use masked time in order to simplify the optimization models. An action time that is "short enough" can be considered as instantaneous, i.e., performed within the formulated time interval (Case 1).

The difference that DES models typically consider the spatial aspects of systems being modeled, and DMPO formulations do not, causes cross-model validation challenges. To mitigate this issue we propose a compromise approach: control the transfer time by making it a process step in the simulation that does not consider distances and speeds. This easily permits setting the transfer time to zero so that it can be compared to the mathematical programming model, yet enables an easy extension to the simulation model in order to incorporate more realistic aspects, such as probability distributions and location/speed considerations.

Model structure and size

Model complexity is often defined by the model structure and its size. DES models and DMPO models define model structure and size quite differently.

In DMPO, size is not of special concern in formulating or describing a model since the constraints are specified in a tight mathematical notation and input parameters are provided in a structured manner. However, the size or theoretical complexity of the problem drives the choice of solution method, and therefore the solution accuracy and speed. Optimal solutions may be found, but if the problem is NP-hard, execution time increases exponentially with the problem size. On the other hand, heuristics do not guarantee optimality, thus affect accuracy, but can be easily scaled and provide solutions for big data sets.

In DES, model size is defined only partially in terms of the number of objects considered (number of processor units, number of workers, etc.). The model structure considers the types of objects used and, more importantly, the number and type of relationships among the objects. Size, and thus complexity, is heavily dependent upon structure and in particular the number and type of relationships. Even if a DES model has been designed to be easily scalable, the relationships among objects makes scalability in most cases a significant challenge − it is necessary to change the structure in order to change the size. In contrast to the DMPO models, the complexity of a DES model does not affect the choice of solution method and only slightly impacts solution speed (the model run time increases linearly with the problem size).

Typically, models are verified, at least initially, using small size and structure, typically few objects, few time periods, or both. However, it may be necessary to test models in larger contexts. For example, Case 1 was validated for a crossdock facility model with 3-input doors and 3-output doors, but a realistic case would be a 50-input doors by 50-output doors arrangement. Since it may be difficult to scale up the structure of simulation models, and since changing the optimization solution method requires considerable research and development, it is important to specify the size early on in the project.

OPERATIONAL DIFFERENCES

The second set of dissimilarities between DMPO and DES include operational differences in modeling the underlying system. These differences are described in terms of task dependencies, resource assignment and process logic.

Task dependencies

Precedence relationships are used to define the order in which tasks occur. For DMPO models, if the order is not a key consideration, it will typically not be included in the model for the sake of simplification and computation time. In that case only the number of tasks happening in a time interval will be considered; the order, the batch size, and the parallelism of the tasks are not taken into account.

However, in DES modeling, processing order is inherent: typically, unless explicitly specified, tasks are executed in first in, first out order. This fundamental difference can lead to discrepancies between the models during validation. For example, consider a single-channel process (c = 1) working at a rate r and a multi-channel process with c channels and rate r/c per channel. The output from the two options appear to be the same – they are on the average, but may not be true within a time interval. We use the pallet transfer process from Case 1 as an illustration. Assume the transfer rate per resource is r = 10 pallets/hour and the number of available resources is c = 3. If an outbound truck arrives at 10:00, then any pallet transferred from inbound before that time goes to storage, while any pallet processed after 10:00 goes directly into the outbound truck. A process with capacity c = 1 and rate per channel of r = 30 pallets/hour transfers each pallet in 2 minutes. Therefore, between 9:55 and 10:00, two pallets are processed and they both go into storage. However, a process with capacity c = 3 and rate per channel of r = 10 pallets/hour transfers each pallet in 6 minutes. Therefore, between 9:55 and 10:00, no pallet is fully transferred and no pallet goes into storage.

Modelers need to be aware of how basic processing order, batch size and precedence relationships are handled in each type of model. Typically, this is implicit in DES models and explicit, and often ignored, in DMPO models.

Resource assignment

The basic manner in which resources are selected for use may differ between DMPO models and DES models, thus leading to discrepancies in results and validation challenges. A common application of mathematical programming models is to make assignments between resources and tasks, as in our Case 2. When DES is used to test the implementation of an assignment, the default simulation logic may not result in comparable results. For example, in a simulation if a task needs to be performed by a resource and several resources are available, a default first-in, first-out criteria may not match the optimized assignment. Therefore, information on the DMPO assignments must be provided to the DES model so that the task can select the appropriate resource. In addition, if none of the available resources result in a match with the optimized assignment, then logic must be provided in the DES model in order to guide the task's selection from the available resources; or, the task must wait until the appropriate resource is available.

Similarly, in a DES model if a resource becomes available and there are multiple tasks that need to be completed, a default first-in, first-out criteria may not match the optimized assignment. Therefore, as indicated above, information on the DMPO assignments must be provided to the simulation so the resource can select the appropriate task. In addition, if none of the tasks result in a match with the optimized assignment, then logic must be provided in the DES model in order to guide the resource's selection of the available task; or, the resource must be made idle and wait until an appropriate task becomes available.

In short, modelers need to address ways to incorporate DMPO results into DES models, typically by modifying default resource assignment logic inherent in the simulation.

Process logic

By its nature, DES is greedy, i.e., it processes all items (pallets in our cases) that are scheduled in an event (instance in time), while DMPO models can transfer less pallets per time period if it improves the objective function in the optimization. In our Case 1, in order to force the simulation model to obtain a result similar to the optimization model, the amount of pallets that can flow through the model during each time period needs to be limited. This can be accomplished by directly using the output of the DMPO model as input to the DES, i.e., the capacity of the transfer process in the simulation model. Of course, this capacity will need to vary over time. It is interesting to note that this adjustment may make the simulation closer to reality. Therefore, the process logic issue can be mitigated by adding logic to the simulation model that provides flexible capacity over time to the activity.

Since DES is event-driven, priorities are often required in order to represent the appropriate behavior. For example, if both an inbound truck needs to be unloaded and an outbound truck needs to be loaded, which should an available resource service first? In Case 1 and Case 2 we include process logic in the DES model to push items from inbound trucks and pull resources from the arriving outbound truck. The pulling algorithm gives the priority to the outbound trucks; thus, we first seek to fill the outbound trucks that have to leave rather than emptying the inbound trucks. This logic is similar to what a manager would do. However, it may not agree with the optimal solution given by the DMPO model in all cases.
Therefore, it is important to note that, when testing a DMPO model with a DES model, the former gives the optimal solution (when exact solution methods are used) while the latter does not. While the simulation can be driven towards a solution closer to the optimal, it cannot determine the optimal solution unless it embeds an optimization module (this is the case (a) in Figure 1, and beyond the scope of this paper). Using the optimal solution determined by the DMPO model as an input in the DES model is a good approach, but the simulation model needs to include decision logic for handling changes due to stochastic events.

CONCLUSIONS

DES and DMPO modeling take very different approaches to address operations problems. Of course this is due to fundamental differences in the way the two types of models are structured and solved. Even though quite different, simulation and optimization are often used in complementary roles to improve the decisions that result from using the models. These inherent differences provide challenges to modelers, especially in validation and verification.

This paper is based on the cases of two simulation models that are used to evaluate the performance of DMPO models (integer programming, mixed integer linear programming) in a stochastic environment. Those application cases model crossdocking problems, but the issues we point out can occur regardless of the application field. We describe several key challenges occurring when the simulation models have to be validated, i.e., when the behavior of the simulation model and the optimization model are compared under deterministic conditions. We offer suggestions for mitigating those challenges.

We hope that the insights given on these issues can and will encourage an increase in the use of DES to assess the performance of DMPO models. We also hope that other modelers encountering various modeling issues will be encouraged to communicate them so that the community can benefit from their experience.

REFERENCES

Aickelin, U., & Adewunmi, A. (2006). Simulation optimization of the crossdock door assignment problem. In *UK Operational Research Society Simulation Workshop* (pp. 1–3). Leamington Spa, UK.

Almeder, C., Preusser, M., & Hartl, R. F. (2009). Simulation and optimization of supply chains: alternative or complementary approaches? *OR Spectrum*, *31*, 95–119.

Clausen, U., Goedicke, I., Mest, L., & Wohlgemuth, S. (2012). Combining simulation and optimization to improve LTL traffic. *Procedia - Social and Behavioral Sciences*, *48*(1), 1993–2002.

Deshpande, P. J., Yalcin, A., Zayas-Castro, J., & Herrera, L. E. (2007). Simulating less-than-truckload terminal operations. *Benchmarking: An International Journal*, *14*(1), 92–101.

Greenwood, A. G., Vanguri, S., Eksioglu, B., Jain, P., Hill, T. W., Miller, J. W., & Walden, C. T. (2005). Simulation optimization decision support system for ship panel shop operations. In *Winter Simulation Conference* (pp. 2078–2086). Orlando, Florida, USA.

Hauser, K. (2002). *Simulation and optimization of a crossdocking operation in a just-in-time environment*. University of Kentucky.

Ladier, A.-L., & Alpan, G. (2013). Scheduling truck arrivals and departures in a crossdock: earliness, tardiness and storage policies. In *International Conference on Industrial Engineering and Systems Management*. Rabat, Marocco.

Ladier, A.-L., Alpan, G., & Penz, B. (2014). Joint employee timetabling and rostering: a decision-support tool for a logistics platform. *European Journal of Operational Research*, *234*(1), 278–291.

Liu, Y., & Takakuwa, S. (2009). Simulation-based personnel planning for materials handling at a cross-docking center under retail distribution environment. In *Winter Simulation Conference* (pp. 2414–2425). Austin, USA.

Liu, Y., & Takakuwa, S. (2010). Enhancing simulation as a decision-making support tool for a crossdocking center in a dynamic retail-distribution environment. In *Winter Simulation Conference* (pp. 2089–2100). Baltimore, USA.

Wang, J.-F., & Regan, A. (2008). Real-time trailer scheduling for crossdock operations. *Transportation Journal*, *47*(2), 5–20.