DOI: 10.47899/ijss.1193183

Queue models Evaluation of Simulation for Managerial Decision Support Systems: Application of Two-Stage Production Control*

Kuyruk Modelleri Simülasyonunun Yönetsel Karar Destek Sistemleri İçin Değerlendirilmesi: İki Aşamalı Üretim Kontrolü Uygulaması

Ahmet KAYA[1]

Abstract

Managers benefit from information systems in decision-making processes. A number of cognitive tools are included in decision support systems. When these tools are examined, it is seen that most of them produce statistical solutions. However, in some cases, statistical solution approaches cannot produce exact results. At this point, using simulation methods can give effective results in making managerial decisions. In order to benefit effectively from simulation methods in decision support systems and to obtain results that can support decisions, it is necessary to define the system with all its dimensions and features. In this study, queuing models simulation, which is one of the information tools that strengthen management decisions, is introduced and a two-stage production control application, which is possible to be made in a factory environment selected as the application environment, is introduced by using GPSS macros, which is a system simulation package.

Keywords: Decision Support System (DDS), Managerial decision making, System simulation, Simulation and modelling, Production control

Öz

Karar alma süreçlerinde yöneticiler bilişim sistemlerinden faydalanmaktadırlar. Bir dizi bilişsel araç, karar destek sistemleri içinde yer almaktadır. Bu araçlar incelendiğinde birçoğunun istatistiksel çözümler ürettiği görülmektedir. Buna rağmen bazı durumlarda istatistiksel çözüm yaklaşımları sonuç üretememektedir. Tam bu noktada simülasyon yöntemlerini kullanmak yönetsel kararların alınmasında etkili sonuçlar verebilmektedir. Simülasyon yöntemlerinden karar destek sistemlerinde etkin şekilde faydalanmak ve kararlara destek olabilecek nitelikte sonuçlar elde edebilmek için, sistemin bütün boyutları ve özellikleri ile tanımlanmasına gereksinim vardır. Bu çalışmada, bilişim araçlarından kuyruk modelleri simülasyonu, yönetim kararlarını güçlendirirken, bir fabrika ortamında yapılması muhtemel iki aşamalı üretim kontrolü uygulaması GPSS makroları kullanılarak uygulama ortamı olarak seçilmektedir. Ayrıca, simülasyonu kullanan bir sistem paketi olan GPSS makrolarının kullanımı ile bir fabrika ortamında gerçekleştirilebilecek iki aşamalı üretim kontrol uygulaması tanıtılmaktadır.

Anahtar Kelimeler: Karar Destek Sistemi (KDS), Yönetsel karar verme, Sistem simülasyonu, Simülasyon ve modelleme, Üretim kontrolü

1. INTRODUCTION

Managers often have to make decisions. The decision-making process aims to think multi-dimensionally and transform environments of uncertainty into relatively stable environments. In addition, it is a process that requires considering the situations of individuals who will be affected positively or negatively by the decisions taken, possible financial returns, losses and risks. The effects of the decisions made by the managers are wide. There are a number of changes that each managerial decision will cause. Changes are generally aimed at increasing competitive advantage and productivity and can be used to create new employment areas (Peppard, 1993: 1). In this way, it becomes possible to implement a new dynamism with the information required by change, to capture new

* In this article, the principles of scientific research and publication ethics were followed. / Bu makalede bilimsel araştırma ve yayın etiği ilkelerine uyulmuştur.

[1] **Ahmet KAYA**
ORCID ID: 0000-0002-6105-0787
Prof.Dr., Ege Üniversitesi, Tire Kutsan Meslek Yüksekokulu, Bilgisayar Programcılığı, İzmir, Türkiye. ahmet.kaya@ege.edu.tr
Prof.Dr., Ege University, Tire Kutsan Vocational School, Computer Programming, İzmir, Türkiye. ahmet.kaya@ege.edu.tr

Geliş Tarihi/*Received* : 22.10.2022
Kabul Tarihi/*Accepted* : 24.12.2022
Çevrimiçi Yayın/*Published* : 24.12.2022

Makale Atıf Önerisi /Citation (APA):
Kaya, A. (2022). Queue models Evaluation of Simulation for Managerial Decision Support Systems: Application of Two-Stage Production Control. *İzmir Sosyal Bilimler Dergisi*, 4(2), 89-96. DOI:10.47899/ijss.1193183

gains and opportunities, and to adapt to changes (Kaya, 2015:348-350) and (Gates, 1999).

There are a number of systems used to support management decisions in terms of businesses. These systems are called decision support systems. Most of the decision support systems are cognitive and are obtained as a result of the modelling process. Simulation methods are used because modelling is difficult, data acquisition is impossible and stochastic processes cannot be applied. Even in the case of partial modelling of the process, it is possible to obtain successful results from the system simulation. In order to obtain the expected results from the simulation processes, all the behaviours of the system should be transferred to the computer environment and all possible situations should be taken into consideration. Otherwise, simulation methods may make it possible to obtain results that may cause serious risks and misconceptions. In this respect, simulation should be considered as a cognitive process that should be considered if stochastic tools cannot be used.

2. QUEUE MODELS

Queuing systems are one of the most important random processes frequently encountered in simulation and modelling studies. A queuing system basically consists of a population represented by potential customers, a waiting line for customers, and a service period. Queues of people in banks, stops, supermarkets, hospitals, ticket offices and similar places, ships waiting to be loaded or unloaded at the port, business queues waiting to benefit from a computer system can be given as examples. The concepts in the queuing system are generally key concepts. A group of machines in a factory make up the population of potential customers, machines that fail make up customers for the repair service. If the repair person is busy with the service of another malfunctioning machine, the malfunctioning machines will form a queue (waiting line). As can be understood from the examples, the term customer refers to people, parts, machines, planes, ships, computer work, etc. similar jobs.

In the queuing theory, if there are many servers that can fulfil the requests of all customers at the same time, queuing problems will not arise in such systems.

2.1 Some Characteristics of the Queuing System

Characteristics that are likely to occur in the queuing system are defined by (Korukoğlu, 1995) as follows:

- *Population*: The population of potential customers can be either finite or countably infinity (many).
- *Arrival Process*: Arrivals can be at scheduled times or at random times depending on a certain probability distribution.
- *System Capacity*: System capacity is the customer limit that the system can accommodate at any given time. It affects the system capacity, the formation of the queue with a finite or countably infinity.
- *Service Process*: The two main factors affecting the service process are the service time and the service channel. Service time; the time required for the service, and the service channel; Indicates the number of individuals that can be served together. Apart from these two factors, the availability of the service is important. For example, services can affect the downtime of a used tool or machine. In addition, as the length of the queue increases, the service time may be affected, etc. features are present. According to the structure of the service time, the service time can be either fixed or stochastic.
- *Queue Discipline*: It determines the formation, style and number of the queue (Parallel queues). The major queuing disciplines are:

i. *FIFO (First In First Out):* The English abbreviation of FIFO is First In First Out. It is one of the most widely used methods in the industrial field. By means of this method, goods are not kept on the side and all goods are used in order. According to the FIFO method, the goods are queued. This order is determined according to the delivery time of the goods. The goods that come first are placed at the top of the list, while the goods that come later are placed at the back of the list. After sorting in this way, the use of the goods is carried out according to the list. Thus, there are no goods waiting on the side for a long time. All of the goods are used in order, and when selling, they are sold and disposed of in order. Among the advantages of using the FIFO technique are; It makes it easier to keep track of inventory, reducing record keeping as the oldest items are constantly used. In-stock items represent final prices, thus offsetting the cost of goods sold.

However, it has several disadvantages. With inflation in inventory costs, taxable income increases. It makes it difficult to calculate costs when inventory is exchanged or returned.

ii. *LIFO (Last In First Out):* Last in, first out (LIFO) is an inventory valuation method based on the assumption that the last stock item will be sold first. However, this technique has proven to be counterintuitive and illogical to the inventory movement. In a scenario where there is inflation in an economy, the value of unsold goods will decrease while the value of cost of goods sold will increase, resulting in lower profits and income tax. However, in a deflationary scenario in an economy, while the value of unsold goods increases, the cost of goods sold decreases, resulting in high profits. Other disadvantages of using the LIFO method of stock valuation include: It causes a larger discrepancy between the cost basis on initial inventory and the final market price, it complicates the interpretation of operating activities as well as current inventory activities in a company.

iii. *Priority Queue:* In a priority queue, items are processed based on their priority level, with higher priority items being processed before lower priority items. This can be useful for handling time-sensitive or important tasks, but it can also lead to unfairness if some items are given higher priority than others.

iv. *Round-Robin:* In a round-robin queue, items are processed in a rotating order, with each item being given a fixed time slice to be processed. This can help to prevent one item from monopolizing the queue and can be useful for handling a large number of items with similar priority levels.

v. *Weighted Fair Queue:* In a weighted fair queue, items are processed based on their weight, with heavier items being given a larger share of the queue's resources. This can be useful for allocating resources fairly among different types of items.

vi. *Random Early Detection (RED):* In a RED queue, the probability of an item being dropped from the queue increases as the queue becomes more congested. This can help to prevent the queue from becoming overloaded and can improve overall network performance.

There are many other queue disciplines that can be used, and the appropriate queue discipline for a particular application will depend on the specific requirements and goals of the system.

In addition to what was pointed by Korukoğlu, some general features that may arise in a queuing system can be listed as follows:

- Arrival rate: This is the rate at which items (e.g., packets, customers, etc.) arrive at the queue. The arrival rate can affect the length of the queue and the waiting times for items.
- Service rate: This is the rate at which items are processed by the queue. The service rate can affect the length of the queue and the waiting times for items.
- Queue length: This is the number of items waiting in the queue. The queue length can affect the waiting times for items and the utilization of the system.
- Utilization: This is the proportion of time that the system is busy processing items. High utilization can lead to longer waiting times for items, while low utilization may indicate that the system is underutilized.
- Throughput: This is the rate at which items are processed by the system. The throughput can be affected by the arrival rate, the service rate, and the queue length.
- Waiting time: This is the time that an item spends in the queue before being processed. The waiting time can be affected by the arrival rate, the service rate, and the queue length.
- Response time: This is the total time that an item spends in the system, including the waiting time and the processing time. The response time can be affected by the arrival rate, the service rate, and the queue length.

These characteristics can be used to analyze and optimize the performance of a queuing system. For example, by analyzing the arrival rate, the service rate, and the queue length, it may be possible to identify bottlenecks or inefficiencies in the system and to make changes to improve performance.

3. RANDOM NUMBERS

Chance or random numbers used in simulation planning studies should have some properties. The most important of these features is that numbers can be obtained by chance. The dependence on chance is defined as follows: As a result of the derivation of n independent numbers by any means, each number must be able to be derived with 1/n probability and be independent of the number derived before it (Atıl, 1980; Aydin and Dalkilic, 2018). This feature is sufficient in terms of using the number n as a chance number. While the derivation of chance numbers was made with different and primitive methods at the beginning, today it can be done with an extremely fast approach that can fully provide the properties of the distributions of numbers to which they belong.

In random events, the sequence of symbols or steps often has no order (Kösemen et al.,2018a). It does not contain a predictable or predictable pattern, combination or pattern (Kösemen et al.,2018b). Individual random events are by definition unpredictable, but the frequency of different outcomes on repeated events (or "trials") is predictable if the probability distribution is known. For example, when rolling two dice, the outcome is absolutely unpredictable, but the sum of 7 will tend to occur twice as often as 4. In this view, randomness is not randomness; It is a measure of the uncertainty of an outcome. It applies to the concepts of randomness, chance, probability and information entropy (Cabuk et al., 2017).

In statistics, a random variable is the assignment of a numerical value to every possible outcome of an event space. This association facilitates the identification and calculation of probabilities of events. A random process is a set of random variables whose results do not follow a deterministic pattern, but an evolution defined by probability distributions. Randomness is most commonly used in statistics to indicate well-defined statistical features.

In computer programming, a random number is a number generated by a computer program that is intended to be unpredictable. Random numbers are often used in programs to simulate real-world events or to generate data for various purposes, such as testing algorithms or generating cryptographic keys. There are several ways to generate random numbers, including using algorithms and hardware devices that produce truly random numbers, as well as using algorithms that generate pseudo-random numbers.

True random numbers are generated by physical processes, such as the radioactive decay of atoms or the movement of electrons in a circuit. These processes are inherently random and can be used to generate random numbers. However, true random number generators (RNGs) can be expensive and may not be practical for all applications.

Pseudo-random numbers, on the other hand, are generated using algorithms that produce a sequence of numbers that appears random, but is actually deterministic. Pseudo-random number generators (PRNGs) use a seed value and a set of rules to produce a sequence of numbers that appears random, but will be the same every time the program is run with the same seed value. PRNGs are commonly used in computer programs because they are fast, easy to implement, and produce good results for many applications.

It is important to note that both true and pseudo-random numbers have limitations and are not suitable for all applications that require truly random numbers. For example, pseudo-random numbers may not be suitable for cryptographic purposes, as they can potentially be predicted if the seed value is known. In these cases, it may be necessary to use a true random number generator or to gather randomness from other sources, such as user input or external events.

4. SIMULATION APPLICATION TOOLS

Simulation requires real-world randomness to be generated within the model as well. How to decide about this randomness structure and the selection of probability distributions corresponding to the related chance variables are very important in terms of being realistic in the simulation model and interpreting the results (Korukoğlu, 1995).

In order to decide on this issue, examining the data obtained from the real world plays an important role. However, it is also very important that these data are obtained in sufficient size and based on appropriate sampling methods.

Sufficiently large and inappropriate samples may result in the use of incorrect distributions. Using wrong distributions can cause wrong decisions to be taken from the simulation results, which is larger than the selection error to be made at the beginning.

Information from real events;

i. Direct sample information themselves,

ii. Summarizing statistics such as mean, variance, minimum, maximum, mode, median, skewness, kurtosis,

iii. Non-qualitative (qualitative) information based on experience, for example from people who may be related to the relevant problem or situation.

In this respect, after the system modelling is done, the scenario must be transferred to a computer system. This necessity arises from the hardware features of computers in terms of processing speed and data storage capacity, and the prevalence of simulation software and general-purpose programming languages running on this hardware. Thanks to these possibilities, information technology is used intensively not only for operations research problems such as queuing models, but also for simulating physical systems that cannot be applied in real life (aircraft simulators, architectural simulators, etc.).

Law and Kelton (1991) listed the main features that draw attention in the computer programming of modelling studies examined within the scope of discrete position simulation in terms of simulation technique such as queuing models and source systems modelling;

- Random number generation from a uniform U(0,1) distribution between 0 and 1,
- Derivation of random values from a known probability distribution,
- Running the simulation clock,

- Establishment of the control system in transition to the appropriate simulation blocks,
- Adding and removing records to the simulation list,
- Use of appropriate data analysis methods,
- Printing the results,
- Error monitoring is indicated as.

These and some of the features that will be mentioned later force the use of special-purpose simulation languages in simulation. These languages later led to the expansion of the use of simulation techniques. However, there has been a long-standing discussion of advantages and disadvantages between special-purpose simulation languages and general-purpose programming languages in terms of programming simulation scenarios on the computer.

5. COMPARISON OF SIMULATION LANGUAGES AND GENERAL-PURPOSE LANGUAGES

One of the most important decisions an analyst must make when modelling a simulation scenario is which language to choose. This choice can significantly affect the success and timing of the simulation project. Advantages and disadvantages of simulation languages compared to general-purpose languages such as C, C+, C++, C# or pascal are given (Law and Kelton, 1991).

Simulation languages provide most of the features required from programming a particular simulation model, which can be beneficial in reducing programming time.

Basic blocks are easier to create than programming languages.

The use of simulation languages allows easy modifications to the simulation model.

Most simulation languages provide dynamic memory allocation during processing.

Easier to debug as there are fewer program lines (So it may be more difficult for users to track bugs in versions).

On the other hand, most simulation models (especially those close to the security and war industry) are still built with general-purpose programming languages.

In most applications, general purpose programming languages are more amenable to simulation languages.

When a program written in a general-purpose programming language is created effectively, it can reach the analysis stage in a shorter time than one written in a simulation language.

General purpose programming languages have a more flexible structure.

General purpose programming languages can result in lower software costs associated with the simulation project.

When the advantages and disadvantages of the simulation languages mentioned above are considered together, it becomes clear that the features expected from a good simulation software should be critically examined.

5.1. Features Expected from Simulation Software

In general, the features related to the issues to be considered in simulations for discrete position processes have been examined above. In addition to these basic features, it is possible to list the features expected from an ideal simulation computing package, considering the recommendations made by Low and Haider (1989).

General features:

- Modelling flexibility (renewal of a model with different parameters),
- Ease of developing new models,
- Fast model processing,
- Maximum model size,
- Ability to work on different hardware possibilities.

Animation Feature:

Animation helps the key elements of the simulation to be easily viewed by the user. Other benefits expected from animation can be listed as follows:

- Debugging of the simulation program,
- Demonstrating the validity or negativity of the model,
- Uploading a control logic to the system,
- Ensuring dynamic flow to the system,
- Ensuring users are trained.

Statistical Possibilities:

Since some sort of luck factor plays a role in most real-life problems, it should be possible to use most standard probability distributions in simulation modelling (Poisson, exponential, gamma, etc.). In addition, the frequency distributions obtained at the end of the observations should also be used.

User Support:

The feature mentioned here is a feature that should be emphasized not only for the support that the institution that

Cilt/*Volume 4*, **Sayı**/*Number 2*, Aralık/*December* 2022 **Sayfa**/*Pages* 89-96

implements the software should provide to the user, but also for all software. Starting from this sentence:

- Software introduction seminars,
- Providing periodic technical support,
- Good documentation,
- Free software trials and demo discs are features that can be discussed for good support.

Reporting:

A good simulation software should be able to support such results with graphics and even animations, in addition to necessarily providing basic statistics reflecting the performance of the system it is working on.

5.2. Most Common Simulation Languages

Simulation of discrete events requires a large computational volume. Therefore, it cannot be done by human hands. Transaction volume is not the only constraint for calculations using computers. At the same time, the programming language used is also a constraint. Serious programming knowledge and experience are required to model calculations with a programming language created for general purposes such as C/C++ or Java. If there are no specially created libraries, people with high-level programming knowledge will be needed. Having repetitive patterns in the simulation may in some cases allow the creation of libraries for these programming languages. Such operations can be given as sorting, random number generation and statistics operations. For Java, there is information in some books about such libraries (Rossetti 2008). Considering all these, the creation of special programming languages for simulation operations provides significant convenience and performance advantages.

GPSS (General Purpose System Simulation) (Gordon, 1961) is among the main commercially marketed simulation languages that can be used in operations research problems such as stock control and queuing systems in general. It is mostly developed for tail models. It has been used in education for many years due to IBM's strong influence in the computer industry. Later, this software was replaced by GPPS/H (1977) and GPSS/PC (1988). However, the following simulation languages have also had an intense usage rate.

- SIMAN (SIMulation ANalysis)
- SIMSCRIPT
- SLAM (Simulation Language for Alternative Modeling)
- INSIGHT
- MODSIM
- SIM++

Such languages above can be listed. In addition to the previous list, the list below can be given for the simulation of production systems:

- AutoMod
- ProModel
- SIMF ACTORY
- WITNESS
- XCELL +

Also the following languages are developed for network applications:

- NETWORK
- COMNET

6. GPSS (GENERAL PURPOSE SYSTEM SIMULATION)

GPSS, first developed by IBM in 1962, is a discrete system simulation system with block-oriented and block diagram and flowchart equivalents. After the first version of the system, different versions have been released. It is suitable for the programming system developed based on total events and total time. Objects in the system are called Transactions in GPSS. These are created or destroyed by certain blocks. Apart from the standard dumps and reports of GPSS programming, results can be created with certain blocks in line with the user's request. In the application part of this study, GPSS simulation package software was used.

6.1. Application

The queuing problem planned by (Korukoğlu, 1995) was simulated using the Macros of the GPSS simulation package software:

Televisions that have been produced in a factory arrive at the control station every 5.5 ± 2 minutes from the previous station. The control process is carried out in two stages and in parallel, and this process is completed in 9 ± 3 minutes. 85% of the televisions inspected are found to be faultless. Those found to be faulty are sent to the packaging service, and the 15% found to be defective are sent to the relevant service for readjustment. There is also a single employee in this service, each control process is carried out in 30 ± 10 minutes and the adjusted televisions are sent to the control service for rechecking. Since it is known that queues may form in front of both the control and adjustment services, let's plan the 8-hour simulation of the system with GPSS:

Simulation Application with GPSS:

```
*STORAGE CAPACITY DEFINITION
STORAGE S$TEST, 2

*MODEL SEGMENT
GENERATE 55, 20
BACK QUEUE AREA1
ENTER TEST

DEPART AREA1
ADVANCE 90, 30 LEAVE TEST3
TRANSFER 0.15 ,, FIX
TERMINATE
FIX QUEUE AREA2
SEIZE FIXER
DEPART AREA2
ADVANCE 300, 100
RELEASE FIXER
TRANSFER, BACK

*MODEL SEGMENT 2
GENERATE 4800
TERMINATE 1
CONTROL CARDS
START 1
END
```

With the macro commands of the GPSS indexed above, the process presented as a problem in the application is simulated. As you can see, simulating a system that seems complex can make it sufficient to write a maximum of 20 or 30 lines of macro code. However, when we want to implement the same software with computer programming languages, hundreds of lines of code may need to be written. However, for simulation applications written in programming languages, it is a separate problem to determine at what level the results are sufficient and valid. However, the results produced by the macro codes are largely sufficient and valid.

7. CONCLUSION AND SUGGESTIONS

As can be easily seen from the explanations made, simulation modelling is one of the most important analysis methods used in the recognition of systems in most real-life problems. The use of these methods is inevitable, especially in systems that cannot be physically realized. It is undeniable that the mentioned simulation software and their constantly developed new versions are effective in this widespread use. This efficiency of simulation languages causes them to be superior to general-purpose programming languages. However, it should not be forgotten that general-purpose languages also have a special place in solving queuing systems with special demand distribution and similar models, as explained in the sections in the study.

The most important result that emerged in the discussion of the advantages and disadvantages of simulation languages is that it would be appropriate to support these languages with general-purpose programming languages in the solution of simulation scenarios, thus allowing users to use their own abilities according to the application area. It is expected that new versions of simulation languages will be created with this feature.

REFERENCES

Atıl H. (1980), Şans Sayıları Türetme Yöntemleri, Yüksek Lisans Tezi, Ege Üniversitesi Elektronik ve Hesap Bilimleri Enstitüsü, Bornova-İzmir.

Aydın, Ö., & Dalkılıç, G. (2018, July). A hybrid random number generator for lightweight cryptosystems: xorshiftLplus. The 3rd International Conference on Engineering Technology and Applied Sciences (ICETAS).

Cabuk, U. C., Aydin, Ö., & Dalkiliç, G. (2017). A random number generator for lightweight authentication protocols: xorshiftR+. Turkish Journal of Electrical Engineering and Computer Sciences, 25(6), 4818-4828.

Cohen, S. S (1985), Operational Research. Edward Arnold, London.

Gates, B. (1999), Düşünce Hızında Çalışmak, (Çeviren, Ali Cevat Akkoyunlu), Doğan Kitap, İstanbul.

Hillier F.S ve Lieberman, G. J (1990), Introduction to Operations Research Mc Graw Hill Inc. 5. Ed. NewYork.

Kaya, A. (2000), Bir Üretim Sürecinin Simülasyon Yaklaşımı ile Çözümü Üzerine Bir Araştırma, D.E.Ü İİBF Dergisi, Cilt: 15, Sayı:1 Yıl:2000, 115-123.

Kaya, A. (2015), "Bilişim Eğitiminde Değişim Yönetimi", International Conference on Research in Education and Science (ICRES), April 23-25, 2015, Antalya, Turkey.

Korukoğlu, S. (1995), Simülasyon ve Modelleme Ders Notları, Ege Üniversitesi Bilgisayar Mühendisliği Bölümü, Bornova- İzmir.

Kösemen, C., Aydın, Ö., & Dalkılıç, G. (2018a, September). The Pseudorandom Number Generator Generation Method with Genetic Programming for Lightweight Devices. In 2018 3rd International Conference on Computer Science and Engineering (UBMK) (pp. 17-

21). IEEE.

Kösemen, C., Dalkiliç, G., & Aydin, Ö. (2018b). Genetic programming-based pseudorandom number generator for wireless identification and sensing platform. Turkish Journal of Electrical Engineering and Computer Sciences, 26(5), 2500-2511.

Law, A.M ve Kelton, W.D. (1991), Simulation, Modelling and Analysis, Mc Graw-Hill. Inc N.Y.

Law, A.M ve Haider, S. W (1989), Selecting Simulation Software for Manufacturing Applications. Ind. Eng., 31. 33-46 Peppard, J. (1993), IT strategy for Business, Pitman Publishing, New York.

Rossetti, M. D. 2008. "JSL: An Open-Source Object-Oriented Framework for Discrete-Event Simulation in Java." International Journal of Simulation and Process Modeling 4 (1): 69–87.