An Experimental Study for Simulation Based Assessment of Information System Design Performance

Bulent Ayyildiz, Ibrahim Akman* and Ali Arifoglu

***Abstract*—This paper presents an experimental study for evaluating the decision support value of queueing network (QN) based simulation models for information system design performance. For illustration, queueing network simulation models have been extracted corressponding to three annotated design alternatives of a selected case study. The design alternatives are produced using logical requirements of the selected system. The performance of each alternative is then predicted using quantifiable parameters considering the dynamics of the system such as service time, waiting time and number of entities waiting in the system. In particular, results have shown that the first alternative performs better than the other two in terms of the selected parameters. In general, the case study revealed that QN-based simulation models are capable to distinguish the performance of design alternatives in terms of selected parameters and under given assumptions. This also means that the use of simulation may lead to better designed information systems.**

***Index Terms*—Information System (IS), performance, queueing network, simulation,**

I. INTRODUCTION

The goal of the IS design process is to find the best possible design, within the limitations imposed by the requirements and the physical and social environment in which the system will operate. The designer plans 'how' an information system should be developed in order to make it functional, reliable and reasonably easy to understand, modify and maintain [1]. In many of the applications intuition and experience are used to provide the desired judgment on design performance rather than quantitative techniques, and designs are measured subjectively [2].

Using any one of the available design methods the design step produces an architectural design, a data design and a procedural design [3]. Among these, architectural design has primary impact on the quality of software since it is based on the logical model to represent the control relationships between components of IS and melds program and data structures. In application, all feasible architectural design alternatives are studied and then evaluated to identify the structure that meets the system objectives best.

Information System (IS) developers generally benchmark the performance of software components of proposed designs without considering their dynamics. This is unfortunate because IS dynamics are important and modeling these dynamics should be a routine part of design decisions in many IS applications [4]. In order to include system dynamics effectively, the design process must include techniques that facilitate the effectiveness of the verification of the target design resulting from that process. These techniques should be universal in nature and the system designer should be able to use these techniques to verify the design process with the help of a common set of tools [5]. Traditional stochastic queueing network simulation is one of such powerful analysis tools available to those responsible for the design of complex systems. By using QN simulation models for the dynamics of architectural design, one can increase the accuracy of early estimates of performance and even provide early capacity estimates to support hardware procurement. In addition, one may be able to anticipate problems and often indicate design changes before programming starts, thus saving effort at an earlier stage of development, rather than waiting until after the system is implemented [2].

However, the use of QN-based simulation has not been a common practice for IS design [6] due to the fact that dynamics are not likely to be considered as a major issue in many projects.

As such, the decision support value of QN-based simulation for architetural design of information systems needs to be discovered and this sequel presents an experimental study for this purpose. The experiment predicts the performance of design alternatives for the information system using QN simulation models based on synthethic workload and in terms of scarce resources such as time and

B. Ayyildiz is the head of Software Development Branch, Information Systems Department of theTurkish General Staff, 06100 - Bakanlıklar Ankara Turkey. ((Phone: +90-3124022211, e-mail: bayyildiz@tsk.mil.tr).

I. Akman is the head of Computer Engineering Dept., Atilim University, 06836 Incek Ankara Turkey. (Phone: +90-3125868300, Fax: +90 312 5868091, e-mail: akman@atilim.edu.tr).

A. Arifoglu is with the Informatics Institute, Middle East Technical University, Inonu Bulvari, Ankara, Turkey. (e-mail: sas@ii.metu.edu.tr)

* corressponding author

cost. Assumptions of the application are based on the dynamic behaviour of the system.

II. THE METHODOLOGY

The experiment proposes a methodology that uses two main steps:

- Extraction of design alternatives,
- Performance modelling and analysis.

A.Extraction of Design Alternatives

This study is based on IS design containing more than one function or module. These functions or modules are combined with various hierarchical relations in structure chart form through the analysts' intuition or experience to obtain a design of a working system satisfying the user's needs [8]. Alternative design models may be produced in one of the following ways:

- Different architectural designs are produced by different modules/functions for the information system of interest.
- Different architectural designs are produced by the same modules/functions but with different hierarchical relations for the information system of interest.

In this study the combination of these two is adopted.

B. Performance Modelling and Analysis

One of the approaches to solving performance models of design problems in software engineering is QN-based simulation [3]. The major two approaches for performance simulations are:

- Trace driven simulation: This has been mostly used for deterministic models and often for the systems without queues such as the evaluation of hardware devices and operating system algorithms.
- Stochastic discrete event simulation: This approach is based on the simulation of some queuing models driven by synthetic non-executable workload models. This study adopts the second approach since it is an extremely versatile and powerful technique in performance modelling for software design and implementation with the following advantages [9]:
- Provides insight to the performance of designs before any programs are developed.
- Describes the system performance from the user's perspective, showing where there are problems in transactions or communications.
- Shortens the development time frame with early detection of problems and decreases the effort spent on code that must be reworked due to design problems.

Details of the general framework for the experiment is given in Fig.1. This framework initially takes project-specific logical requirements, time constants, conditions for the dynamics of the system, assumptions and limitations of the system as its input. On output, using simulation results, the best design alternative is obtained.

The processes in Fig.1 are not strictly sequential. It may often be necessary or desirable to return to one of the previous steps for validation and reformulation.

Step 1. Produce Design Alternatives:

Design alternatives are produced using different sets of scenarios at this stage. These scenarios are based on the number and characteristic of components, input/output data elements, their types and the alternative paths that the input data should follow to be transformed into output data.

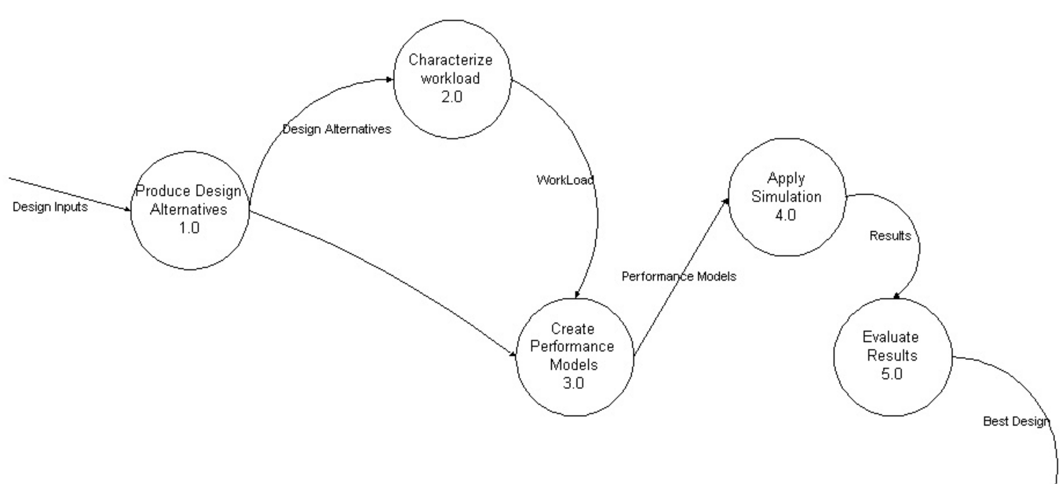

Fig. 1: General Framework for Process Flow of the Experiment

Extra functional components such as reliability, security and capacity are other system characteristics of interest but are not used here.

Step 2. Characterize Workload:

A stochastic QN-based performance model is meaningful due to its workload which represent requests to the system arriving from external users. This process provides workload model for each of the design alternatives such as containing total number of users, the delay between the completion of a transaction and start of a new one, the pattern of interarrival times, service (processing) times.

Step 3. Create Performance Models:

Queueing network models provide the analyst with a powerful tool for designing and evaluating the performance of dynamic systems. Along with the syntethic workloads, this stage is used to develop the QN-Based performance models for each of the architectural design alternatives of the information system of interest. The key elements, in this case, are the entities and servers. The "entity" refers to an inquiry from the system and the term "server" refer to modules of the design.

Step 4. Apply Simulation:

The fourth step of the proposed approach is the development of a simulation program considering the workload and queueing model to obtain prediction of the selected parameters. GPSS is a problem oriented simulation software which is well suited to queueing systems [10] and is used to implement the proposed approach.

In order to convert the static behaviour of the architectural designs to a form which reflects the dynamic behaviour of the resultant information system, its elements are mapped onto the elements of simulation. The relationships between these elements are given in Fig. 2.

The elements related to design functions are mapped to servers while the data elements of design are mapped to transactions. That mean, sources, entities, inputs and outputs correspond to transactions, and processes/functions/events correspond to servers/facilities.

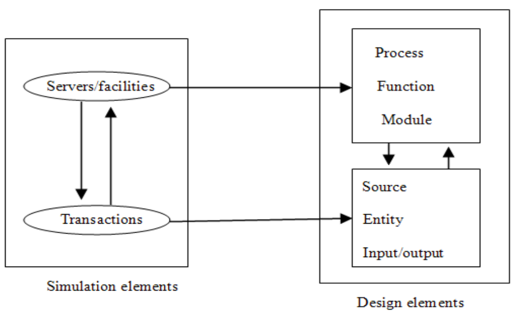

Fig. 2. Map of simulation system element

Step 5. Evaluate Results:

QN-bases simulation results are used to compare and predict the performance of the alternatives. For sound conclusions proper statistical analysis is needed in terms of selected parameters.

In this study, we selected a standard parameters to compare and judge the performance of alternative designs for simplicity.

1) Total time (TT) : Total service time.

2) Maximum content (MC) : Highest number of entries waiting in the system in a given period.

3) Average content (AC) : Total waiting /simulation time in a given period.

The total time is related to employee cost and capability of the resultant information system. The other two parameters are related to the physical conditions of the resultant system such as the number of service stations and number of waiting lines.

The simulation produces statistically independent generations and thus statistical analysis is based on standard conventional measures such as average and standard deviation.

Analysts may extend the output analysis to include other parameters such as number of simulation runs conducted, total number of replications, total number of entries serviced in a given period and absolute clock time, and to include more advanced statistical techniques such as experimental design and variance reduction techniques.

III. THE CASE STUDY

A. Queueing Network and Workload Models

The present stock control subsystem of the departmental information system at the university was selected for the case study. The architecture of this subsystem contains eight main functions namely accepting requests, arrival of materials, materials withdrawal, end-of-year operations, print lists, add records, delete records and update records.

External entities to the stock control subsystem are the members of the department. There is one office, one office staff with a personal computer and one store very close to the office in the present subsystem. Briefly an entity arrives to the stocks office with a request and the office staff provides the appropriate service for that request. Therefore system status changes through an arrival or a departure. The office staff is either busy or idle at the time of the request. A request of an entity is served if the office staff is idle i.e. is not serving another member. Otherwise, the entity waits in the queue where the queue discipline is assumed to be "first come first serve" (FIFO). Each entity can demand only one request at a time. After a request is completed, the office staff returns to his/her desk to update stock records and then to receive the next request.

The office opens at 8 am. and closes at 4 pm. During lunch time, a reserve office staff comes to serve for an hour. Therefore there is no service break in a working day. After 4 pm. no entity is accepted to the office. However, all the entities waiting in the queue at the closure time are served by the office staff. Requirements analysis showed that, corresponding to the functions of stock control subsystem, request types are: arrival of materials, materials withdrawal, end-of-year operations, print lists, add records, delete records, and update records requests.

Based on the observations arrivals are assumed to be exponentially distributed. Of the arrivals 25 percent arrive with an average interarrival rate of 7 time units, 50 percent arrive with an average interarrival rate of 10 time units and 25 percent with an average interarrival time of 13 time units. Observations have also shown that each arrival submits a different type of request into the system. Of the requests, 40 percent is arrival of material, 40 percent is material withdrawal, 7 percent is add new record, 6 percent is delete a record, 4 percent is print lists, 3 percent is update a record and 1 percent is end-of year operations.

B. Design Alternatives and Simulation Programs

This study uses architectural design of the present stock control subsystem at the university and its two alternatives constructed with different hierarchical combinations of the modules of the present system. Information collected during system analysis, intuition and experience lead to development of the alternative designs.

The architectural structure for the first two alternatives contain five branches controlled by a main controller named "request" whereas the third alternative contains only four branches. Observations from figurative representation of three alternatives are as follows:

- The structure of "end-of-year branch is the same in all three alternatives.
- The "arrival of material" and "material withdrawal" branches of the first alternative were completely changed in the other two alternatives. Instead of these branches the second alternative contains "material additions" and "material deletions" branches and third alternative contains "material additions and lists" and ""material deletions" branches.
- The "update stock master file" branch of the first alternative exits in alternative-2 and alternative-3 but with a different structure.
- The "print lists" branch is the same in the first two alternatives.
- The only difference between the second and third alternatives is that "print lists" and "material additions" branches are combined in the third alternative.

Simulation programs prepared for each alternative consist of segments corressponding to above mentioned branches. The list of these segments, along with their descriptions, is as follows:

- *Faculty Member Request Segment:* This segment shows the arrival of faculty members, queue in the office and starting point of the service of the faculty member.
- *Arrival of Material Segment:* This segment consists of facilities that complete the processes for arrival of materials.
- *Material Withdrawal Segment:* This segment consists of facilities that complete the processes for material withdrawals and, in the case of lack of material, it includes requesting the material from the warehouse or purchase department.
- *End-of-Year-Operations Segment:* This segment contains facilities to delete arrival file and withdrawal files which are created in the year.
- *Print Lists Segment:* This segment contains facilities to list, sort and print out the requested materials.
- *Update Stock-Master-File Segment:* This segment contains facilities to add a new material to the file, to delete a material from the file and to update the information of material in the file.
- *Material Additions Segment:* This segment contains two parts, which are the arrival of material and the arrival of a new material, and their related facilities which complete these processes.
- *Material Deletions Segment:* This segment contains two parts, which are material withdrawal and the deletion of a material, their related facilities which complete those processes, including the request of the material from the warehouse or purchase department for material withdrawals.
- *Material Additions and Lists Segment:* This segment contains three parts which are the arrival of material, arrival of a new material and print lists, and the related facilities which complete these processes.
- *Faculty Member Stop Segment:* This segment shows the termination of the faculty member's service.
- *Simulation Stop and Report Segment:* This segment shows the termination of the simulation and reports the values of the determined parameters.

These segments are produced using different combination of modules with different hierarchical organizations and then are combined to represent the structure of architectural design alternatives. The list of segments for each alternative is:

Alternative-1: Faculty Member Request, Arrival of Material, Material Withdrawal, End-of –year-operations, Print Lists, Update Stock-Master-File, Faculty Member Stop and Simulation Stop and Report Segments.

Alternative-2: Faculty Member Request, Material Additions, Material Deletions, End-of year-operations, Print Lists, Update Stock-Master-File, Faculty Member Stop, Simulation Stop and Report Segments.

Alternative-3: Faculty Member Request, Material Additions and Lists, Material Deletions, End-of-Year-Operations, Update-Stock-Master-File, Faculty Member Stop and Simulation Stop and Report Segments.

The facilities in simulation are the modules to be executed corresponding to requests. The list of facilities of the system and, based on the observations, their assumed execution times are as follows.

1) Acess Material Record (AMR) (5 time units)
2) Increase Stock level (ISL) (4 time units)
3) Append Of Arrivals (AOA) (5 time units)
4) Carry Material Between Office and Store (CMBOS) (8 ±1 time units)
5) Decrease Stock level (DSL) (4 time units)
6) Check Stock Level (CSL) (4 time units)
7) Append Of Withdrawals (AOW) (5 time units)
8) Material Requested From Warehouse (MRFW) (6 time units)
9) Material Requested From Purchasing Department (MRFPD) (6 time units)
10) Get Arrival File (GAF) (3 time units)
11) Delete Arrival File (DAF): Execution time is decided with the following rule

Probability of being required	0.25	0.75
Deletion time units	2	3

12) Get Withdrawal File (GWF) (3 time units)
13) Delete Withdrawal File (DWF) (same as noted in 11)
14) Materials List Part (MLP) (6 time units)
15) Sorted List Part (SLP) (5 time units)
16) Print-Out Part (POP) (8 time units)
17) Add Record Part (ARP) (5 time units)
18) Delete Record Update (DRU) (5 time units)
19) Update Record Update (URU) (6 time units)
20) Time Delay Between Levels (TDBL) (5 ±1 time units).

The workload assumptions are kept the same for each alternative in order to provide a base line for comparison of the results. In the simulation, we assume that requests arrive from the members of the university and the server is the office staff. Service time for each request type depends on the total execution time of the corresponding modules.

IV. RESULTS

GPSS simulation tool is used to model IS dynamics and to run performance models with given workloads in this study since GPSS is a problem oriented simulation language and well suited to queueing network systems [10][11].

Twenty-two runs (corresponding to one month) were performed for each of the alternative designs. Sensitivity tests have generally shown that outputs are not particularly sensitive to any of the input parameters.

The summary statistics for each alternative are given in Tables I and II.

A close inspection of Table I shows that total service time (TT) spent, highest number of entries waiting in the system (NW) and total waiting times (WT) are the lowest for the present system (Alternative-1). The standard deviations for the present system are also the smallest except NW. Therefore, the present system is relatively more productive since TT is related to employee cost and other parameters, NW and WT are related to the physical conditions such as the need for office area, number of waiting lines, number of servers etc. However, the total number of people waiting in the queue (WT) seems to be too high for all alternatives, which should be taken as an indication of the need to have a better organization of the stock control subsystem. This may be possible by changing the physical design of the office and stock control area and by increasing the number of employees. The average content (WT) figures also support this conclusion.

Furthermore, tests for TT values between alternative-1 and other alternatives also support the hypothesis that there is a significant difference between alternatives at 95% significance level. However, the same tests for WT and NW parameters do not show any significant difference. Although the second alternative is slightly more efficient than the third one in total service time (TT), the tests results state that there is no significant difference between them.

Table I. Summary statistics for alternatives

	Alternative-1		**Alternative-2**		**Alternative-3**	
	Average	**Std.Dev.**	**Average**	**Std. Dev.**	**Average**	**Std.Dev.**
TT	1038	170	1217	207	1224	204
NW	26	7	29	7	29	7
WT	12	4	14	4	14	7

Table II. Confidence limits for alternatives

Prob.	**95%**	**99%**

Val.												
	ALT-1		**ALT-2**		**ALT-3**		**ALT-1**		**ALT-2**		**ALT-3**	
	Low.	**Up.**	**Low.**	**Up.**	**Low.**	**Up.**	**Low.**	**Up.**	**Low.**	**Up.**	**Low.**	**Up.**
TT	962	1113	1125	1309	1113	1314	935	1141	1091	1342	1100	1348
NW	23	29	26	32	26	32	23	30	25	33	25	34
WT	11	14	12	15	12	16	10	15	11	16	11	16

V. IMPACT OF THE METHOD

This experimental study employs traditional queueing network based simulation modelling. As a result of the modeling efforts the user is be able to compare the present and alternative systems and make extensive changes before the implementation of the system. The experiences have shown that QN-based performance modelling certainly reduces the risk of unacceptable system performance in advance by anticipation. This, of course, reduces the cost and schedule delays and increases reliability in a development project.

It is important to note here that QN-based simulation has the advantage of testing different theoretical performance targets for later stages because it is parametrized. Another important advantage is that it enables the user to consider different workloads and configurations in modelling and decision.

However, as noted in [6], the analyst must understand the importance of sufficient sample size/number of replications of simulation and should be well experienced in correct interpretation of the simulation output. Besides the cost of conducting a dynamic analysis may be high. Furthermore, interpretation of the results should not be done blindly and the analyst should pay extra care for protecting himself against misinterpretation.

Finally, quantitative data may be used as a base line for the modelling and analysis of new projects.

VI. CONCLUSION

In this study, an experiment for the assesment of QN-based simulation approach for the performance evaluation of information system designs is presented. The experiment used the present stock control subsystem at the university and two of its alternatives.

The alternative design models are obtained by combining modules with different hierarchical relations based on information collected during analysis, intuition and experience. The representation of design alternatives are mapped to QN-based models using syntethic workload. Monthly simulation runs are performed for each alternative by using GPSS simulation software. The statistics obtained as a result of the simulation runs show that QN-based simulation is significantly useful in providing information about the performance of design alternatives. Particularly, total service time, highest number of entries waiting in the system and total waiting time are capable of investigating IS design performances.

In general, the case study revealed that the QN-based simulation models can increase the accuracy of early estimates of performance and indicates design changes before programming starts. It, therefore, leads to saving effort at an earlier stage of software development. Besides the results have also shown that QN-based performance simulation approach is capable of extracting other usefull information for information systems under consideration such as the need for additional resources and facilities.

ACKNOWLEDGMENT

We would like to thank to Dr. Alok Mishra for his effort in providing some recent literature.

REFERENCES

[1] Sommerville, I., Software Engineering, Sixth Edition, 2001, Addison-Wesley

[2] Hall T. et. al., Implementing Software Process Improvement: An Empirical Study, *Software Process Improvement and Practice*, 2002, 7: 3-15.

[3] Pressman, R.S., *Software Engineering: A Practitioner's approach*, Fifth edition, 2001, McGraw Hill.

[4] Warren J. R. et al., Stochastic Simulation of Information Systems design From data Flow diagrams, *J. Systems software*, 1992, pp.191-99.

[5] Hamilton, M. & Zeldin, S., The Relationship Between Design and Verification, *Journal of Systems & Software*, Vol 1, 1980, 29-56.

[6] Warren J. R. et al., Experimental Evaluation of a Simulation Environment for Information System Design, *J. Systems software*, 1996, 32, pp. 3-20.

[7] Pfahl D. And G. Ruhe, IMMoS: A Methodology fo Integrated Measurement, Modelling and Simulation, *Software Pprocess Improvement and Practice*, 2002, 7: 189-210.

[8] Anderson, G.E., The Coordinated Use of Five Performance Evaluation Methodologies", *Communications of the ACM* Vol. 27, No. 2, 1982, 119-125.

[9] McBeath D.F. and W.S. Keezer, Simulation in Support of Software Development, *Proceedings of the 1993 Winter Simulation Conference, G.W. Evans et. al. (eds), 1993*, pp.1143-51.

[10] Herper H. and I. Stahl, Micro-GPSS on the WEB and for Windows: A Tool For Introduction to Simulation in High Schools, *Proceedings of the 1999 Winter Simulation Conference,* P.A. farrington *et. al. (eds)*, 1999, pp. 298-306.

[11] Schriber, T., *An Introduction to Simulation Using GPSS/H,* Wiley, New York, 1991.