**INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09**
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

# STATISTICAL ANALYSIS OF PROCESS SIMULATIONS

**Arie Karniel, Yoram Reich**
Tel Aviv University

## ABSTRACT
Simulations of New Product Development (NPD) processes using their specific contexts can provide project managers with decisions-making aids. The NPD context, which incorporates knowledge about the product, requirements, technology, and other factors, is dynamically evolving during the process. The Design Structure Matrix (DSM) could be used to model this product knowledge and reordering algorithms could be used for process planning. The plan should be updated as the product knowledge evolves; however, the transition from DSM-based plan to process scheme implementation is not unique. The process can then be used for simulating process related measures that can guide the decision making regarding the preferable implementation strategy. However, such decisions should reflect the statistical confidence of the simulation results. In the current work, the properties of multiple repetitive simulations are used for supporting decision-making based on statistical confidence interval derived by evaluating the significance of the difference between the results of applying different DSM-based plan implementation options (defined as business rules). The same approach is applicable to similar decision-making related to processes in general.

*Keywords: New Product Development (NPD), Design Structure Matrix (DSM), Product Design, Simulation, Business rules, Statistics*

## 1 INTRODUCTION
Product development processes (PDP), and primarily New Product Development (NPD) processes are unique and highly complex [1]; involve multiple disciplines contributing to the development of complex multidisciplinary products (e.g., machatronics); have limited resources, shortened development time [2], and increased quality and regulatory demands [3]. They are dynamic, because they evolve and change due to various reasons including market, technology, and organizational changes. Consequently, the scope of work needed for a new product cannot be define nor could be *a-priori* planned [4]. Being iterative processes, PDP planning and simulation should cope with iterations, which are not modeled well by the commonly used GANTT and PERT charts [4]. In order to simulate such processes, where the plan keeps changing to changes in the product knowledge, the recurrent plan translation to a process scheme should be explicit [5], and correct [6].
The Design Structure Matrix (DSM) [7], is a square matrix utilized to capture product knowledge, including activities interdependencies and process iterations (feedback loops). The DSM is further used for planning the process. The typical DSM planning is based on DSM reordering algorithms that try to minimize the number of iterations. Minimum iterations marks are expected to yield optimal processes [7]-[9]. Ideal sequencing without feedback marks (acyclic process) is unlikely to exist in development processes [10]. Feedback marks in an ordered DSM create loop of coupled activities. Reordering of coupled activities within a loop requires additional reordering algorithms [6], [11].
The common DSM method typically addresses the minimum iteration marks objective, and is criticized for not addressing other process measures such as project duration or cost. Taking such measures into account may change the decision regarding the appropriate process plan [12]-[13]. Alternately, given the process plan, simulation techniques are required for analyzing process objectives, regarding time and cost, or additional process data such as: rework effort [12], risk propagation [14], communication time [15], or handling process data variations such as uncertainty and learning [16].
The implementation of a DSM plan into an executable process scheme is not unique [17]. Different implementation options, defined as business rules, can apply to different business cases (e.g., the

availability of resources or deadline considerations). Simulation results can guide decision making towards optimal selection between applicable business rules. Simulations typically utilize random generated values for the measures being considered; therefore, appropriate statistical methods are required for decision-making.
The article describes the use of parameters in DSM-based simulations, and reviews the use of statistical considerations in DSM-based simulations. In the DSM literature, the simulations results are used for analyzing the DSM-based planning by comparing mean and standard deviation, but without checking the results significance. Moreover, the DSM implementation is assumed unique and decisions are not applied to selecting between implementation options.
The current article describes a different approach. The focus is the choice of the process implementation of DSM-based plans (being an additional factor besides choosing between plans). Additionally, the statistical approach described utilizes the analysis of the expected value of a difference function between simulation results of two potential implementation options. Calculations of variance and confidence interval are then used for decision-making. The presented approach is useful for statistical analysis of simulation results in general.

## 2 LITERATURE REVIEW

The scheduling literature addresses the need to satisfy desired project quality requirements such as minimum time, resources, or other objective functions [18]. The activity relations regarded may include traditional precedence constraints (finish-start) [18], or Generalized Precedence Relations (GPR) [19]. The main problems addressed in the scheduling literature are Resource-constrained Project Scheduling Problem (RCPSP), and uncertainty in activities duration (with or without resource constraints)[18]-[21]
Proactive scheduling that accounts for statistical knowledge of uncertainty focuses on the schedule robustness to changes, (e.g., by buffers). Reactive scheduling involves revising the schedule when unexpected events occur, typically resources and time variations, but also adding a new activity [22][23]. However, the scheduling of iterative activities is typically not considered [21].
Iterations could be modeled by DSM, yet the basic DSM model typically does not include the activity duration, duration changes due to iteration (learning), and the impact of iterations or rework. Therefore, the information presented in DSM is partial and was criticized as insufficient for process plan evaluation. The basic algorithms used for process planning are based on minimizing iterations (feedback) marks. The expectation of minimum iteration marks to yield optimal processes in terms of total time and robustness to activity duration variations [3], [7],[3][24]is an appealing general planning heuristic. However, the counter example simulation results presented in [12]-[13] contradicted that basic assumption by demonstrating that shortest process time required more iterations than minimal.
In [12] process duration and process cost were estimated, using iterations with overlap. Increasing the concurrency of the activities increased the number of iteration marks, and the number of iterations. Thus, cost has increased, but due to overlapping only part of the activity had to be executed again, and the overall duration decreased.

| | A | B | C | D |
|---|---|---|---|---|
| A | | | | 2 |
| B | 1 | | | |
| C | | 1 | | |
| D | | | 1 | |

(a) minimum iterations

| | D | C | B | A |
|---|---|---|---|---|
| D | | 1 | | |
| C | | | 1 | |
| B | | | | 1 |
| A | 2 | | | |

(b) minimum time

*Figure 1. The impact of minimum iterations marking*

The example in [13] demonstrated a case where more feedback links yielded a shorter process. The marking indicates the number of repetitions; thus, a reordering with more feedback markings, where the span of iterated activities was smaller, has created an overall shorter process time. A simplified example that demonstrates the essence of the approach in [13] is depicted in Figure 1. In (a), there is one marking that indicates two iterations of the whole process, i.e., each activity is executed three times. Assuming equal activities duration $X$ for all activities we get an overall process time of

*T=12*X*. A reordering of the process is presented in (b). The first activity *D* executes, then *C*, then due to iteration *D* and *C* are executed again, then *B* executes, iteration of *B* and *C*, then *A*, and finally iteration of *B* and *A*. Overall, *D* and *A* have executed twice; *B* and *C* have executed three times; and the total process time is *T=10*X* .

It should be noted that the minimum iteration marks concept was developed for a binary DSM, where all marks are equal, while the markings in the above examples have values (iteration probability values in the first case, and number of iterations in the latter case). In such cases, the rule of thumb of minimum iterations is not enough and simulations are needed to address the process specifications.

## 2.1 DSM-based simulation parameters

DSM-based simulations are used for DSM reordering, and calculations of process measures (e.g., duration). Most DSM-based simulations address duration issues using Monte Carlo sampling for the stochastic activities duration. Resource issues were addressed in [25]. Resource scheduling, based on DSM plan was applied to assignment of computer resources in real time calculations [26]; and to a weekly assignment of project resources in [27].

The typical DSM-based planning algorithms do not utilize the diagonal [28]. However, if there is only one simulation parameter involved per each activity, the diagonal cells could be used to present it. Such presentation was used in [29] for presenting the design effort of an activity. A summary of simulation parameters used in DSM-based simulations appears in Table 1. Simulation parameters might address activity properties, marked by (A); or relations between activities that are presented by additional matrices (M). The various optimization objectives yield broad and diverse range of parameters. Moreover, the range expands with the type of parameter changes. Some studies have stochastic parameters within a frame of a deterministic process [13], [26], [30]. Other studies have deterministic parameters within a changing process, where the process route is probabilistically selected [29], [31]; and some studies use both sources of process parameters change.

## 2.2 Objective function

Two of the studies in Table 1, are using a DSM-based plan with minimum iteration criterion. Once the plan is set, it is assumed that there are no feedback links, and the scheduling problem is solved. Coates *et al.* [26] maximized resource utilization (and minimized time) by resource scheduling according to the actual duration of tasks, given the predefined DSM-based plan. In [27], the duration and resource availability parameters are used for implementing activity weekly schedule according to the plan (considering the information needs, and resources availability). In both cases, simulation parameters are not used for further process plan optimization.

Browning and Eppinger [12] suggested calculating the risk factors for duration and cost, which are the integral of the impact of unwanted result, multiplied by the probability of such result (based on simulation). The impact is calculated as a square of the difference between unwanted outcome and required outcome when overrunning the scheduled due date, or the budget, respectively. Lévárdy and Browning [32] used a weighted project risk that includes duration, cost, and technical performance risks. The risk is evaluated in each process step and guides choosing the best task to be performed next. The simulated parameters in [29], are the DSM links according to decision regarding modularization or standardization of a product component. Simulation results are used to calculate the best process plans, and the overall results are then used for choosing the components that should be standardized or modularized. Minimum process duration is used in [16] [13]. The variance of the process duration is additionally considered in [16] and is the objective function in [33].

## 2.3 Statistical analysis for decision-making

Using Monte Carlo simulation, the simulated processes proceed by generating random numbers from probability density functions *f(x)*. Parameters of the resulting process may not have a formal distribution, but such distribution can be generated by multiple repetitions of the simulation. Asmussen and Glynn [34][1] indicated that according to the Law of Large Numbers (LLN), and the Central Limit Theorem (CLT), the distribution generated by the simulation converges to the actual distribution (that might be unknown, or cannot be expressed by a formula).

Discrete probability mass distribution of the process time can be generated for evaluating mean and

[1] Statistical definitions and equations are described in annex A.

standard deviation. In [16], probability mass distribution was used for comparing various model assumptions; in [13] for comparing various DSM rearrangement cases; and in [30] for comparing fluctuations strength. Browning and Eppinger [12] generated the distributions of cost and duration, and then used them to calculate the risk of overrun.

*Table 1. DSM-based simulations parameters mapping*

| Source | Optimization objective | Simulation parameters | Parameter type |
|---|---|---|---|
| Melo and Clarkson (2001) [31] | Minimum cost / risk | Path choice | Probabilistic path choice |
| Browning and Eppinger (2002) [12] | Minimum risk factor of duration and cost | Duration (A) Cost (A)<br>Learning Curve(A)<br>Rework probability (M)<br>Rework impact (M) | Stochastic: Triangular distribution of Duration and Cost (min, expected, max) |
| Coates *et al*. (2003) [26] | Maximum Resource utilization | Actual Duration (A);<br>Resource availability | Stochastic duration |
| Cho and Eppinger (2005) [16] | Minimum mean and variance of time | Duration (A)<br>Learning curve (A)<br>Overlap amount (M)<br>Overlap impact (M)<br>Rework probability (M)<br>Rework impact (M) | Stochastic: Triangular distribution of Duration (min, expected, max)<br>Learning curve (initial, %reduction, minimum) |
| Huberman and Wilkinson (2005) [30] | Minimum instability due to fluctuations | Work transformation fluctuations | Stochastic fluctuations |
| Lévárdy and Browning (2005) [32] | Minimum project risk | Duration (A)<br>Cost (A)<br>Performance (A) | All parameters are stochastically generated |
| Sered and Reich (2006)[29] | Minimum design effort | Effort (A)<br>DSM links values (M) | Link values per decision (standardization/ modularization)<br>Probabilistic path choice |
| Abdelsalam and Bao (2006)[13] | Minimum duration | Duration (A) | Stochastic duration |
| Yassine (2006) [33] | Minimum duration/cost variability | Duration (A)<br>Cost (A)<br>Task Volatility (M) | Stochastic: Triangular distribution of Duration (min, expected, max) |

One of the issues that need to be addressed is the number of simulations required for the result to converge. In many cases, a large number (assumed to be large enough) is taken: 800 in [13], 1000 in [16], and 10000 in [30][30]. Asmussen and Glynn [34] describe a typical approach of estimating the required number by a small size batch of simulations. Browning and Eppinger [12] defined criteria for estimating convergence of the distribution by running batches until the relative mean and deviation of the additional batch is below a threshold. However, such relative threshold has no statistical meaning.

Given the simulation results, a decision-making procedure should choose the best results according to the criterion. Since the results are distributed, statistical measures should be used to support the decisions, i.e., to check if the results are statistically significant. However, the DSM-based studies reviewed do not address that requirement, and typically just compare mean and standard deviations without checking their significance.

The expected value of a large number of simulation results has the properties of a normal distribution. If we split the simulation runs to $K$ sections of size $M$, the expected values taken from these sections have a $t$-distribution. Furthermore, a function over the expected value of simulation parameters has a normal distribution; and function estimations over expected values taken from several sections have a $t$-distribution. These properties are used [34] for estimating variance and confidence interval of the function[34].

In the current work, the properties of multiple repetitive simulations are used for supporting decision-making by statistically evaluating hypothesis regarding the differences between the results of applying different implementations of the DSM plan defined as business rules.

## 3. Simulating a dynamic scheme process

### *Applying design blocks at run time*

Business Rules indicate different business cases, and may indicate different strategies. Choosing the appropriate strategy might be done by using rules of thumb; e.g., if there are enough resources make the activities as much as parallel to reduce overall time. However, in the case of iterative processes, the “best” strategy may not be very clear.

Two business cases are depicted in Figure 2. The first case is a serial performance of coupled activities (i.e., one activity at a time), Figure 2(a); the second example is performing the activities as a *Design block* (DB) all activities start in parallel and complete together, in a case of iteration all activities are performed again. The DB has a self-iteration probability $p=0.36$, according to Equation 1 [35].

$$Pd = 1 - \Pi(1-Pi) \quad (1)$$

Where $Pd$ is the combined probability cell of a DB, and $Pi$ are the values of the merged probability cells. The probability cells, which are on forward direction (sub diagonal) to (from) design activities within the DB are all merged to the probability of forward link to (from) the merged DB. The probabilities within the DB are merged to the self-iteration probability of the DB.

The duration of the DB is defined as the maximal duration of its activities. The duration assigned to activity *X*, is defined as D*(X)*. The activity durations are: D(*A*)=1; D(*B*)=2; D(*C*)=3; D(*D*)=4. The duration of the DB D(*BC*)=max(D(*B*),D(*C*))=3.

| | A | B | C | D |
|---|---|---|---|---|
| A | | | | |
| B | 0.3 | | 0.2 | |
| C | | 0.2 | | |
| D | | | 0.1 | |

(a) Serialized process

| | A | BC | D |
|---|---|---|---|
| A | | | |
| BC | 0.3 | 0.36 | |
| D | | 0.1 | |

(b) Design block (parallel)

*Figure 2: Serialized and Parallel process execution*

The distribution of the overall process duration (for 100 runs) is depicted in Figure 3, indicating duration results of the serialized process (P1) and process with DB (P2). The X axis is the overall process duration; the Y axis is the number (and %) of runs with the indicated duration. The distribution has discrete values and a decreasing shape (due to decreasing probability of iterations). It is apparent that the distribution is right (positive) skewed, and not a normal distribution.

In order to analyze the results, several decision statistic parameters were compared: average duration [16], median (not mentioned in DSM literature), and paired comparison [36].

The results for average and standard deviation of averages, and average and standard deviation of medians are depicted in Table 2Table . Results are presented for a range of values for the parameters: probability P(*BC*); probability of iteration probability P(*C*B); duration D(*B*) and D(*C*). The appropriate parameters for the parallel process (DB) are calculated as in the above example.

The results are averaged of 10,000 run-time simulations. The average, median, and their respective standard deviations were calculated from 100 averages (medians) of 100 runs, for each set of parameters.

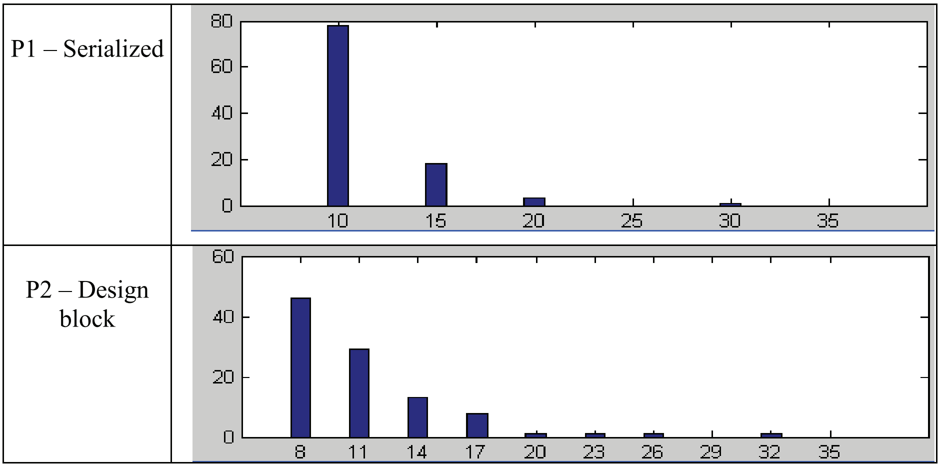


*Figure 3: Processes duration distribution*

For the range of parameters checked, the decisions based on simple comparison of the different statistical parameters (average, median, differences) resembled, but were not similar. The differences of average and median were not statistically significant. The decisions, according to each of the statistical measures are marked by colors: green indicates that process P1 was preferred, and pink indicates preference of the DB P2 process. Having enough samples should finally create a normal curve of the averages, allowing comparing averages with statistical significance.

*Table 2. Business Rules comparison*

| | Serial | | | | Design Block | | Serial (avg) | | Db (avg) | | Serial (med) | | Db(med) | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| # | P (B-C) | P (C-B) | D(B) | D(C ) | Self P (BC) | D (BC) | Avg-avg (P1) | Std – avg (P1) | Avg – avg (P2) | Std-avg (P2) | Avg-med (P1) | Std – med (P1) | Avg-med (P2) | Std – med (P2) |
| 1 | 0.6 | 0.6 | 1 | 2 | 0.84 | 2 | 12.39 | 0.55 | 17.40 | 0.97 | 10.91 | 0.56 | 14.10 | 1.09 |
| 2 | 0.6 | 0.6 | 1 | 5 | 0.84 | 5 | 20.10 | 1.14 | 35.36 | 2.83 | 16.82 | 1.11 | 27.70 | 3.21 |
| 3 | 0.6 | 0.6 | 4 | 5 | 0.84 | 5 | 27.49 | 1.82 | 35.91 | 2.81 | 22.68 | 1.95 | 27.60 | 3.31 |
| 4 | 0.4 | 0.4 | 1 | 2 | 0.64 | 2 | 9.99 | 0.31 | 10.56 | 0.44 | 8.10 | 0.53 | 9.07 | 0.32 |
| 5 | 0.4 | 0.4 | 1 | 5 | 0.64 | 5 | 14.96 | 0.61 | 18.93 | 1.21 | 11.12 | 0.84 | 15.32 | 1.16 |
| 6 | 0.4 | 0.4 | 4 | 5 | 0.64 | 5 | 20.05 | 0.89 | 19.06 | 1.05 | 14.05 | 0.45 | 15.12 | 0.74 |
| 7 | 0.2 | 0.2 | 1 | 2 | 0.36 | 2 | 8.74 | 0.16 | 8.12 | 0.18 | 8.0 | 0.0 | 7.0 | 0.0 |
| 8 | 0.2 | 0.2 | 1 | 5 | 0.36 | 5 | 12.52 | 0.31 | 12.79 | 0.45 | 11.0 | 0.0 | 10.0 | 0.0 |
| 9 | 0.2 | 0.2 | 4 | 5 | 0.19 | 5 | 16.15 | 0.51 | 12.71 | 0.45 | 14.0 | 0.0 | 10.0 | 0.0 |

Paired comparison [36] is a count of how many times the duration of process P1 (serialized) was longer than the duration of process P2, D*(P1)*>D*(P2)*. Generation of such result is done by running the process twice, once with serialized parameters and once with DB parameters and comparing the durations. The direct counting of pairwise comparison of simulation results are presented in the "paired" column of Table 3. The first process is preferred when its duration was longer in less than 50% of the cases. This measure had self-similarity in various ranges, i.e., it had relatively the same results for 1000 cases and 10,000 cases. A decision-making criterion based on these results indicates that when the percentage of D*(P1)*>D*(P2)* is less then 50%, then P1 (serial) is preferred, otherwise P2 is preferred. This method gives similar results to the two previous methods.

The decision-making approach presented in this article is applying the concepts of Law of Large Numbers (LLN) and Central Limit Theorem (CTL) in statistics [34]. Instead of comparing "paired" result we generate a difference function (*Dif=T(P1)-T(P2)*), and compute the difference function distribution.

In order to estimate the required number of simulation runs we made 100 runs for each set of parameters. For the serial process (case A), the estimates variance is $s_A^2=9.17$, thus for confidence $\alpha$=5%, $K_A > s_A^2 * 1.96/0.05 = 704.7$. In the same manner, we get for the DB (case B), $s_B^2=20.09$, $K_B$>1543.6. Using the results of 10000 simulations is more than enough.

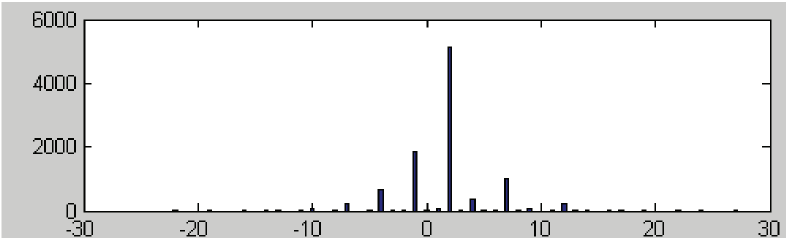


*Figure 4: Difference function distribution*

The difference function distribution for the example in Figure 3, is depicted in Figure 4. As expected, the difference of the basic process (i.e., no iterations) is the most common outcome (10-8=2).
The results of performing difference calculation statistics for 10000 runs and 1000 runs are presented in Table 3. A batch of 10000 runs is very large compared to the required number (according to the initial run of 100 cases) and a batch of 1000 runs is in the order of the required number. The calculations were made for α=5%. Under the assumption of no difference, the interval is $[-Z_{1-\alpha/2} \cdot s/\sqrt{R} \quad Z_{1-\alpha/2} \cdot s/\sqrt{R}]$ (equation A.6), where $s$ is the standard deviation calculated using equation A.5, and $Z_{1-\alpha/2}$ is the normal distribution percentile (Z=1.96 for α=5%).
If the average result is out the interval, we can decide that the total time of the serial process (*P1*) is significantly larger (smaller) than the parallel process (*P2*). Since the interval is symmetric, only the upper bound of the confidence interval is presented.

*Table 3. Decision-making based on difference*

| # | Paired | Difference *R*=1000 Interval (5%) | | | | Difference *R*=10000 Interval (5%) | | | |
|---|---|---|---|---|---|---|---|---|---|
| | % T(P1) >T(P2) | Avg | Std | Confidence Interval | Decision | Avg | Std | Confidence Interval | Decision |
| 1 | 35.08 | -5.04 | 12.52 | 0.76 | T(P1)< T(P2) | -5.02 | 12.48 | 0.24 | T(P1)< T(P2) |
| 2 | 32.63 | -15.79 | 30.03 | 1.87 | T(P1)< T(P2) | -15.86 | 29.99 | 0.59 | T(P1)< T(P2) |
| 3 | 41.62 | -8.47 | 32.76 | 2.02 | T(P1)< T(P2) | -8.42 | 32.72 | 0.64 | T(P1)< T(P2) |
| 4 | 49.43 | -0.55 | 5.43 | 0.36 | T(P1)< T(P2) | -0.56 | 5.41 | 0.11 | T(P1)< T(P2) |
| 5 | 48,95 | -3.91 | 12.91 | 0.80 | T(P1)< T(P2) | -3.96 | 12.85 | 0.25 | T(P1)< T(P2) |
| 6 | 54.13 | 0.98 | 14.97 | 1.01 | T(P1)~ T(P2) | 0.98 | 14.89 | 0.32 | T(P1)> T(P2) |
| 7 | 68.79 | 0.62 | 2.54 | 0.15 | T(P1)> T(P2) | 0.61 | 2.54 | 0.05 | T(P1)> T(P2) |
| 8 | 69.33 | -0.26 | 5.79 | 0.36 | T(P1)~ T(P2) | -0.26 | 5.76 | 0.11 | T(P1)< T(P2) |
| 9 | 70.72 | 3.43 | 6.72 | 0.41 | T(P1)> T(P2) | 3.44 | 6.68 | 0.13 | T(P1)> T(P2) |

Comparing the results in Table 3 with the results of Table 2, it was found that the decisions based on large sample (*R*=10000) were the same as the decisions based on averages. The difference function distribution converges to the distribution of the averages. Therefore, in the case of very large sample, such alignment of the results is expected.
The smaller sample (*R*=1000) was less decisive and the decision-making result indicated insignificant difference between different process parameters, i.e., a “don't care” decision. These “don't care” cases were aligned with the cased where using average, median, and paired results yielded different decisions. The indifference zone resulting from using confidence intervals in the smaller (but big enough) sample provides an important indication to decision-makers regarding preferred decision. While additional simulations could provide decisive statistical conclusions, in reality, we have a single process case for which a decision is sought. Therefore, if for a small and sufficient sample the result is

inconclusive, so it would be for the single case.

## 4. Discussion and conclusions

Unlike administrative processes or manufacturing processes where the order of activities can be logically extracted, the design activities modeled by the DSM have no predefined order. Furthermore, the product knowledge used for ordering keeps changing during the process. Consequently, the need arises to make choices between different possible process plans and process implementations. This article presented a statistical hypothesis testing method for aiding decision-makers in making such choices. The method applies not only to DSM-based planning but to other planning methods in which process simulations must be performed to calculate process measures used for decision-making.

The main conclusion that can be obtained from the results is that no business rule is better then others for all situations. Simulation of various parameters revealed that different process settings yielded different preferences (in regards to choosing the most applicable business rules). In iterative and continuously changing processes, general rules of thumb cannot support decision-making, and therefore simulation-based tools are required for supporting decision-making for the specific case context. The implementation of statistical analysis for the case of dynamically evolving process is presented in [38]; it demonstrates the significance of statistical analysis of simulation results for guiding the choice of process implementation, where rules of thumb cannot provide such guidance.

By utilizing such process simulations and statistical analysis, practitioners of process management (e.g., product managers of new products) can leverage existing and newly discovered knowledge for better planning and better reaction to changes that are inherent to NPD processes.

## Annex A – Statistics of stochastic simulated process

The following describes the statistical formulation in Asmussen and Glynn [34], for analyzing stochastic simulation results.

**Distribution definitions:**

Probability cumulative distribution *P* of a probability density function *f(x)* is defined by $P(x < X) = \int_{-\infty}^{X} f(x)\,dx$.

In the discrete case, we define the discrete distribution *P* of a probability mass function *p(x)* by $P(x < X) = \sum_{x_i < X} p(x_i)$.

**Process definitions:**

(i) A Markov Chain is a process $\{X_n\}$ with finite or countable state space, where the next state depends only on the previous state. Examples: a process defined by the transition probabilities $p_{ij}=P(X_{n+1}=j \mid X_n=i)$, i.e., $p_{ij}$ is the probability to move from state *i* to state *j*; as the serial process in [29], using probability DSM. Autoregressive process $X_{n+1}= aX_n+\varepsilon_n$, with $\varepsilon_n$ being an independent identically distributed variable, is another example. The Markov chain is time independent (Gilks *et al.*) [37].

(ii) A Markov process is time dependent, with finite or countable state space. For example, the time of making the transition is exponential with an intensity matrix $\Lambda=\lambda_{ij}$; and the process holding time at state *i* is exponential with rate $T=exp(-\lambda_{ij})$, and the next state *j* is chosen with probability $\lambda_{ij}/\lambda_{ii}$.

The main property of simulation process is the ability to generate large number of examples (i.e., as large as required given processing availability). This ability is used according to the Law of Large Numbers (LLN). For a stochastic random variable $W_n$, and the function $f(W_n)$; if for $n\rightarrow\infty$ the steady state $W_\infty < \infty$ (i.e., $W_\infty$ is a random variable with a limited distribution), then we can calculate the expected value of the function for $N \rightarrow\infty$, using Equation B.1. We get an asymptotic convergence to the expected value $E[f(W_\infty)]$ for the random variable $W_\infty$.

$$\frac{1}{N}\sum_{n=0}^{N-1} f(W_n) \rightarrow E[f(W_\infty)],\ N\rightarrow\infty \tag{A.1}$$

The expected value $z= E[Z]$ , where z is not available analytically, but $Z$ can be simulated. Using the Monte Carlo method, we simulate $R$ replicas $Z_1...Z_R$ of $Z$, and estimate the expected value $z$ by the statistic $z_R$

$$z_R = \frac{1}{R}\sum_{r=1}^{R} Z_r \tag{A.2}$$

Assuming the variance $\sigma^2$=Var[$Z$]<∞, the Central Limit Theorem (CLT) states that the distribution converges to normal distribution, as $R\rightarrow\infty$.

$$\sqrt{R}(z_R - z) \xrightarrow{D} N(0,\sigma^2),\ R\rightarrow\infty \tag{A.3}$$

Where $\xrightarrow{D}$ indicates distribution convergence.
The result can be interpreted as

$$z_R \overset{D}{\approx} z + \sigma V / \sqrt{R} \tag{A.4}$$

Where $\overset{D}{\approx}$ is interpreted as “has the same distribution as", and $V$ has a standard normal distribution $V$~$N(0,1)$; i.e., $z_R$ is distributed as $z$ plus an error. The error for large R is approximately normally distributed, and the approximation has a slow convergence rate $\sqrt{R}$ .
In practice, $\sigma^2$ is unknown, and should be estimated. The estimate is the sample variance $s^2$ defined by

$$s^2 \overset{def}{=} \frac{1}{R-1}\sum_{r=1}^{R}(Z_r - z_R)^2 = \frac{1}{R-1}\sum_{r=1}^{R} Z_r{}^2 - R\, z_R{}^2 \tag{A.5}$$

The use of ($R$-$1$) follows the standard statistical tradition for making this estimate unbiased, though for large R the difference is minor.
Using the CLT we can define the confidence interval $I_\alpha$ for $z$. Using the normal distribution cumulative function $\Phi(Z_\alpha)=P(Z<Z_\alpha)=\alpha$

$$I_\alpha = z_R \pm \frac{Z_{1-\alpha/2}\, s}{\sqrt{R}} \tag{A.6}$$

i.e., z∈ $I_\alpha$ with confidence level 1-$\alpha$. For the typical choice $\alpha$=5%, the corresponding interval is $z_R \pm 1.96\, s/\sqrt{R}$ .
For setting a required accuracy, we can make a two stages procedure. First stage will be a small simulation (e.g., R=50) for estimating the variance $s^2$, and estimating R accordingly; then simulating R occurrences.

## REFERENCES


[1] Browning T.R., Fricke E., Negele H., Key concepts in modeling product development processes, *Systems Engineering*, 9(2):104-128, 2006.

[2] Eppinger S.D., Nukala M.V., Whitney D.E.., Generalized models of design iterations using signal flow graph, *Research in Engineering Design*, 9(2):112-123, 1997.

[3] Kusiak A., Wang J.R., He D.W., Feng C.-H., A structured approach for analysis of design processes, I*EEE Trans. on Components, Packaging, and Manufacturing Technology* - Part A, 18(3):664-673, 1995.

[4] Westfechtel B., *Models and tools for managing development processes*, *Lecture Notes in Computer Science*, vol. 1646, Springer, Berlin, 1999.

[5] Karniel A. and Reich Y., From DSM based planning to design process simulation: A review of process scheme logic verification issues, *IEEE Transactions on Engineering Management*, in press**,** 2008(a).
[6] Karniel A. and Reich Y., DSM Based implementation of process schemes, submitted, 2009(b).
[7] Steward D.V., The design structure system: a method for managing the design of complex systems, *IEEE Transactions on Engineering Management*, 28:71–74, 1981.
[8] Gebala D.A., Eppinger S.D., Methods for analyzing design procedures, *ASME 3rd Int. Conf. On Design Theory and Methodology*, pp. 227-233, 1991.
[9] Eppinger S.D., Whitney D.E., Smith R., Gebala D., A model-based method for organizing tasks in product development, *Research in Engineering Design*, 6(1):1-13, 1994.
[10] Yassine A. and Braha D., complex concurrent engineering and the design structure matrix method, *Concurrent Engineering Research and Applications*, 11(3):165-176, 2003.
[11] Karniel A., Belsky Y., Reich Y., Decomposing the problem of constrained surface fitting in reverse engineering, *Computer-Aided Design*, 37:399-417, 2005.
[12] Browning T.R. and Eppinger S.D., Modeling impacts of process architecture on cost and schedule risk in product development, *IEEE Transactions on Engineering Management,* 49(4): 428-442, 2002.
[13] Abdelsalam H.M.E. and Bao H.P., A simulation-based optimization framework for product development cycle time reduction, *IEEE Transactions on Engineering Management*, 53(1):69-85, 2006.
[14] Eckert C.M., Keller R., Earl C., and Clarkson P. J., Supporting change processes in design: Complexity, prediction and reliability, *Reliability Engineering and Safety System*, 91(12):1521-1534, 2006.
[15] Maheswari J.U. and Varghese K., Project Scheduling using dependency structure matrix, *International Journal of Project Management,* 23:223–230, 2005.
[16] Cho S.H. and Eppinger S.D., A simulation-based process model for managing complex design projects, *IEEE Transactions on Engineering Management*, 52(3):316-328, 2005.
[17] Karniel A. and Reich Y., A coherent interpretation of DSM plan for PDP simulation, *in Proceedings of the International Conference on Engineering Design, ICED 07,* Paris, 2007(c).
[18] Brucker P., Drexl A., Möhring R., Neumann K., Pesch E., Resource-constrained project scheduling: Notation, classification, models and methods, *European Journal of Operational Research*, 112:3–41, 1999.
[19] Elmaghraby S. E., Activity nets: A guided tour through some recent developments, *European Journal of Operational Research*, 82:383-408, 1995.
[20] Kolisch R. and Padman R., An integrated survey of deterministic project scheduling, *Omega*, 29(3):249–272, 2001.
[21] Herroelen W. and Leus R., Project scheduling under uncertainty—Survey and research potentials, *European Journal of Operational Research*, 165:289–306, 2005.
[22] Herroelen W. and Leus R., Robust and reactive project scheduling: A review and classification of procedures, *International Journal of Production Research*, 42(8):1599–1620, 2004.
[23] Vonder S. van de, Demeulemeester E., and Herroelen W., A classification of predictive-reactive project scheduling procedures, *Journal of Scheduling*, 10(3):195-207, 2007.
[24] Smith R.P. and Eppinger S.D., A predictive model of sequential iteration in engineering design, *Management Science*, 43(8):1104-1120, 1997(b).
[25] O'Donovan B.D., Clarkson P.J., and Eckert C.M., Signposting: Modeling uncertainty in design processes, *International Conference on Engineering Design (ICED 03),* Stockholm, August 19-21, 2003.
[26] Coates G., Duffy A.H.B., Whitfield I., and Hills W., An integrated agent-oriented approach to real-time operational design coordination, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing,* 17:287–311, 2003.
[27] Choo H. J., Hammond J., Tommelein I. D., Austin S. A., Ballard G., DePlan: A tool for integrated design management, *Automation in Construction,* 13:313-326, 2004.
[28] Karniel A. and Reich Y., Simulating design processes with self-iteration activities based on DSM planning, *IEEE Proceedings of the International Conference on Systems Engineering and Modeling* - *ICSEM'07*, pp. 33–41, Haifa, March, 2007a.
[29] Sered Y., Reich Y., Standardization and modularization driven by minimizing overall process

effort, *Computer-Aided Design*, 38(5):405-416, 2006.
[30] Huberman B. A. and Wilkinson D.M., Performance variability and project dynamics, *Computational & Mathematical Organization Theory*, 11:307–332, 2005.
[31] Melo A.F. and Clarkson P.J., Design process planning using a state-action model, *in 13th Intl. Conf. Engineering Design (ICED 01)*, Glasgow, 2001.
[32] Lévárdy V. and Browning T.R., Adaptive test process – Designing a project plan that adapts to the state of a project, *Fifteenth Annual International Symposium of the International Council on Systems Engineering (INCOSE)*, July, 2005.
[33] Yassine A., Investigating product development process reliability and robustness using simulation, *Journal of Engineering Design*, 2006.
[34] Asmussen S. and Glynn P.W., *Stochastic simulation algorithms and analysis*, Springer, 2007.
[35] Karniel A. and Reich Y., Managing dynamic new product development processes, *in Proceedings of the 17th Annual International Symposium of The International Council on Systems Engineering- INCOSE'07,* San Diego, California, June 2007 (b).
[36] Reich Y. and Paz A., Managing product quality, risk, and resources through quality function deployment, *Journal of Engineering Design*, 19(3):249-267, 2008.
[37] Gilks W.R., Richardson S., Spiegelhalter D.J., *Markov chain Monte Carlo in practice*, Chapman and Hall/CRC, 1996.
[38] Karniel A., Managing the dynamics of product development process for new product development, *PhD dissertation,* Tel Aviv University, 2009.

Contact: Yoram Reich
Tel Aviv University
School of Mechanical Engineering
Tel Aviv 69978
Israel
+972-3-6407385
+972-3-6407617
Yoram@ENG.TAU.AC.IL
http://www.eng.tau.ac.il/~yoram

Arie Karniel is a CAD / PLM system engineer at Israel Aerospace Industries. He received his PhD in managing development processes knowledge from Tel Aviv University. He develops methods and processes for creating, transferring, and managing design knowledge of various design disciplines to the shop floor. He is reviewer of the Journal of Eng. Design and Research in Eng. Design.

Yoram Reich is a professor of mechanical engineering at Tel Aviv University. He received his PhD in applications of AI to design from Carnegie Mellon University. He develops design methods for early product development, knowledge management and information systems, as well as design theory and research methodology. He is Co-editor-in-chief of the journal Research in Engineering Design.