Proceedings of the 2022 Winter Simulation Conference
B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, E. Song, C.G. Corlu, L.H. Lee, E.P. Chew, T. Roeder, and P. Lendermann, eds.

A TALE OF THREE SIMULATIONS FOR PROJECT MANAGERS

Sanjay Jain

Department of Decision Sciences
George Washington University
2201 G Street NW, Suite #415
Washington, DC 20052, USA

ABSTRACT

Simulations are a good way for project managers to assess the impact of uncertainties on project plans and subsequent execution. Three types of simulations have been used in literature and to a lesser extent in practice for such purpose, Monte Carlo, discrete event, and systems dynamics. It behooves the project managers to understand the applicability of these different simulations and associated advantages and disadvantages. This paper presents the approach to build a systems dynamics model based on a project plan and compares the learning with those from Monte Carlo and discrete event simulations of the same project plan. The paper will help analysts focused on single type of simulation in the past to appreciate the capabilities of alternate approaches. It will also help practicing project managers to appreciate the effort involved, the analysis generated from the three simulations, and factors to determine which one to employ based on their objectives.

1 INTRODUCTION

Planning and executing any project beyond trivial sized ones is a complex task with the complexity increasing exponentially with project size and novelty among other factors (Bakhshi et al. 2016). The sponsors and the manager of a new project have to hammer out clearly defined deliverables and use them to generate activities needed to produce the deliverables. While arriving at clearly defined deliverables and identifying activities have their own challenges, their complexity pales in comparison to developing a project plan that accommodates uncertainties and has a completion date that can be committed to the internal and/or external clients. Missing committed completion dates many times triggers penalty clauses with associated costs. Project managers have the unenviable task of using widely varied estimates of activity durations and resource requirements from a number of different sources to develop the plan.

A number of different types of uncertainties have to be considered when planning a project. These include the uncertainties due to activity estimations and those due to known and unknown risks that may affect the project. The uncertainties due to activity estimations have also been identified and assessed as a risk (Khamooshi and Cioffi 2013). The risk refers to the effect of the estimates used for project plan being different from the actual duration realized during project execution. Actual activity durations vary based on a number of factors including the morale of the team, variations in the inputs to the activity, and variations in the micro and macro environment. Indeed, most project planning text books recommend capturing the associated variability with each activity duration estimate in the form of representative random distributions or at least as three-point estimates (optimistic, most likely, and pessimistic). The impact of known and unknown risks that interrupt the execution of some or all the activities of a project may be assessed via their impact on individual activities and/or additional planned mitigation activities with their own variabilities.

The impact of all uncertainties on the project plan needs to be quantified. A layered approach is recommended for such assessments with a set of uncertainties added to the plan in each pass to be able to quantify its impact individually. The activity estimation uncertainties are generally included first to help identify an initial project plan with an associated completion date estimate. Subsequent layers may include adding uncertainties due to identified risks and their associated impact and mitigation activities. The base plan would generally accommodate estimation uncertainties and "usual" risks. For example, a construction project with planned execution during winter in the northern part of the United States will include consideration of average days lost due to snowstorms. Additional scenarios may be planned and analyzed for mitigation of selected known risks.

Simulation is the primary technique for quantifying the impact of uncertainties in general and in particular in the project management environment. Multiple simulation paradigms, that is, multiple types of simulation may be used individually or jointly in a hybrid manner for project management level objectives. The three simulation types that are used individually or jointly for these objectives are Monte Carlo simulation (MCS), discrete event simulation (DES), and system dynamics simulation (SDS). Some researchers such as, Suslov and Katalevsky (2019), suggest Agent Based Simulation (ABS) as being applicable in the context of project management modeling. ABS has been relatively a recent development in the field of simulation and few applications in project management domain have been reported. Also, while ABS uses a different perspective than DES and SDS to model systems, such models can be implemented using DES and SDS software. ABS has not been treated separately in this effort but may be explored in future. Additional simulation types such as various engineering or physics-based simulations may be applicable for technical explorations depending on the project domain.

Project managers gain from developing an understanding of the impact of uncertainties. The author has found in his experience with graduate and executive education programs that even some practicing project managers find it surprising that a project planned with average activity durations and no added buffers has around 50% chance of completion on time. Most experienced project managers have an appreciation of the effect of uncertainty. While a number of project managers appear to rely on past experience and thumb rules to build time and resource buffers to accommodate uncertainties, project managers in organizations with mature project management practices sometimes use simulation for quantifying the impact of uncertainties and determining the size of needed buffers and project completion date. Use of simulation among project management practitioners appears to be limited due to lack of understanding of the technique and exposure to such applications. Leading graduate programs in project management generally include discussion of Monte Carlo simulation in project planning and scheduling courses and a few allow access to electives on discrete event and system dynamics simulations.

This paper has the long-term goal of improving the performance of projects, in particular, via developing realistic schedules and budgets that can be actually met. The premise is that appropriate use of simulations will allow development of schedules and budgets that comprehend the various known uncertainties. Simulations can over time also help improve policies and procedures that in turn further improve performance of successive projects. In the short term, the paper has dual objectives. On one hand, it attempts to encourage simulation analysts to consider appropriate simulation types to meet the needs of the project managers. On the other hand, it aims to encourage project managers to integrate the use of appropriate simulation types in project planning and management processes. Simulation analysts would do well to be prepared for presenting alternate simulation types and their applicability to project managers who primarily think of only MCS as simulation and in some cases those who have no exposure to any types of simulation. Increasing accessibility and understanding of applicability of different types of simulation among project managers is expected to help increase its use.

This paper presents development of a SDS model based on a project plan and uncertainty information associated with activity durations. The results from SDS model are discussed. The SDS model results are compared with MCS and DES models and results for the exact same project case presented in Jain (2020). Paper length constraints prevent presentation of all the three models in this paper, but their results and learnings are presented in a comparative manner. This paper goes beyond other works in literature that compared DES and SDS models of a selected system in a few aspects. The earlier works have generally used small systems for such comparisons in different domains and the few that were in project management domain used rather simple project examples. Secondly, this paper includes MCS as one of the simulation alternatives that has not been included in prior comparisons. The inclusion of MCS is due to the selected domain of project management with a number of practitioners considering it as the primary simulation technique.

The rest of the paper is organized as follows. The next section briefly reviews relevant prior work. Section 3 describes the case project that is to be modeled for comparison. Section 4 describes the development of an SDS model of the project plan. Section 5 presents and discusses the outputs of the SDS model against corresponding DES and MCS results. Section 6 compares the learning from modeling using MCS, DES, and SDS. Section 7 concludes the paper.

2 RELATED WORK

The brief review here focuses on comparative papers that include SDS as one of the techniques. The readers are referred to Jain (2020) for comparative efforts focusing on MCS and DES. Also, for applications of SDS to construction project management, readers are referred to the review of over 100 relevant papers published over recent 24-year period by Xu and Zou (2021). While the focus is on construction, most of the discussion is applicable to project management in general, including the future research directions that are identified as integration of SDS with other methods, uncertainty analysis, and human factors analysis. Recently, there has been an increasing use of hybrid simulation that combines multiple simulation types such as SDS and DES. Winter Simulation Conferences have included a track on hybrid simulation from 2014 onwards and the associated conference proceedings provide a good source for reviewing recent developments in this area.

2.1 Comparative Efforts in Project Management Domain

A search of literature brought up only one publication that compared different types of simulation models in the project management domain. Suslov and Katalevsky (2019) present the approach for building a quite detailed SDS model for management of a simple software project followed by a presentation of ABS and DES models for the same software project. The models primarily use a single activity with code development rate as the means to mimic progress. A project network is not modeled. The SDS and DES models are shown to have close results while it is stated that ABS models provide similar results. Brief case studies are provided for application of ABS and DES for logistics operations in support of construction projects, but not of the overall project execution itself. The authors point out that advice from the Internet is to use ABS, SDS, and DES respectively when there are many independent objects, when only global level information is available, and when the system is easily described as a process. Hybrid models are recommended for better fit with different parts of the system. Also, it is recommended that model development efforts should include both simulation and industry practitioners for better representation.

The effort reported in this paper starts with a project using its network representation and thus demonstrates a generic approach that can be applied to projects in any industry. Also, project management practitioners are generally comfortable with the project network representation and it is anticipated that they will find it more relatable.

2.2 Comparative Efforts in Other Domains

There have been multiple efforts comparing DES and SDS techniques with many of them concluding that DES is better for modeling tactical and operational issues while SDS is better for modeling strategic issues (see for example, Sweetser (1999)). On the other hand, Morecroft and Robinson (2005) suggest that it is difficult to directly compare the two techniques since they provide different perspectives and study different aspects of a system.

A few of the comparison papers have modeled the same system or some overlapping concepts using SDS and DES and fewer still compare the results of models of the same system. For example, Tako and Robinson (2009) compared SDS and DES models of prison population change over the years. The outputs of the two models were shown to different batches of MBA students representing managers' perspectives. The students found the results useful without significantly discriminating between the software and types of simulation models used to generate them. The authors did point out that the case study was amenable to modeling by both SDS and DES and suggested using different case studies and a range of modeling techniques and software in future. The effort reported in this paper uses a project as a system to be modeled and thus provides a different case study. The amenability to modeling by different types of simulation was not a priori assessed. Also, any previous efforts comparing models of the same project network could not be located and this paper thus fills that gap.

There also have been multiple efforts reporting hybrid models that combine multiple types of simulations, such as, SDS and DES, or SDS and ABS, etc. Morgan et al. (2017) review a number of methods for combining the SDS and DES and present a tool kit for mixing them based on the project characteristics. In a recently reported effort using hybrid models, Al-Hawari et al (2020) state that a combined DES/SDS model provides better results than only DES or only SDS models of the same system. They compare two DES models with a combined DES/SDS model of a supply chain. The two DES models implement the variable changes differently, the pre-delay model updates the variables before a processing delay while the post-delay model updates after. The results show that the combined model provides "superior" results that generally fall in between the corresponding results from pre and post delay DES models. There have been hybrid models reported particularly in construction project management domain in research literature. However, in the author's opinion the project management practitioner community would need some time to appreciate and embrace the use of different simulation types individually before they start demanding the hybrid models. The effort reported in this paper is aimed at practitioner project manager to help them appreciate the modeling perspective and the potential learning from different types of simulations. It is also aimed at encouraging simulation practitioners to engage with project management community for improvement of project planning and execution and in turn the global economy.

3 CASE PROJECT

A hypothetical project has been used as the basis for comparison of different types of simulation. The particular project was modeled previously using MCS and DES (Jain 2020). The project is briefly described here for readers' convenience. Readers are encouraged to see Jain (2020) for more details on the project and associated objectives.

3.1 Hypothetical Project Description

The hypothetical project used for this study has 31 activities each with stochastic durations with means ranging from 4 to 120 days and with different specified distributions including Normal, Triangular, Beta and Uniform. Each activity may use 1 to 5 units each of 3 to 5 resource types, R1 through R5, with costs ranging from $10 to $50 per hour. The network for the hypothetical project with the critical path activities identified using a vertical lined pattern is shown in Figure 1. The critical path has been identified based on activity durations at their mean times and with no resource constraints.

The project plan was represented using the project planning software, MS Project, initially with no resource constraints. The resulting plan showed resource requirements ranging from a peak of 12 units for R5 to 23 units for R2, and duration of 228 days with the activity durations set at their mean values. The project duration increases to 370 days with the five resource types constrained at 10 units each and using the leveling feature of MS Project. Other project planning software and resource constrained project scheduling problem heuristics may arrive at different solutions with a duration more or less than 370 days.

3.2 Project Manager's Objectives

Objectives for a project can emphasize time, budget, scope, or more usually some attempt at a balanced combination of two or all three of these. For the purpose of this study, it is assumed that project manager's objectives are to complete the project in 350 days with some budget flexibility available to add resources if needed to meet the target project completion time (PCT). As is generally the case, it is assumed that the project manager (PM) will seek a project plan that provides a high probability of finishing the project within the committed PCT. In the interest of keeping this comparison effort manageable, no additional risk factors beyond the variability in activity durations are considered.

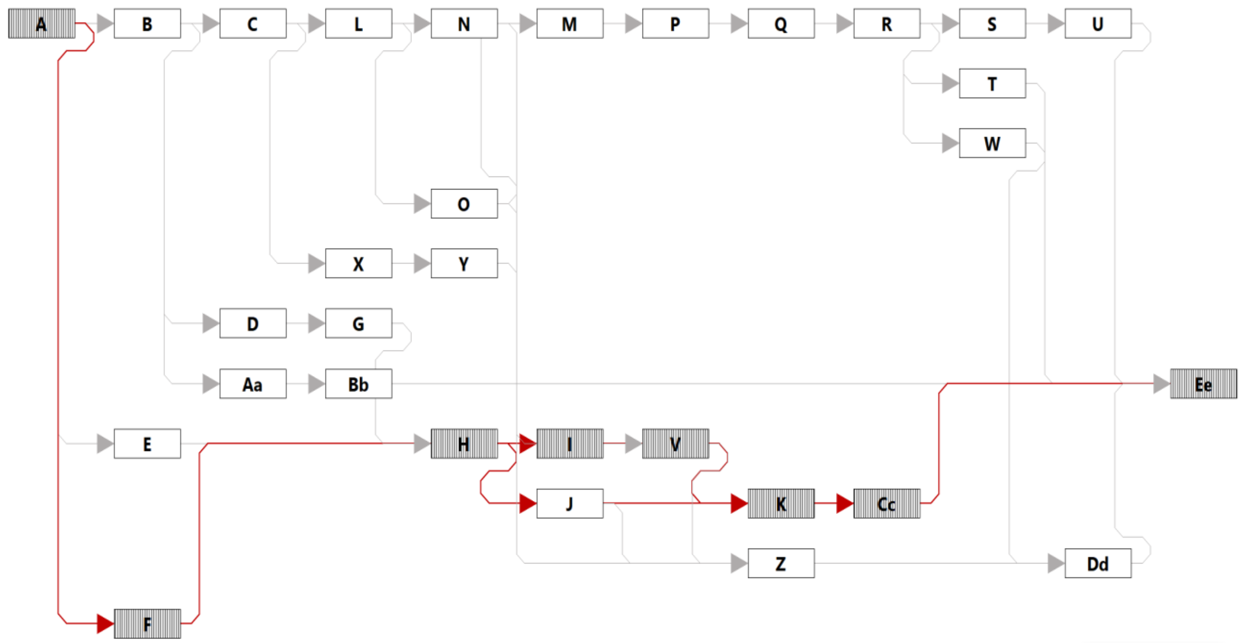


Figure 1: Network for the hypothetical project (activities with vertical line pattern are on the critical path).

4 SYSTEM DYNAMICS SIMULATION MODEL DEVELOPMENT

This section provides an overview of approach used to create the SDS model. It is recommended that simulation practitioners clearly explain the approach in such efforts in the interest of understanding of the model by project management practitioners. The approach discussed here is also intended to help other researchers for duplicating the results provided here.

SDS model may be considered as representing the underlying causes of variabilities that are input as numerical data in MCS and DES models. For example, MCS and DES models of project management represent variability for each activity's duration generally using random distributions. The random distributions may have been developed by collecting data on the activity's duration over multiple instances in previous similar projects or based on input of subject matter experts. MCS and DES use the random distribution for an activity as an input in the model and sample a duration from the distribution for each run representing an instance of execution of the project. The duration may have varied in practice from the defined standard due to factors such as quality issues leading to rework and lower productivity of the assigned team members due to fatigue or morale issues. SDS model represents such factors and captures their impact on the duration. While the representation can be done at each individual activity level, the recommended approach is to represent the aggregate impact at project level. It is tempting to digress into a discussion of aleatory versus epistemic uncertainty and their representation in different types of simulation, but that discussion could be an entire paper in itself and hence avoided here.

4.1 Definition of the Scope

The scope of a model has to be clearly defined to match the study's objectives. It is easy to have scope creep in all modeling efforts, but it is perhaps even more so in SDS due to the ability to model factors that may be considered soft or intangible such as team morale. It is easy to get tempted to add multiple soft factors and associated causal feedback loops in the model. Again, like most modeling effort beyond a threshold there are diminishing returns on the efforts to add more accuracy to the model.

A good comparison across different models requires that the scope should match closely if an exact match is not possible. Both MCS and DES evaluate the impact of stochasticity on system outputs of interest. SDS models do not incorporate stochasticity in the same manner. Indeed, individual runs of SDS are deterministic. One would need to vary the factors across multiple runs of SDS to evaluate the impact of stochasticity. DES outputs are used here as a reference for SDS modeling effort. The MCS outputs were found to have wider variation range compared to DES output even though both used the same input distributions for activity durations. Jain (2020) pointed to software implementation and some modeling limitations in the specific MCS software used as contributing factors for the differences in its outputs compared to DES.

A few factors were considered for inclusion in the SDS model that may generate the kind of variation in system output as realized in the DES model reported in Jain (2020). The exercise would demonstrate the ability of SDS model to generate the range with each output of interest. The factors considered include quality, productivity, schedule pressure, fatigue, etc. It was decided to primarily use quality variation and related factors, such as time to detect errors, for generating the variation in the output measures. This will avoid making the SDS model too complicated and facilitate its understanding by project practitioners and simulation analysts who generally use MCS and DES but not SDS.

Ideally, for developing a SDS model of a project execution one would study the project environment and identify the major causal feedback loops that may affect its performance. The identified loops should be included in the model and the model should be verified and validated using data collected from real projects. However, given that a hypothetical project is being used as a basis for comparison, it is assumed that the associated project system has quality variations as the main determinant of the performance.

The model should include the progression of the project with the network and associated activity information summarized in Section 3.1. Based on the preceding discussion, the scope includes modeling of the quality level and its impact on project progress.

4.2 Model Structure

Generally, SDS models represent project execution at an aggregate level with progress defined as the work flow, that is, the rate of work executed per unit time. The project dynamics tutorial in the modeling guide provided in the help documentation with Vensim software (Ventana Systems, Inc. 2019) uses this approach. The base model structure for this effort has been developed following the Vensim project dynamics tutorial. SDS models utilize stocks and flows to model system phenomena. The project progress is modeled as flow of work at *work-flow* rate from *work-remaining* stock to *work-completed* stock. Some of the work suffers from poor quality and requires rework. This is modeled as a diversion of some of the work flow at *(1-quality)* to *undiscovered-rework* stock. The need for rework is discovered at *rework-discovery-rate* and the discovered rework gets added to the *work-remaining* stock. The *time-to-detect-errors* decreases as the project nears completion, indicated by increasing value of *fraction-complete* variable. The change in *time-to-detect-errors* is implemented in the model using *time-to-detect-error-lookup* function. A few variables are used for tracking output measures including *total-work-input*, *total-cost*, and *project-completion-time*. The SDS model for the project is shown in Figure 2.

Modeling a project plan execution would generally require representation of the precedence network and resource constraints. The work-flow rate mechanism used in the Vensim tutorial model would work well when the project work flow does not vary widely across the duration of the project. Similarly, aggregate representation of entire workforce as done in Vensim tutorial model would work well when the resources are similar and their usage doesn't vary widely across different activities. This effort used different approaches than in the Vensim tutorial model to represent the effect of the precedence network and resource constraints. These approaches are discussed in the following subsections.

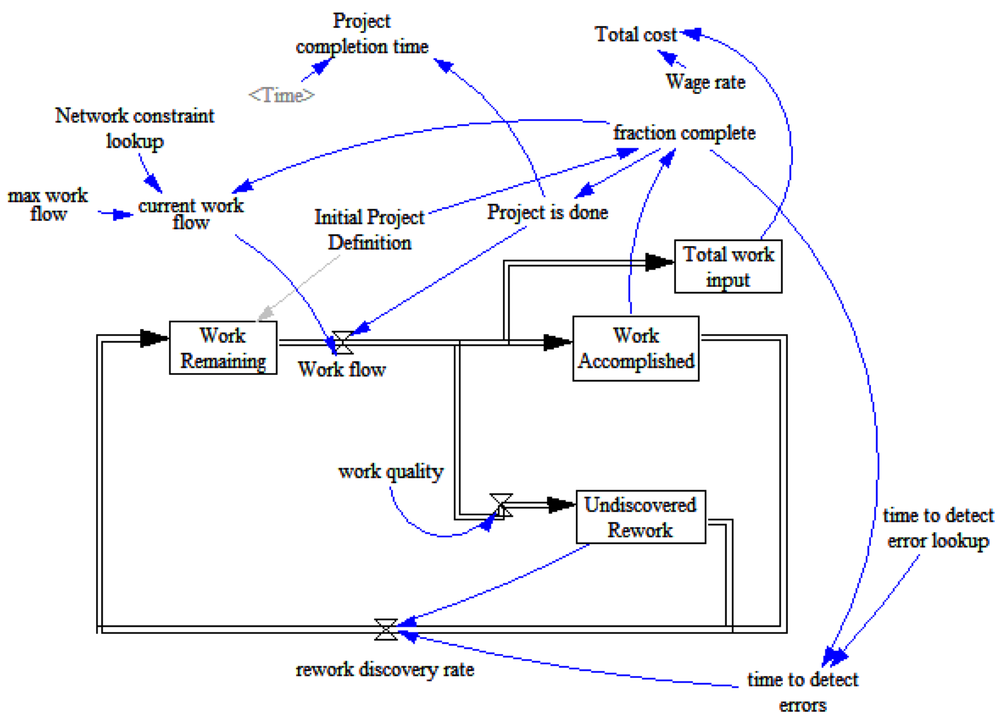


Figure 2: SDS model of the hypothetical project.

4.3 Modeling the Precedence Network

The DES and MCS models explicitly model the precedence relationships of the project network to model the project progress. The total work for the project based on the most likely values for activity durations is 9271 person-days that is executed over 228 days of working time based on the network with deterministic times and unconstrained resources. A possible approach to model unconstrained execution could be to use a work flow of (9271/228=) 40.6 workdays/day. This will generate the project duration of 228 days.

Modeling the impact of precedence relationship in a project network would need a more involved approach beyond the simple use of work flow. Project practitioners would point out that projects don't execute at a constant work rate. Project execution generally follows an S-curve with work starting slowly in the beginning executing at a faster rate in the middle and then ending with a slower rate. Project networks have one to a few activities starting in parallel before multiple activities can be executed in parallel to achieve a fast execution rate. The network generally ends with only a few activities running in parallel. The pace of execution in middle would also vary based on the number of activities running in parallel and their respective resource requirements. In this effort, a novel approach exploiting the rate of project execution is used to represent the impact of precedence relationships. The rate of execution was captured based on the deterministic execution and used in the SDS model based on project progress, captured in the model by the *fraction-complete* variable. The rate of execution at any point is defined as a fraction of the maximum work flow across the project. The maximum work flow was calculated as 1486 person-days/month across the 11-month duration of the project. Figure 3(a) shows the variation in work flow rate as a fraction of maximum work flow across the duration of project represented as fraction complete of the

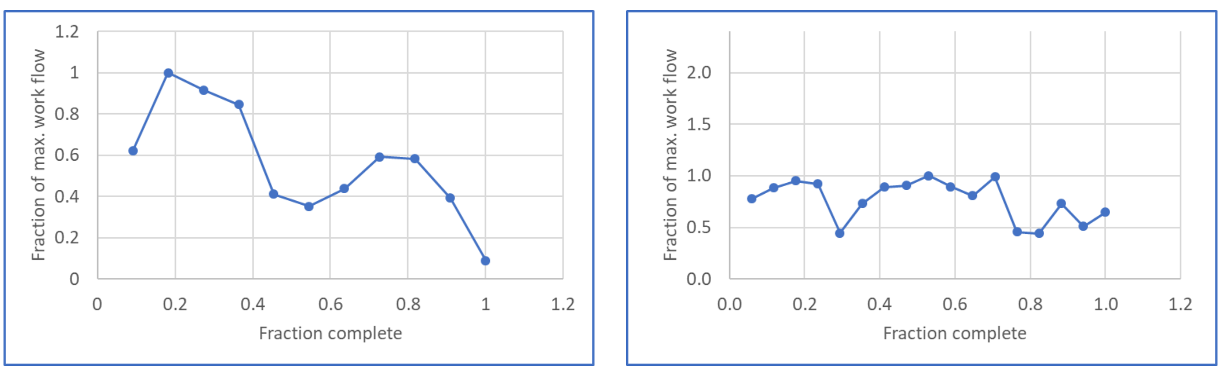
11-month duration. The variation shown in Figure 3(a) is coded in the model as the *Network-constraint-lookup* variable that returns the applicable work flow given the fraction complete. The applicable work flow for any day is determined as:

$$current\text{-}work\text{-}flow = max\text{-}work\text{-}flow * Network\text{-}constraint\text{-}lookup(fraction\text{-}complete). \quad (1)$$

4.4 Modeling Resource Constraints

SDS models represent resources at an aggregate level similar to the aggregate representation used for project progress. SDS models generally represent the total workforce and track hiring, firing, productivity, morale, etc. and their impact on the project progress represented by the work flow. The project plan used for this study calls for varying numbers of 5 types of resources for various activities. Using MS Project for project planning, the project duration expands from 228 days in unconstrained setting to 370 days with resources constrained at 10 units for each of the 5 resource types. It will take a rather complex and involved SDS model to represent individual activities and their respective resource requirements. More importantly, the gain in accuracy in aggregate output measures may not justify the effort for such a detailed representation.

The impact of resource constraints was represented by again exploiting the project execution rate over the project duration. The activity execution sequences change due to resource constraints and that is reflected in the monthly work flow. The variation in work flow across the project duration represents the combined impact of resource constraints and the precedence relationships. The maximum work flow with constrained resources dropped down to about half the work flow of unconstrained setting to 714 person-days/month across the expanded project duration of 17 months. Figure 3(b) shows the variation in the flow rate as a fraction of the maximum work flow rate that is coded as the *Network-constraint-lookup* for representing the constrained resource setting.



(a) Unconstrained resources. (b) Constrained resources.

Figure 3: Variation in work flow rate across the duration of project. The vertical axes of the figure on the right has been purposely adjusted to indicate that the max. work flow for constrained resource setting is about half that of the unconstrained setting.

5 SIMULATION OUTPUTS

Selected outputs of the SDS model of the hypothetical project are presented in this section against corresponding results of the DES and MCS models of the same project extracted from Jain (2020). This being a hypothetical project, it was not possible to study previous similar projects and the variations in included factors such as quality and time to detect errors. The SDS model was verified for the simple case of deterministic execution and then calibrated against the case of unconstrained stochastic execution of DES model results. The calibrated SDS model is then used to make runs for the same base and crashing cases used in Jain (2020) for DES and MCS model comparison.

5.1 Verification and Calibration of SDS Model

The first step to verify the initial SDS model was to ensure that the project will complete in the same duration as determined in MS Project runs in unconstrained resource setting. The work flow rate was determined as total person-days divided by the project duration in MS Project. This work flow rate was used in the SDS model with 100% quality to check that it generated the same duration. This exercise was carried out for two cases, first with all durations set to their optimistic values, and then with durations set to mean values. While this appears to be a simple calculation check done across two different software, it provides a basic level of verification.

Next the SDS model was calibrated against the DES model results for unconstrained resource setting. The variations in SDS output measures are generated by varying the quality input. For example, quality level of 70% (0.7) results in higher rework and thus large total work input, long duration, and high cost that are close to corresponding values generated by DES for the 95-percentile case. Similarly, quality levels of 77% and 81% generate values that are close to mean and 5 percentile case in DES outputs. Table 1 presents the calibrated results of SDS model for the unconstrained resource setting with stochastic activity durations modeled in DES and MCS models. For the sake of brevity, only the duration results are presented. The calibrated quality values are used in other cases presented in Sections 5.2 and 5.3 below to approximate the 5-percentile, mean, and 95-percentile values.

Table 1: Calibrated outputs of SDS model with **unconstrained** resources and stochastic durations.

Simulation Paradigm	Software Used	Project Duration in Working Days		
		5-percentile	Mean	95-percentile
MCS	@Risk	215.7	232.2	250.3
DES	ExtendSim	215.4	232.0	250.6
SDS* (quality value)	Vensim	216 (81%)	228 (77%)	252 (70%)

*: SDS values are based on single deterministic runs for each of the 3 cases generated based on the selected quality level.

5.2 Base Case Results

The base case represents the realistic case of constrained resources. Each of the 5 resource types is limited to 10 units each. The resource constraints are modeled in SDS model using the approach described in section 4.4. Table 2 presents the results for the base case of constrained resources with stochastic activity durations for DES and MCS models.

Table 2: Results of base case with **constrained** resources and stochastic activity durations.

Simulation Paradigm	Software Used	Project Duration in Working Days		
		5-percentile	Mean	95-percentile
MCS	@Risk	371.5	394.8	417.8
DES	ExtendSim	331.8	350.6	366.0
SDS* (quality value)	Vensim	336 (81%)	353 (77%)	389 (70%)

*: Same comment as in Table 1.

The SDS output values are generated using deterministic runs with quality levels set to 70%, 77%, and 81% to estimate the 95-percentile, mean, and 5-percentile values as earlier. While in the previous case the values were calibrated to match, it is interesting to note that using the same quality levels for the base case the SDS outputs match the DES outputs quite closely for the 5-percentile and mean project durations. The

SDS model duration in 95-percentile case shows an increase of 6% compared to the DES output. The difference is quite low given the aggregate representation used in SDS model including the approaches used for modeling the effect of precedence network and resource constraints.

5.3 Crashing with Additional Resources

The next case explores representing crashing with additional resources if the project is running behind schedule. Specifically, the project manager is given the target of finishing the project in 350 days. If the project is running behind at the half duration point of 175 days, 2 units each of the 5 resource types are brought on to the project team. The conditional logic was represented in DES model but it was not possible in MCS due to software limitations (Jain 2020). In the SDS model, with each run being deterministic, the status can be determined at the 175-day point. Similar to the option taken for the MCS model in Jain (2020), the increase in resources is implemented after 175 days in the SDS model regardless of the project status. In the SDS model the increase in the resources is represented via an increase in the *max-flow-rate* used to determine *current-work-flow* per equation (1) provided in Section 4.3. The increase in the max flow rate is calculated as 20% of the average work flow rate to estimate the impact of increase of 20% of resources (2 units added to 10 units for each resource type). The average work flow rate is determined for the constrained case using MS Project. Table 3 presents the results for the crashing scenario.

Table 3: Results of the case of crashing with additional resources.

Simulation Paradigm	Software Used	Project Duration in Working Days		
		5-percentile	Mean	95-percentile
MCS	@Risk	348.9	369.6	392.4
DES	ExtendSim	317.7	335.0	352.7
SDS* (quality value)	Vensim	316 (81%)	333 (77%)	363 (70%)

*: Same comment as in Table 1.

The project durations generated by SDS model are again quite close to those generated by the DES models for all the three instances. The largest difference is about 3% in the 95-percentile case. The calibration approach works quite well for representing the selected cases in SDS model.

6 DISCUSSION

The approach used for comparison of SDS, DES, and MCS has some limitations but it provides a good idea about their respective capabilities for the selected scenario. The use of a hypothetical project led to calibrating the SDS model based on DES model results. If a real project were being modeled and there was a good database of similar past projects, the SDS model wouldn't need to be calibrated using the DES model. Information on variations in the represented factors in the SDS model, such as quality levels and time to detect errors would be extracted from the database and set up in the SDS model. Similarly, information on factors represented in the DES model, such as variability in durations of activities, would be used to set up that model. Since the subject phenomena being modeled is the same, factors in different modeling perspectives are interlinked. SDS model is representing the underlying causal factors that generate the numerical variations that are represented in DES and MCS models. The use of calibration approach to approximate the values of causal factors in SDS model is hence a reasonable approach in the absence of access to a real project.

The comparison exercise indicates that while all three types of simulations can be used to model the hypothetical project, the outputs can be different based on the scenario. The three of them provide closely matching results for unconstrained resource scenario. However, there are differences in constrained resource scenarios. The difference in the outputs of MCS and DES can be largely attributed to difference in heuristics for developing the solution to the underlying resource constrained project scheduling problem. The ability or lack thereof to model addition of resources based on project status also contributed to differences in the output durations for the crashing scenario. Table 4 summarizes the comparison of the three types of simulations included in this study in the context of project plan evaluation.

Table 4: Comparison of the three simulation paradigms in project management context.

Comparison Aspect	**Simulation Paradigm**		
	MCS	**DES**	**SDS**
Expertise requirement for a project practitioner	Low, should be able to use themselves	High, would generally require access to a simulation analyst	High, would generally require access to a simulation analyst
Required project management maturity	Low	Medium	High
Recommended purpose	Estimate project performance measures	Estimate project performance measures generally with greater detail than MCS or SDS	Understand the underlying causes of performance variations and estimate aggregate level measures
Project plan representation	Available software usually interfaces with project planning software such as MS Project	Generally requires modeling the network from scratch	Network model of project generally not used. Uses aggregate representation of work flow or task completion rates.
Resource constraints representation	Requires special software routines and long run times	Represented well at detailed level	Represented at aggregate level generally
Stochasticity	Native capability using random distributions	Native capability using random distributions	Requires additional effort to set up
Conditional policies based on random events (such as crashing if project delayed)	Not possible	Native capability	Generally not possible
Project performance measures estimates	Distribution generated, constrained resource scenario impacted by heuristics in planning software	Distribution generated, constrained resource scenario impacted by heuristics selected by analyst for resource assignments	Generally provides deterministic values at aggregate levels

It should be noted that above assessments are based on the specific software used for the three types of simulations. It is possible that the assessments may change somewhat if a different set of software were used.

7 CONCLUSION

This effort set out to help project practitioners understand the applicability of different types of simulation for project planning. Simulation can be applied in a number of aspects of project management. This effort focused on the primary application area of evaluation of the impact of variability on project performance measures. The current applications by practitioners generally involve use of MCS. The intent of this effort is to have practitioners develop understanding of other types of simulation, in particular, DES and SDS and utilize them to improve project planning specifically and project management in general. Another motivation was to have simulation analysts see the opportunities of applying different types of simulation to support improvements in project management practice. It is believed that the description of modeling approaches, respective results and comparison discussions provided in this paper and in Jain (2020) have together made substantial progress towards the stated objectives. Interested readers are encouraged to contact the author for the model files particularly if they share the goals of improving project management practice via use of simulation.

A significant number of publications are available that compare multiple types of simulation and in particular SDS and DES. The contribution of this paper is in providing the comparison in the context of project planning applications of different types of simulation. This effort also contributed approaches for modeling that may be useful or simulation analysts. Specifically, the approach for implementation of resource constrained crashing scenarios for MCS reported in Jain (2020), and the approaches to represent the impact of precedence relationships and resource constraints in SDS models were developed as part of this effort.

Future work may explore inclusion of agent-based simulation as the fourth type of simulation for comparison. Teaching materials may be developed using the models and results from this effort for inclusion in classes in a graduate level program in project management.

REFERENCES

Al-Hawari, T., A. Gailan Qasem, H. Smadi, H. Arauydah, and N. Al Theeb. 2020. "The Effects of Pre or Post Delay Variable Changes in Discrete Event Simulation and Combined DES/System Dynamics Approaches in Modelling Supply Chain Performance". *Journal of Simulation* 16(2):147-165.

Bakhshi, J., V. Ireland, and A. Gorod. 2016. "Clarifying The Project Complexity Construct: Past, Present and Future". *International Journal of Project Management* 34(7):1199-1213.

Jain, S. 2020. "A Tale of Two Simulations". In *Proceedings of the 2020 Winter Simulation Conference*, edited by K.-H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing, 2470–2481. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Khamooshi, H., and D.F. Cioffi. 2013. "Uncertainty in Task Duration and Cost Estimates: Fusion of Probabilistic Forecasts and Deterministic Scheduling". *Journal of Construction Engineering and Management* 139(5):488-497.

Morecroft, J., and S. Robinson. 2005. "Explaining Puzzling Dynamics: Comparing the Use of System Dynamics and Discrete-Event Simulation". In *Proceedings of the 23rd International Conference of the System Dynamics Society*. July 17th-21st, Boston, MA, USA, 17–21.

Morgan, J.S., S. Howick, and V. Belton. 2017. "A Toolkit of Designs for Mixing Discrete Event Simulation and System Dynamics". *European Journal of Operational Research* 257(3):907-918.

Suslov, S., and D. Katalevsky. 2019. "Modeling and Simulation Toolset". In *Evolving Toolbox for Complex Project Management*, edited by A. Gorod, L. Hallo, V. Ireland, and I. Gunawan, 417-450. Boca Raton, FL: CRC Press.

Sweetser, A. 1999. "A Comparison of System Dynamics (SD) and Discrete Event Simulation (DES)". In *Proceedings of the 17th International Conference of the System Dynamics Society*. July 20th -23rd, Wellington, New Zealand, 1-8.

Tako, A., and S. Robinson. 2009. "Comparing Discrete-Event Simulation and System Dynamics: Users Perceptions". *Journal of the Operational Research Society* 60(3):296-312.

Ventana Systems, Inc. 2019. *Vensim PLE for Windows Version 8.2.1.*

Xu, X., and P.-X. Zou. 2021. "System Dynamics Analytical Modeling Approach for Construction Project Management Research: A Critical Review and Future Directions". *Frontiers of Engineering Management* 8(1):17-31.

AUTHOR BIOGRAPHY

SANJAY JAIN is an Associate Industry Professor in the Department of Decision Sciences, School of Business at the George Washington University. Before moving to academia, he accumulated over a dozen years of industrial R&D and consulting experience working at Accenture in Reston, VA, USA, Singapore Institute of Manufacturing Technology, Singapore and General Motors North American Operations Technical Center in Warren, MI, USA. His research interests are in application of modeling and simulation of complex scenarios including supply chains, smart manufacturing systems and project management. His email address is jain@email.gwu.edu.