CHAPTER 3

MODEL DEVELOPMENT

INTRODUCTION

This chapter demonstrates and validates the generic simulation modelling methodology that is conceptualised in Chapter 2 by applying the generic methodology to develop two simulation models of the Synthetic Fuel plant in two different simulation software packages.

The first section investigates three simulation software packages that were considered during this research as candidates for the development of a simulation model of the Synthetic Fuel plant. The three candidates are Arena, Simul8 and Extend. Unfortunately, Extend was excluded from the list of candidates because it was impossible to determine its compatibility with the specific requirements. In the end it was decided to develop simulation models of the Synthetic Fuel plant in Arena and Simul8.

In the second section a simulation model breakdown is derived from the system description of the Synthetic Fuel plant. The simulation model breakdown provides guidelines for the development of the Arena and Simul8 simulation models. The 28 points of evaluation of the Synthetic Fuel plant are identified and they are divided into three types, namely: primary, secondary and tertiary points of evaluation. The 21 primary and secondary points of evaluation are represented in the "real" part of the simulation model by 21 instances of the four different high-level building blocks of the ERM method, while the seven tertiary points of evaluation are accommodated in the "virtual" part of the simulation model by the logic engine high-level building block. The points of evaluation are also classified as either primary or secondary "bottlenecks".

The third section describes the development of two identical simulation models of the Synthetic Fuel plant in Arena and Simul8. The structure of the simulation models is based on the simulation model breakdown of the Synthetic Fuel plant that is discussed in the previous paragraph. In each of the simulation software packages the five high-level building blocks of the generic simulation modelling methodology are developed and then used to construct the simulation models. The primary and secondary points of evaluation are accommodated by the four different high-level building blocks of the ERM method. The tertiary points of evaluation and all the concepts that are necessary for the simulation model to function are accommodated by the logic engine high-level building block. The simulation models use input and output files and spreadsheet variables as input and output mechanisms. They also use two hierarchical levels to represent the Synthetic Fuel plant.

In the fourth section an appropriate iteration time interval for the simulation models of the Synthetic Fuel plant is determined. The results from a series of simulation runs (conducted with the Simul8 simulation model) are interpreted and the assumption that a one hour iteration time interval should be appropriate is substantiated. It is also indicated that the simulation runtime of the Simul8 simulation model with an iteration time interval of one hour represents a twentyfold improvement over the simulation runtime of the original simulation model with an iteration time interval of one hour.

Two possible techniques to determine minimum sufficient sample size are discussed in the fifth section and one of the techniques is identified as the appropriate one to use in this instance. A FORTRAN software programme that determines the minimum sufficient sample size is detailed. The name of the programme is N.FOR and an example of its use is provided.

The sixth section discusses and demonstrates some of the verification and validation concepts of the Arena and Simul8 simulation models with examples. The first example demonstrates the verification of the simulation models and indicates that the simulation models operate as intended, insofar as the number of failures created is concerned. In the second example the simulation models are validated by comparing the mean output throughput values of the Gas Production plant of the simulation models with the mean output throughput value of the Gas Production plant during the 1993 production year. The results indicate deviations of less than 1% from the 1993 production year and therefore the simulation models can be accepted as valid representations of the Synthetic Fuel plant. A sensitivity analysis confirms that the simulation models are not overly sensitive to variation in the start times of the service cycles. Confidence intervals for the results are also determined.

The Arena and Simul8 simulation models are enhanced by the inclusion of an additional evaluation method option in the seventh section. With this enhancement the simulation models now make provision for two different evaluation method options, namely: an iteration time interval (ITI) evaluation method option and an event-driven (ED) evaluation method option. The ED evaluation method option evaluates the simulation models only when an event takes place and not every time interval like the ITI evaluation method option. The concept of event density is introduced and it is indicated that the event density value of a simulation model can be used to determine which of the evaluation method options is appropriate for that specific application. Simulation runs are completed with the ED evaluation method option simulation models and the simulation models are validated by comparing the mean output throughput values of the Gas Production plant of the simulation models with the mean output throughput value of the Gas Production plant during the 1993 production year. The results indicate deviations of less than 1% from the 1993 production year and therefore the ED evaluation method option simulation models can be accepted as valid representations of the Synthetic Fuel plant. The evaluation methods are also compared and their strengths and weaknesses are discussed.

In the last section the ED evaluation method option Arena and Simul8 simulation models and the Arena and Simul8 simulation software packages are compared. An important result that follows from the simulation model comparison is that the simulation runtimes of the ED evaluation method option simulation models represent an approximate fortyfold improvement over the simulation runtime of the original simulation model. The strengths and weaknesses of the two simulation software packages are also discussed.

* * * * *

3.1 INVESTIGATION OF THE SIMULATION SOFTWARE PACKAGES

The generic simulation modelling methodology is conceptualised in Chapter 2 and in the last section the structure of the generic methodology is developed. It is indicated that the generic methodology is divided into two separate parts. The first is the iterative-loop technique part that determines the governing parameters before the start of a simulation run and the second is the simulation model part that uses six methods and techniques continuously during the simulation run. The six methods and techniques are encapsulated in five high-level building blocks that can be used to construct simulation models of stochastic continuous systems.

In Section 2.5 the iterative-loop technique is detailed and a FORTRAN software programme called PSCALC.FOR is used to determine the governing parameters of the Synthetic Fuel plant for the system description that is provided in Section 1.2. This chapter is primarily concerned with the simulation model part of the generic simulation modelling methodology.

The first obvious step is to identify candidates from the available simulation software packages that could be used to develop a simulation model of the Synthetic Fuel plant. A superficial perusal of the possible candidates revealed three simulation software packages worthy of in-depth scrutiny.

The three candidates are the following:

a) Arena
b) Simul8
c) Extend

Extend is a trademark and is usually denoted by Extend™. However, for the sake of simplicity it will be written simply as Extend in this document. Extend is a simulation software package from Imagine That, Inc.

The inclusion of Arena in the shortlist follows logically from the fact that the final 1996 simulation model (that included the whole Sasol Synfuels complex) was upgraded to one of the first versions of Arena (see Section 1.4). Therefore it seems a logical point of departure to use Arena for the development of a simulation model that demonstrates the use of the generic simulation modelling methodology. An important factor is also that the Arena Standard Edition simulation software package was available for this research. Arena is an accomplished and widely accepted simulation software package.

Simul8 was introduced as a contender when the Simul8 Standard simulation software package was made available for the research. Simul8 is a relative “newcomer” to the simulation software package fraternity and it was concluded that it would be a worthwhile exercise to determine its prowess with this challenging application.

Extend was perceived to be a possible contender because of its claims in terms of continuous modelling ability. A demonstrator version of Extend was procured and evaluated. Unfortunately it was very difficult to adequately fathom the capabilities of Extend because the demonstrator version is severely restricted. For example, a modeller is only allowed to build simulation models that contain up to 25 blocks and the save function has been disabled. These restrictions made it virtually impossible to determine without doubt whether the simulation model part of the generic simulation modelling methodology could be developed in Extend and consequently Extend was disqualified as a contender after the in-depth scrutiny of the simulation software packages. It is worthwhile to note that Imagine That, Inc. responded very quickly (less than one month for the package to arrive by post) to the request for the demonstrator version of Extend and that the Extend user’s guide is exemplary among its peers.

It was therefore decided to use the high-level building blocks of the simulation model part of the generic simulation modelling methodology to develop two identical simulation models of the Synthetic Fuel plant in two different simulation software packages, namely: Arena and Simul8.

Two simulation models were built to illustrate clearly that the generic simulation modelling methodology is not based on, or restricted to, a specific simulation software package.

Summary

This section discusses three different simulation software packages that were considered to develop a simulation model of the Synthetic Fuel plant. The three candidates are the following: Arena, Simul8 and Extend. Extend was disqualified from the list of candidates because it was impossible to determine its compatibility with the requirements from the demonstrator version. It was finally decided to develop simulation models of the Synthetic Fuel plant in Arena and Simul8.

* * * * *

3.2 SIMULATION MODEL BREAKDOWN

Before a simulation model can be constructed, it is necessary to develop a simulation model breakdown of the system that is being modelled. From the system description of the Synthetic Fuel plant that is provided in Section 1.2 and Table A1 it is apparent that the total plant consists of 20 smaller plants, or alternatively, 21 smaller plants if the extra oxygen "train" is considered as a separate smaller plant. Some of the smaller plants consist of groupings of different types of modules, namely: the Oxygen plant (three types of modules), the Oxygen Extra plant (three types of modules), Plant(II) (two types of modules) and Plant(IV) (three types of modules). That implies that there are actually 28 points of evaluation in the Synthetic Fuel plant. The points of evaluation can be ranked into three levels of evaluation in terms of their importance.

The three levels of importance (or types of evaluation points) are the following:

a) Primary points of evaluation.
b) Secondary points of evaluation.
c) Tertiary points of evaluation.

The primary points of evaluation are the points of evaluation in the smaller plants that influence the throughput of the Synthetic Fuel plant directly and that are also subject to services and failures of their modules. The second rule of operation in Appendix B states that the smaller plants that form part of the main-gas-cycle influence the throughput of the Synthetic Fuel plant. There are 10 smaller plants and 13 points of evaluation in the main-gas-cycle. These smaller plants are sometimes referred to as the "heart" of the Synthetic Fuel plant. The 13 primary points of evaluation are Coal Processing, Steam, Gas Production, Temperature Regulation, Oxygen-A, -B and -C, Plant(I), Plant(II)-A and -B, Plant(III), Division Process and Recycling. These 13 primary points of evaluation can act as primary "bottlenecks" and the two "bottleneck" identification techniques that are developed in Section 2.6 are used to prioritise them. If the extra oxygen "train" is also considered, it ads another three points of evaluation, namely: Oxygen Extra-A, -B and -C. Oxygen Extra-A, -B and -C cannot act as primary "bottlenecks" because their output throughput is added to that of Oxygen-A, -B and -C respectively, if the extra oxygen "train" is included in the simulation run. In total there are thus 16 primary points of evaluation.

The secondary points of evaluation are the points of evaluation in the smaller plants that do not influence the throughput of the Synthetic Fuel plant directly but that are subject to services and failures of their modules. The third rule of operation in Appendix B states that the Electricity Generation plant, Plant(IV), Plant(V) and Sub(I) to Sub(VI) do not form part of the main-gas-cycle and therefore they do not influence the throughput of the Synthetic Fuel plant directly. These smaller plants are referred to as the peripheral plants. However, a scrutiny of Table A2 reveals that Sub(I) to Sub(VI) are not subject to services and failures of their modules and therefore they are excluded from the secondary points of evaluation. That leaves five secondary points of evaluation, namely: the Electricity Generation plant, Plant(IV)-A, -B and -C and Plant(V). The fourth rule of operation in Appendix B states that if Plant(IV), Plant(V) and Sub(I) to Sub(VI) do not have the capacity to process the throughput at their respective positions, then the portions of the throughput that cannot be processed are flared. Once again Sub(I) to Sub(VI) are excluded because they are not subject to services and failure of their modules. The five secondary points of evaluations can act as secondary “bottlenecks” and therefore the portions of the throughput that are flared at Plant(IV) and Plant(V) are determined to ensure that these secondary “bottlenecks” can be identified and prioritised.

The tertiary points of evaluation are the points of evaluation in the smaller plants that do not influence the throughput of the Synthetic Fuel plant directly and that are also not subject to services and failure of their modules. From the previous paragraph it follows that Sub(I) to Sub(VI) qualify. The Water Treatment plant also qualifies because its modules are not subject to services and failures and even though it actually forms part of the main-gas-cycle it never influences the throughput of the Synthetic Fuel plant (see Points b) and f) of the rules of operation in Appendix B). That gives a total of seven tertiary points of evaluation, namely: the Water Treatment plant and Sub(I) to Sub(VI). Sub(I) to Sub(VI) can act as secondary “bottlenecks”. Therefore the portions of the throughput that are flared at Sub(I) to Sub(VI) are determined to ensure that these secondary “bottlenecks” can be identified and prioritised.

It is obvious that the primary and secondary points of evaluation have to be represented in the “real” part of the simulation model by the four different high-level building blocks of the ERM method because they are subject to services and failures of their modules. That gives a total of 21 ERM method high-level building blocks (16 for the primary points of evaluation if the extra oxygen “train” is included and five for the secondary points of evaluation). The seven tertiary points of evaluation are accommodated in the “virtual” part of the simulation model by the logic engine high-level building block.

The 13 primary points of evaluation that are left after Oxygen Extra-A, -B and C have been excluded are included in the FC method and they also make up the primary “bottlenecks”. The secondary and tertiary points of evaluation that flare excess throughput make up the secondary “bottlenecks”.

The previous paragraphs are summarised in tabular format in Table 3.1: *Simulation Model Breakdown*.

Table 3.1: Simulation Model Breakdown

No.	Name	POE No.	POE Type	ERM Method Block No.	"Bottleneck" Type
1	Coal Processing	1	Primary	1	Primary
2	Water Treatment	2	Tertiary	(Logic Engine)	-
3	Steam	3	Primary	2	Primary
4	Gas Production	4	Primary	4	Primary
5	Temperature Regulation	5	Primary	2	Primary
6-A 6-B 6-C	Oxygen-A Oxygen-B Oxygen-C	6 7 8	Primary Primary Primary	2 2 2	Primary Primary Primary
6E-A 6E-B 6E-C	Oxygen Extra-A Oxygen Extra-B Oxygen Extra-C	9 10 11	Primary Primary Primary	3 3 2	- - -
7	Electricity Generation	12	Secondary	2	-
8	Plant(I)	13	Primary	2	Primary
9-A 9-B	Plant(II)-A Plant(II)-B	14 15	Primary Primary	1 2	Primary Primary
10	Plant(III)	16	Primary	4	Primary
11	Division Process	17	Primary	4	Primary
12	Recycling	18	Primary	3	Primary
 13-A 13-B 13-C	Tank Plant(IV)-A Plant(IV)-B Plant(IV)-C	- 19 20 21	- Secondary Secondary Secondary	- 4 4 4	- Secondary (Flare-A) Secondary (Flare-A) Secondary (Flare-A)
14	Sub(I)	22	Tertiary	(Logic Engine)	Secondary (Flare-C1)
15	Sub(II)	23	Tertiary	(Logic Engine)	Secondary (Flare-C2)
16	Sub(III)	24	Tertiary	(Logic Engine)	Secondary (Flare-C3)
17	Sub(IV)	25	Tertiary	(Logic Engine)	Secondary (Flare-C4)
18	Sub(V)	26	Tertiary	(Logic Engine)	Secondary (Flare-C5)
19	Sub(VI)	27	Tertiary	(Logic Engine)	Secondary (Flare-C6)
20	Plant(V)	28	Secondary	4	Secondary (Flare-B)

Where:

No.	:	The plant identification number.
POE No.	:	The point of evaluation number.
POE Type	:	The point of evaluation type.

The numbers in Column 5 of Table 3.1 indicate which one of the four different high-level building blocks of the ERM method is needed at each of the primary and secondary points of evaluation.

The numbers that identify the four different high-level building blocks of the ERM method are the following:

a) No.1 - A smaller plant with a multiple service cycle and failures of the modules.
b) No.2 - A smaller plant with a service cycle and failures of the modules.
c) No.3 - A smaller plant with a service cycle of the modules.
d) No.4 - A smaller plant with failures of the modules.

Summary

This section provides a simulation model breakdown of the Synthetic Fuel plant. The breakdown is derived from the system description. The 28 points of evaluation of the Synthetic Fuel plant are divided into three types, namely: primary, secondary and tertiary points of evaluation. The 21 primary and secondary points of evaluation are represented in the "real" part of the simulation model by 21 instances of the four different high-level building blocks of the ERM method and the seven tertiary points of evaluation are accommodated in the "virtual" part of the simulation model by the logic engine high-level building block. The points of evaluation that form part of the FC method and that are primary "bottlenecks", as well as the secondary "bottlenecks" that flare excess throughput, are identified.

* * * * *

3.3 SIMULATION MODEL CONSTRUCTION

Section 3.1 indicates that it was decided to develop two identical simulation models of the Synthetic Fuel plant in two different simulation software packages, namely: Arena and Simul8. Section 3.2 provides a simulation model breakdown of the Synthetic Fuel plant and this section details the Arena and Simul8 simulation models.

In both the Arena and Simul8 simulation modelling environments the first step was to develop the five high-level building blocks of the generic simulation modelling methodology. (The four different high-level building blocks of the ERM method are detailed in Section 2.3 and the logic engine high-level building block is detailed in Section 2.7.) Each high-level building block is constructed from several basic simulation software package building blocks in the respective simulation software packages. The way that the high-level building blocks manifest themselves in the two different simulation software packages differs because each software package has its own unique philosophy, conventions, logic, nomenclature, *etc.* This is especially true for the logic engine high-level building block that is constructed mainly from basic simulation software package building blocks in the Arena environment, but in the Simul8 environment it consists primarily of a block of Visual Logic (VL) code. The following quotation from the *Simul8®: Manual and Simulation Guide* (1999:29) explains what VL is and how it is used in a simulation model:

> "*Visual Logic (VL) is Simul8's logic building environment. In a simulation of significant complexity you will want to add your own rules for deciding how to process work. VL lets you add very detailed logic to control the operation of your simulation.*"

The four different high-level building blocks of the ERM method accommodate the primary and secondary points of evaluation and are all based on the basic structure of the three separate parts of each of the smaller plants that is shown in Figure 2.1. The basic structure is simply adapted to suit the needs of each of the four different high-level building blocks of the ERM method. The logic engine high-level building block accommodates the tertiary points of evaluation and all the concepts that are necessary for the simulation model to function (see Figure 2.4). The five high-level building blocks represent the "virtual" part (*i.e.* the logic engine high-level building block) and the "real" part (*i.e.* the four different high-level building blocks of the ERM method) of the simulation model (see Figure 2.4).

The five high-level building blocks of the Arena environment were used to develop a simulation model of the Synthetic Fuel plant in the Arena environment and the five high-level building blocks of the Simul8 environment were used to develop a simulation model of the Synthetic Fuel plant in the Simul8 environment. The simulation model of the Synthetic Fuel plant in the Arena environment is referred to as the Arena simulation model and the one in the Simul8 environment is referred to as the Simul8 simulation model in the rest of this document. Both the Arena and Simul8 simulation models consist of two No.1 ERM method high-level building blocks, nine No.2 ERM method high-level building blocks, three No.3 ERM method high-level building blocks, seven No.4 ERM method high-level building blocks and one logic engine high-level building block (see Table 3.1). That is a total of 21 ERM method high-level building blocks and one logic engine high-level building block in each of the simulation models. The Arena and Simul8 simulation models are identical in the sense of conforming to exactly the same system description (see Section 1.2) but they differ in terms of the construction of the high-level building blocks (as explained previously in this section).

The high-level building blocks of each of the four different types of high-level building blocks of the ERM method are truly generic because all the high-level building blocks of a specific type are absolutely identical except for the modules that populate them. Each high-level building block of the ERM method is populated with the correct number of entities that represents the modules of the Synthetic Fuel plant. The relevant information about each module is stored in the attributes of the entity that represents the module.

To a large extent, the logic engine high-level building block is generic because most of the concepts that are necessary for the simulation model to function are basically the same for every simulation model that is developed with the generic simulation modelling methodology. However, the unique concepts of a specific simulation model that are usually described by the process logic or rules of operation of that specific simulation model cannot be accommodated generically and therefore a part of the logic engine high-level building block of that specific simulation model will contain certain concepts that are unique to that specific simulation model. For instance, Point g) of the rules of operation of the Synthetic Fuel plant in Appendix B states that steam is only supplied to the Electricity Generation plant once the Gas Production and Oxygen plants have been supplied. It is virtually impossible to make provision to accommodate all possible combinations and permutations of such rules of operation generically in the logic engine high-level building block. Other concepts, like the inclusion of a tank to buffer flow, are more universal and therefore lend themselves more readily to generic use.

The logic engine high-level building block controls the functioning of the simulation model. Before the start of the simulation run the ERM method high-level building blocks are populated with the corresponding correct number of entities that represent the modules and appropriate values are assigned to the attributes of the entities (*i.e.* the modules). In the Arena simulation model this process is handled by the logic engine (*i.e.* centralised populating) but in the Simul8 simulation model this process is handled by the ERM method building blocks themselves (*i.e.* decentralised populating).

The three main tasks (already touched upon in Section 2.7) of the logic engine high-level building block are the following:

a) Before the start of the simulation run the logic engine sets up the simulation model and populates the ERM method high-level building blocks with entities (in the case of the Arena simulation model). The simulation model is set up with input values that reflect the system description of the scenario that is under scrutiny. The input values are accessed with the appropriate input mechanisms of the Arena and Simul8 simulation models.
b) During the simulation run the logic engine uses the fixed time interval technique to advance the simulation model in time. Every time interval an evaluation of the state of the simulation model takes place and the logic engine completes all the tasks that are necessary for the simulation model to function. The tasks that are completed by the logic engine during every evaluation are indicated in Figure 3.1: *Tasks of the Logic Engine (Every Evaluation)*.
c) After the completion of the simulation run the logic engine prepares the results and writes it to the appropriate output mechanisms of the Arena and Simul8 simulation models. (The results that follow from a simulation run are detailed in Section 4.1.)

Figure 3.1 indicates the detail and the sequence of the tasks that are completed by the logic engine during every evaluation and which are described in a more generic and less detailed format in Section 2.7.

Both the Arena and Simul8 simulation models use the theoretical probability distributions that are provided in the respective simulation software packages to model the failure rates and repair times of the modules (see Section 1.2). The failure rates are modelled with the exponential distribution and the repair times with the triangular distribution (see Section 1.2 and Table A2).

Tasks of the Logic Engine (Every Evaluation)

Determines Evaluation Start Variables that Keep Record of Functioning of Simulation Model

Determines Accumulated Values (With System Behaviour of Previous Time Interval):
a) Primary POE: Throughput, "Bottleneck" Time and Production Lost (Primary "Bottlenecks"), Number Available, Switched On, Switched Off
b) Secondary POE: Throughput, Tank Volume, Flares (Secondary "Bottlenecks"), Number Available, Switched On, Switched Off
c) Tertiary POE: Throughput, Flares (Secondary "Bottlenecks")

Controls ERM Method (For Each ERM Method Building Block):
a) Evaluates Next Service Attribute of Each Module in Availability Part of ERM Method Building block
b) Moves Due for Service Modules from Availability Part to Service Part of ERM Method Building Block
c) Evaluates Next Failure Attribute of Each Module in Availability Part of ERM Method Building Blocks
d) Moves Failed Modules from Availability Part to Service Part of ERM Method Building Block

Determines Values (For System Behaviour of Next Time Interval):
a) Determines Number of Available Modules in Each Primary and Secondary POE
b) Uses FC Method to Determine Momentary "Bottleneck" and Actual Output Throughput of Primary POE
c) Determines Actual Output Throughput of Secondary POE

Determines Evaluation End Variables that Keep Record of Functioning of Simulation Model

Figure 3.1: Tasks of the Logic Engine (Every Evaluation)

The Arena simulation model uses input files to provide access to, and manipulation of, the most important aspects of the system description of the Synthetic Fuel plant that is provided in Section 1.2. For instance, the service schedules are addressed in an input file called SERVIC.DAT. An example of SERVIC.DAT is provided in Appendix E: *SERVIC.DAT (Arena Simulation Model Service Schedules Input File)*. This example contains the input values for the service schedules of the smaller plants of the Synthetic Fuel plant that are detailed in Section 1.2 and Table A2. A scrutiny of SERVIC.DAT reveals that it bears a close resemblance to the part of Table A2 that addresses the service schedules of the smaller plants. Each of the smaller plants that is subjected to services is represented by a header line that identifies the smaller plant and one (for a regular service cycle) or more (for a multiple service cycle) lines of three values each. The first value of each line represents the start time of the first service of the first service cycle, the second value represents the cycle time and the third value represents the service time. The way that the service schedule values are used to control the services is detailed in Section 2.3. The determination of the start times is detailed in Section 3.6. The input files are manipulated with a text editor.

The Arena simulation model uses WKS files as the output mechanism for the results that are generated by a simulation run. The following excerpt from the Arena help function explains what a WKS file is:

> "*The worksheet format, specified by the WKS File keyword, refers to a binary, sequential access data structure used by LOTUS™ spreadsheets. Numeric values can be read from or written to these files to facilitate data collection or analysis using LOTUS™ products. Worksheet files are sequential access only.*"

An example of a WKS output file is shown in Appendix F: *PRIORI.WKS (Arena Simulation Model "Bottleneck" Identification Output File*. Each line of values represents the results of one of the replications that was completed during the simulation run. Kelton *et al.* (1998:36) defines replications as identical, independent simulation runs.

> "*Each run starts and stops the same way and uses the same input-parameter settings (that's the "identical" part), but uses separate input random numbers (that's the "independent" part) to generate the interarrival and service times.*"

Kelton *et al.* use the term "simulation run" to define replications as identical, independent simulation runs, but in this document the term "simulation run" is used exclusively to indicate a complete simulation experiment that usually consists of more than one replication of a simulated scenario.

The first value in each line identifies the replication and the following 13 values in each line represent the possible throughput that was lost (as a percentage of the steady state maximum possible throughput) due to each of the 13 possible "bottleneck" points in the main-gas-cycle. This example shows the results of a simulation run that comprises 20 replications. The WKS output files can easily be imported into Microsoft Excel or Quattro Pro for further manipulation and output analysis (see Section 4.1).

Microsoft Excel and Quattro Pro are registered trademarks and are usually denoted by Microsoft® Excel and Quattro® Pro respectively. However, for the sake of simplicity they will be written simply as Microsoft Excel and Quattro Pro in this document. Microsoft Excel is a spreadsheet software package from the Microsoft Corporation and Quattro Pro is a spreadsheet software package from Corel®.

The Simul8 simulation model uses spreadsheet variables as the input and output mechanisms of the simulation model. In Simul8 every variable that is used by the simulation model is defined in the *Information Store*. A variable is called a *Global Data Item* and may be defined as a spreadsheet. This is a very useful feature because it allows easy manipulation of variables and simplifies the import and export of values into and out of the simulation model. For example, the values that define the service schedules of the Synthetic Fuel plant can be arranged in either a Microsoft Excel or a Quattro Pro spreadsheet and are then simply copied into the Simul8 simulation model after manipulation to reflect the system description of the scenario that is under scrutiny. This process can be simplified even more by instructing the Simul8 simulation model to automatically read the service schedules from a Microsoft Excel spreadsheet when the simulation run starts. The problem with this technique is that the appropriate Microsoft Excel file has to be open and therefore it restricts the amount of Random Access Memory (RAM) that is available to the Simul8 simulation software package during the execution of the simulation run and adversely affects the simulation runtime.

The input files and WKS output files of the Arena simulation model and the spreadsheet variables of the Simul8 simulation model greatly simplify the manipulation of input and output variables and therefore they enhance the user-friendliness of the simulation models. These concepts also support the user-friendliness design criterion (see Point c) of the design criteria in Section 1.5) of the generic simulation modelling methodology.

Both the Arena and Simul8 simulation models use two hierarchical levels to represent the Synthetic Fuel plant. The use of hierarchical levels in simulation models ensures that the simulation models are logical, structured and orderly. The higher hierarchical level of both the Arena and Simul8 simulation models consists of 21 ERM method high-level building blocks and one logic engine high-level building block. On the higher hierarchical level each instance of the five high-level building blocks of the generic simulation modelling methodology is represented as a singular entity. Such an entity is referred to as a *submodel* in the Arena environment and as a *sub-window* in the Simul8 environment. The content of the *submodels* and *sub-windows* represents the next or lower hierarchical level. The lower hierarchical level of both the Arena and Simul8 simulation models consists of the basic simulation software package building blocks of the Arena and Simul8 simulation software packages respectively.

The higher hierarchical level *submodels* and *sub-windows* are arranged in the simulation windows of the Arena and Simul8 simulation software packages in such a way that the layout of the *submodels* and *sub-windows* conforms closely to the configuration of the Synthetic Fuel plant that is represented in Figure 1.2. (The simulation windows are the main representations of the simulation models within the simulation software packages.) The realistic representation of a simulation model in a layout or configuration that is immediately recognisable is fundamental to the successful familiarisation with, orientation to, and acceptance of, the simulation model by clients and users (Elder, 1992:150-153).

Appendix G: *Simulation Window of the Higher Hierarchical Level (Simul8 Simulation Model)* shows the higher hierarchical level simulation window of the Simul8 simulation model. In the top left of the simulation window the 21 ERM method high-level building blocks are arranged in a layout that conforms to the configuration of the Synthetic Fuel plant that is depicted in Figure 1.2. In the bottom left of the simulation window are the logic engine and animation engine high-level building blocks. The animation engine controls the animation of the Simul8 simulation model.

The animation concepts that are controlled by the animation engine are the following:

a) The graphical representation of the output throughput of the Gas Production plant of the Synthetic Fuel plant over time as a graph in the bottom centre of the simulation window.
b) The animation of the momentary "bottleneck" status of the 13 possible "bottleneck" points in the main-gas-cycle over time with a grey or a red dot above the icon of the appropriate possible "bottleneck" point (a red dot signifying that the possible "bottleneck" point is the momentary "bottleneck" at that specific moment in time).
c) The animation of the flares at Plant(IV) and Plant(V) over time with a grey or a red flare at the top of the appropriate stack (a red flare signifying that the flare is active at that specific moment in time).

The animation engine is unique to the Simul8 simulation model. The animation features are mostly for demonstration purposes and can be switched off to speed up simulation runtimes when simulation runs are conducted.

The four different high-level building blocks of the ERM method are represented by different icons in the simulation window to facilitate immediate recognition and differentiation. The different icons of the high-level building blocks of the ERM method are identified in the symbol key in the bottom right of the simulation window. The icons of the logic and animation engines are self-explanatory and they are not included in the symbol key.

Appendix H: *Simulation Window of the Lower Hierarchical Level (Arena Simulation Model -*

Example No.1) shows the lower hierarchical level simulation window of one of the 21 ERM method high-level building blocks of the Arena simulation model. This example shows the lower hierarchical level simulation window of the No.1 ERM method high-level building block that represents the Coal Processing plant (*i.e.* a smaller plant with a multiple service cycle and failures of modules). The basic simulation software package building blocks of the Arena simulation software package and the connections between them can clearly be distinguished in the simulation window. The lower hierarchical level simulation windows of the other ERM method high-level building blocks are similar but less complex. The lower hierarchical level simulation window of the logic engine high-level building block of the Arena simulation model contains considerably more basic simulation software package building blocks and is much more complex. This lower hierarchical level simulation window is shown in Appendix I: *Simulation Window of the Lower Hierarchical Level (Arena Simulation Model - Example No.2)*.

Summary

In this section two identical simulation models of the Synthetic Fuel plant are developed in Arena and Simul8. The structure of the simulation models is based on the simulation model breakdown of the Synthetic Fuel plant that is provided in Section 3.2. The five high-level building blocks of the generic simulation modelling methodology were developed in each of the simulation software packages and then used to construct the simulation models. The four different high-level building blocks of the ERM method accommodate the primary and secondary points of evaluation and the logic engine high-level building block accommodates the tertiary points of evaluation and all the concepts that are necessary for the simulation model to function. The Arena simulation model uses input and output files and the Simul8 simulation model uses spreadsheet variables as input and output mechanisms. Both the simulation models use two hierarchical levels to represent the Synthetic Fuel plant. The higher hierarchical level consists of the instances of the high-level building blocks while the lower hierarchical level consists of the basic simulation software package building blocks.

* * * * *

3.4 DETERMINATION OF THE ITERATION TIME INTERVAL

Section 2.2 indicates that a fixed time interval can be used to advance a simulation model in time. Such a fixed time interval to advance a simulation model in time is usually referred to as an iteration time interval. The size of the iteration time interval depends on the required accuracy and is usually chosen in accordance with the dynamic response characteristics of the system that is modelled. If the iteration time interval is chosen correctly, the results that are obtained can be a very close approximation of the real-world situation that is modelled.

In general terms it can be stated that the iteration time interval of a simulation model should be chosen in such a way that it makes provision to accurately register or capture the effect of the shortest event that may occur in the simulation model during a simulation run. The *Magister* dissertation (Albertyn, 1995:64-69) provides a more detailed discussion of this principle.

A scrutiny of the processes of the Synthetic Fuel plant suggests that an iteration time interval of one hour should be appropriate. Table A2 indicates that the shortest service time of the modules in the smaller plants of the Synthetic Fuel plant is one hour for the services of the first service cycle of the Coal Processing plant. Table A2 also indicates that the shortest repair times of the modules in the smaller plants of the Synthetic Fuel plant are those of the Oxygen Extra-C plant, the Electricity Generation plant and Plant(IV)-A. The three values of the triangular distribution that are used to represent the repair times of the Oxygen Extra-C module are 0,5 (minimum), 12 (mode) and 24 (maximum) hours while those of the Electricity Generation plant and Plant(IV)-A modules are 0,25 (minimum), 1 (mode) and 3 (maximum) hours and 0,5 (minimum), 0,5 (mode) and 3 (maximum) hours respectively. Even though smaller values than one are present in these triangular distributions, the modes of the distributions are 12, 1 and 0,5 hours and therefore the assumption that a one hour iteration time interval should be appropriate seems reasonable.

The validity of the assumption that a one hour iteration time interval should be appropriate for simulation models of the Synthetic Fuel plant is tested by conducting a series of simulation runs (*i.e.* simulation experiments) that starts with a very short iteration time interval and gradually increases it, until the answers of the simulation runs start to deviate from the perceived correct one. In this instance, the perceived correct answer will be the one that is generated by the simulation run with the shortest iteration time interval.

Table 3.2: *Effect of the Iteration Time Interval* shows the results if the iteration time interval of the Simul8 simulation model is increased in steps from 0,125 to 24 hours in a series of 10

simulation runs. The Simul8 simulation model was used for this series of simulation runs because the simulation runtimes of the Simul8 simulation model are slightly shorter than those of the Arena simulation model. The input values for the services and failures that were used are those that are represented in Table A2 (service schedules and failure characteristics) and Appendix E (start times of service cycles).

Table 3.2: Effect of the Iteration Time Interval

No.	ITI (hour)	n_{Rep}	Runtime (min)	GasPro (nm^3/h)	StdDev (nm^3/h)	n_{Sam}	Deviation (%)
1	0,125	20	133,9	1331972,2	7185,9	12	0,000
2	0,25	20	67,0	1331894,1	7185,6	12	-0,006
3	0,5	20	33,6	1331780,6	7159,0	12	-0,014
4	1	20	17,0	1331462,8	7154,9	12	-0,038
5	2	20	8,7	1330787,6	7131,7	12	-0,089
6	3	20	5,9	1330159,2	7112,1	12	-0,136
7	4	20	4,5	1329644,9	7153,3	12	-0,175
8	6	20	3,1	1328126,0	7204,7	12	-0,289
9	12	20	1,7	1323017,8	7087,1	12	-0,672
10	24	20	1,0	1309800,0	7781,5	13	-1,665

Where:

No. : The simulation run identification number.

ITI : The iteration time interval (hour).

n_{Rep} : The number of replications completed.

Runtime : The simulation runtime for n_{Rep} replications (minute).

GasPro : The mean output throughput value of the Gas Production plant, calculated from n_{Rep} replications (nm^3/h).

StdDev : The standard deviation from the mean output throughput value (nm^3/h).

n_{Sam} : The minimum sufficient sample size.

Deviation : The deviation of the specific mean output throughput value from the 0,125 hour iteration time interval mean output throughput value (%).

A simulation run consisting of 20 replications of a simulated time period of one year (see Appendix L) was completed for every iteration time interval. The mean and the standard deviation from the mean of the output throughput values of the Gas Production plant are calculated from the results of the 20 replications. The standard deviation is used to calculate the corresponding minimum sufficient sample size with an allowance for a 0,5% deviation from the real-world mean output throughput value of the Gas Production plant (see Appendix M: *Synthetic Fuel Plant Raw Gas Production - 1993*) and a 99% confidence interval. Section 3.5 provides a detailed explanation of the determination of minimum sufficient sample size. The number of replications completed in all instances should be more than, or equal to, the calculated minimum sufficient sample sizes for the answers to be taken as representative of the simulated scenario. A scrutiny of Columns 3 and 7 of Table 3.2 indicates that this constraint is adhered to.

The mean output throughput value of the Gas Production plant is used as the variable of comparison in this series of simulation runs because it is the point in the Synthetic Fuel plant where the coarse coal is transformed into raw gas and the volume of raw gas that is generated determines the final output of the Synthetic Fuel plant.

It is essential to note that one of the benefits of short simulation runtimes immediately becomes apparent when these results are compared to the results of the comparable series of simulation runs that were conducted in the *Magister* dissertation (Albertyn, 1995:64-69). Even though Table 3.4 in the *Magister* dissertation (Albertyn, 1995:66) does not provide the simulation runtimes, it can be deducted from the results presented in Appendix D and E of the *Magister* dissertation (Albertyn, 1995:118-127) that one replication of a simulated time period of one year with an iteration time interval of one hour with the original simulation model, took 17,1 minutes to complete. It can therefore be concluded that a simulation run consisting of 20 replications would have taken approximately 5,7 hours to complete with the original simulation model. If the value of 5,7 hours for a simulation run consisting of 20 replications of a simulated time period of one year with an iteration time interval of one hour for the original simulation model is compared to the value of 17,0 minutes for a simulation run consisting of 20 replications of a simulated time period of one year with an iteration time interval of one hour for the Simul8 simulation model, it is found that the Simul8 simulation model represents a twentyfold improvement in simulation runtime over the original simulation model. This phenomenal improvement in simulation runtime allows the minimum sufficient sample size of the Simul8 simulation model to be determined with an allowance of a 0,5% deviation from the real-world mean output throughput value of the Gas Production plant and a 99% confidence interval, as compared with the 1% deviation from the real-world mean output throughput value of the Gas Production plant and a 99% confidence interval that are used to determine the minimum sufficient sample size of the original simulation model in the *Magister* dissertation (Albertyn, 1995:66). The minimum sufficient sample size in this instance is 12 (see Table 3.2) for the Simul8 simulation model (*i.e.* for an allowance of a 0,5% deviation from the real-world mean output throughput value of the Gas Production plant) and it is five (Albertyn, 1995:76) for the original simulation model (*i.e.* for an allowance of a 1% deviation from the real-world mean output throughput value of the Gas Production plant). Simulation runs of the original simulation model were usually restricted to 10 replications due to the long simulation runtimes and therefore it was impossible to achieve an allowance of only a 0,5% deviation from the real-world mean output throughput value of the Gas Production plant.

The results of Table 3.2 are graphically depicted in Figure 3.2: *Effect of the Iteration Time Interval*.

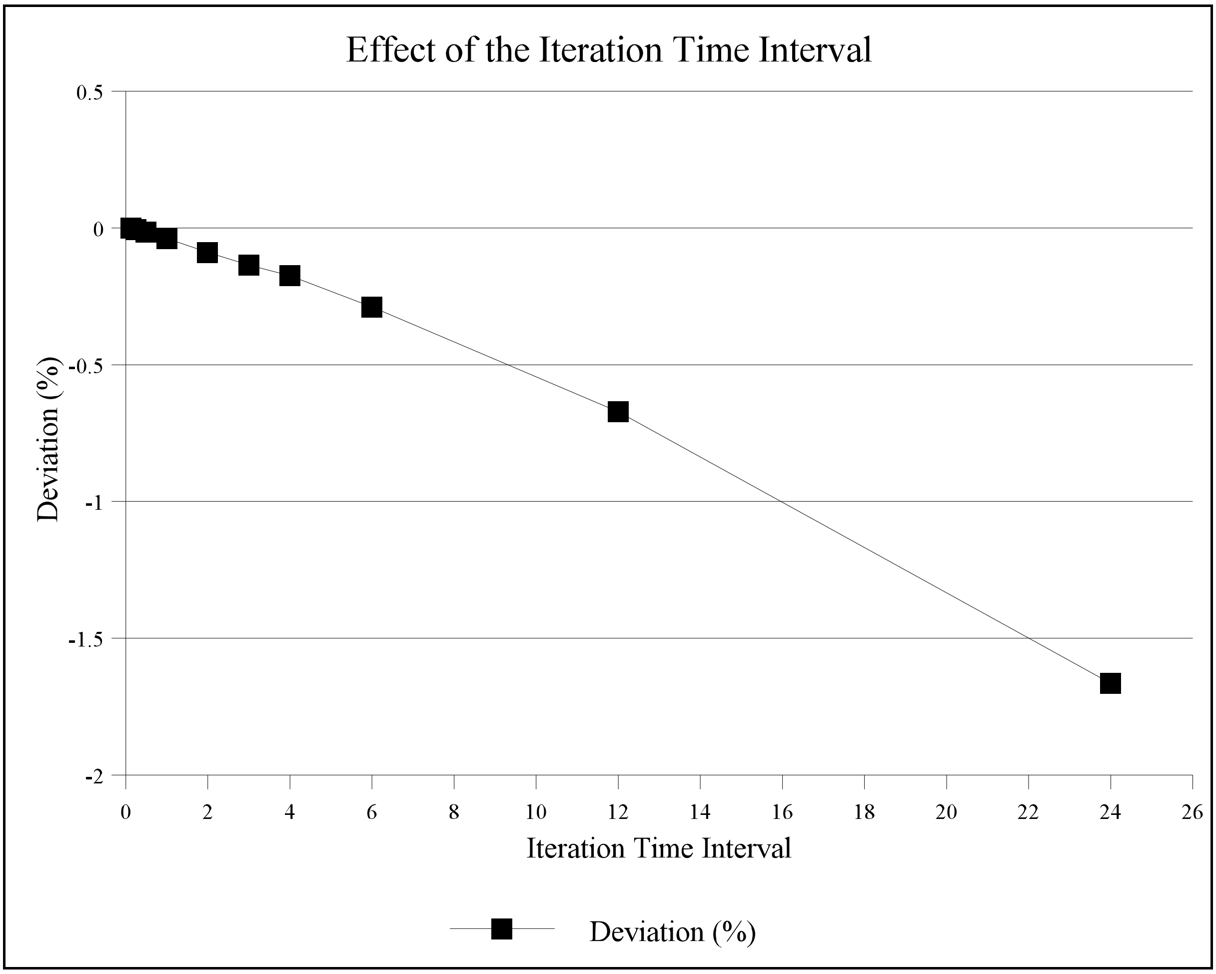

Figure 3.2: Effect of the Iteration Time Interval

A scrutiny of Table 3.2 and Figure 3.2 indicates that the deviation from the perceived correct answer (*i.e.* the one that is generated by the simulation run with the shortest iteration time interval) increases with an increase in the iteration time interval. If a deviation of 0,5% is taken as an acceptable deviation, all iteration time intervals up to and including six hours seem acceptable. The assumption that a one hour iteration time interval should be appropriate for simulation models of the Synthetic Fuel plant is therefore validated by this exercise.

The downward trend in deviation is caused by a fall in the output throughput value of the Gas Production plant if the iteration time interval is increased. This happens because the Synthetic Fuel plant always strives to maintain the maximum possible throughput and would have resumed the maximum possible throughput as soon as possible after the return of a module from service or failure. This return is delayed if the iteration time interval is long. The Synthetic Fuel plant is thus modelled as operating at a lower throughput than that which is actually possible for the remainder of the iteration time interval.

Summary

This section determines an appropriate iteration time interval for the simulation models of the Synthetic Fuel plant. The results from a series of simulation runs are presented and the assumption that a one hour iteration time interval should be appropriate is shown to be realistic. It is furthermore indicated that the simulation runtime of the Simul8 simulation model with an iteration time interval of one hour represents a twentyfold improvement over the simulation runtime of the original simulation model with an iteration time interval of one hour. This huge improvement in simulation runtime allows the minimum sufficient sample size of the Simul8 simulation model to be determined with an allowance of a 0,5% deviation from the real-world mean output throughput value of the Gas Production plant and a 99% confidence interval, as compared with the 1% deviation and a 99% confidence interval that are used to determine the minimum sufficient sample size of the original simulation model.

* * * * *

3.5 DETERMINATION OF THE SAMPLE SIZE

The results of the different replications of a simulation run of a stochastic simulation model are usually not identical because of the random (*i.e.* the stochastic) behaviour of the random phenomena like failures. This implies that a simulation run consisting of more than one replication has to be completed in order to obtain a mean result that is representative of the simulated scenario.

The determination of the minimum number of replications that would yield a mean result that is representative of the simulated scenario is a determination of minimum sufficient sample size problem. Section 2.1 indicates that Leedy (1993:71) perceives a determination of minimum sufficient sample size problem as a pseudo-subproblem. Leedy maintains that the problem to determine the correct sample size (*i.e.* the minimum sufficient sample size) is merely a pseudo-subproblem or procedural indecision, because there are techniques available to determine sample sizes and it is only necessary to identify the correct one to use in every instance.

In the *Magister* dissertation (Albertyn, 1995:70-72) two different techniques to determine the minimum number of replications (*i.e.* the minimum sufficient sample size) of a simulation run of a stochastic simulation model of the Synthetic Fuel plant are scrutinised. The first is a technique proposed by Crow *et al.* (1960:48) and the second is a technique proposed by Miller *et al.* (1990:209).

Crow *et al.* (1960:48) state that if an estimate of the standard deviation is available, Equation 3.1 can be used to give the sample size necessary to obtain a confidence interval with an expected length of 2h.

$$n_{Sam} = ((\sigma t_{(\alpha / 2, n-1)}) / h)^2 \quad (number) \qquad (Eq.:3.1)$$

Where:

n_{Sam}	:	The sample size.
σ	:	The standard deviation, in the appropriate unit of measurement.
t	:	The upper percentage point of the t distribution value.
$100(1-\alpha)$	:	The confidence interval, as a percentage.
$n-1$	:	The sample size minus one.
h	:	Half (50%) of the expected length of the confidence interval, in the appropriate unit of measurement.

Crow *et al.* refer to the "length" of a confidence interval, while many other references on statistics refer to the "width" of a confidence interval.

Miller *et al.* (1990:209) propose that Equation 3.2 can be used to determine the sample size.

$$n_{Sam} = ((Z_{(\alpha / 2)}\sigma) / E)^2 \quad (number) \qquad (Eq.:3.2)$$

Where:

n_{Sam}	:	The sample size.
Z	:	The Fisher Z transformation value.
$100(1-\alpha)$	:	The confidence interval, as a percentage.
σ	:	The standard deviation, in the appropriate unit of measurement.
E	:	The maximum error of the estimate, in the appropriate unit of measurement.

In the *Magister* dissertation (Albertyn, 1995:70-72) examples are presented where Equations 3.1 and 3.2 are used to determine minimum sufficient sample sizes. It is also indicated that Equation 3.2 can only be used for instances where the minimum sufficient sample size is larger than or equal to 30 (Miller *et al.*, 1990:198,208). A scrutiny of Column 7 of Table 3.2 indicates that the minimum sufficient sample size of a simulation run of a stochastic simulation model of the Synthetic Fuel plant is usually in the order of 12 to 13 (with an allowance for a 0,5% deviation from the real-world mean output throughput value of the Gas Production plant and a 99% confidence interval). These minimum sufficient sample sizes are substantially smaller than the "larger than or equal to 30" requirement of Equation 3.2 and therefore it stands to reason that Equation 3.1 is used throughout this document for the determination of minimum sufficient sample sizes.

The technique that is proposed by Crow *et al.* (1960:48) uses a table that gives the upper percentage point of the *t* distribution values for different sample sizes in the rows of the table and for the most frequently used different confidence intervals in the columns of the table. The technique then uses Equation 3.1 to move with increasing sample size downward through the column of a specific confidence interval until a certain condition is met, thus identifying the required sample size. The condition that must be met is that Equation 3.1 must return a real value that is less than or equal to the integer value of the sample size in the table that corresponds to the upper percentage point of the *t* distribution value in the table that was used to resolve Equation 3.1 in that instance.

This technique lends itself to computerisation and a FORTRAN software programme was developed to speed up the repetitive and rather cumbersome process that the technique uses to determine the sample size. The FORTRAN software programme is called N.FOR and it automatically converges to the correct minimum sufficient sample size with an iterative-loop technique. The relevant input values are handled by an input file called N.IN. An example of N.IN is provided in Appendix J: *N.IN (Sample Size Determination Input File)*. A scrutiny of N.IN reveals that line three contains the value of the confidence interval and that line five contains the value of half (50%) of the expected length of the confidence interval. Lines seven to sixteen contain two values each. The first value in each line is an identifier that identifies a specific simulation run in a series of simulation runs (*i.e.* simulation experiments) and the second value is the standard deviation of that specific simulation run. This example contains the input values of the series of simulation runs that is detailed in Section 3.4. A scrutiny of Table 3.2 reveals that Column 2 of the table contains the identifiers (in this instance it is the iteration time interval of each simulation run) and Column 6 contains the standard deviations of the series of 10 simulation runs that is the topic of discussion in Section 3.4.

N.FOR determines the minimum sufficient sample sizes with the input values that are provided in N.IN and writes the output values to an output file named N.OUT. An example of N.OUT is provided in Appendix K: *N.OUT (Sample Size Determination Output File)*. This example contains the output values that are generated with the input values that are shown in Appendix J (*i.e.* the minimum sufficient sample sizes of the series of simulation runs that is detailed in Section 3.4). A scrutiny of N.OUT reveals that line three contains the value of the confidence interval and that line five contains the value of half (50%) of the expected length of the confidence interval. Lines seven to sixteen contain four values each. The first value in each line is the identifier that identifies the specific simulation run, the second value is the standard deviation of that specific simulation run, the third value is the integer value of the minimum sufficient sample size of that specific simulation run and the fourth value is the real value of the minimum sufficient sample size of that specific simulation run that is returned when Equation 3.1 is resolved. The integer values of the minimum sufficient sample sizes of the series of 10 simulation runs are reflected in Column 7 of Table 3.2.

Summary

The determination of minimum sufficient sample size is addressed in this section. It is indicated that this is merely a pseudo-subproblem or procedural indecision. Two possible techniques are discussed and the technique that is proposed by Crow *et al.* is identified as the appropriate one to use in this instance. A FORTRAN software programme that determines the minimum sufficient sample size is detailed and an example of its use is provided.

* * * * *

3.6 SIMULATION MODEL VERIFICATION AND VALIDATION

Various authors and manuals stress the importance of comprehensive simulation model verification and validation before the results that are generated by a simulation run can be accepted as representative of the simulated scenario (Harrell and Tumay, 1999:87-88; Kelton *et al.*, 1998:444-446; Pegden *et al.*, 1995:129-153; *Simul8®: Manual and Simulation Guide*, 1999:34).

The following quotation from Pegden *et al.* (1995:129) provides definitions for, and distinguishes between, verification and validation:

> "***Verification*** *is the process of determining that a model operates as intended. Throughout the verification process, we try to find and remove unintentional errors in the logic of the model. This activity is commonly referred to as* ***debugging*** *the model. In contrast,* ***validation*** *is the process of reaching an acceptable level of confidence that the inferences drawn from the model are correct and applicable to the real-world system being represented. Through validation , we try to determine whether the simplifications and omissions of detail, which we have knowingly and deliberately made in our model, have introduced unacceptably large errors in the results*"

Harrell and Tumay (1999:87) discuss some of the difficulties that are encountered during simulation model verification and validation.

> "*Eliminating bugs* [verification] *in a program model can take a considerable amount of time especially if a general purpose language is used in which frequent coding errors occur.*"

> "*Proving validity* [validation] *is an elusive undertaking.*"

It is obvious that it is no arbitrary task to verify and validate simulation models of the size and complexity of the Arena and Simul8 simulation models of the Synthetic Fuel plant. A detailed discussion of the verification and validation of the Arena and Simul8 simulation models does not fall within the scope of this document. However, some of the verification and validation concepts are demonstrated with examples in the rest of this section.

One of the most basic tests to verify the Arena and Simul8 simulation models of the Synthetic Fuel plant is to count the number of services and failures that are created by the Arena and Simul8 simulation models during a simulation run and to compare it with the real-world number of services and failures that occur.

In Table 3.3: *Verification of the Simulation Models* a comparison is provided between the real-world number of failures of the modules in the smaller plants that are subject to failures and the number of failures of the modules created by the Arena and Simul8 simulation models during a simulation run.

Simulation runs consisting of 20 replications of a simulated time period of one year (see Appendix L) and with an iteration time interval of one hour were completed with the Arena and Simul8 simulation models. The input values for the services and failures that were used are those that are represented in Table A2 (service schedules and failure characteristics) and Appendix E (start times of service cycles). The mean number of failures of the modules in the smaller plants over the simulated time period of one year created by the Arena and Simul8 simulation models are calculated from the results of the 20 replications and are shown in Columns 6 and 8 of Table 3.3 for the Arena and Simul8 simulation models respectively.

It is important to note that the MTBF and real-world number of failures that occur are calculated for a 360-day year (*i.e.* an 8640-hour year). This is done to conform to the 360-day simulation model year that is used by the Arena and Simul8 simulation models. The primary reason why the 360-day simulation model year is used by the Arena and Simul8 simulation models, is to accommodate the service schedules of the modules of the Synthetic Fuel plant. The concept of the simulation model year is discussed in detail in Appendix L.

Table 3.3: Verification of the Simulation Models

No.	Name	Mod.	MTBF (hour)	No. Fail Real	No. Fail Ar	Dev-Ar (%)	No. Fail S8	Dev-S8 (%)
1	Coal Processing	14	336	360,00	334,50	-7,08	335,20	-6,89
3	Steam	9	2880	27,00	22,70	-15,93	25,00	-7,41
4	Gas Production	40	960	360,00	347,20	-3,56	352,10	-2,19
5	Temperature Regulation	8	5760	12,00	12,45	3,75	11,35	-5,42
6-A	Oxygen-A	6	1080	48,00	46,85	-2,40	46,20	-3,75
6-B	Oxygen-B	6	8640	6,00	6,20	3,33	5,80	-3,33
6-C	Oxygen-C	7	840	72,00	71,95	-0,07	72,05	0,07
6E-C	Oxygen Extra-C	1	1234	7,00	6,75	-3,59	7,40	5,69
7	Electricity Generation	4	1440	24,00	24,95	3,96	22,80	-5,00
8	Plant(I)	4	8640	4,00	4,45	11,25	3,95	-1,25
9-A	Plant(II)-A	8	11520	6,00	6,45	7,50	4,90	-18,33
9-B	Plant(II)-B	2	17280	1,00	1,05	5,00	1,25	25,00
10	Plant(III)	2	8640	2,00	2,50	25,00	1,95	-2,50
11	Division Process	2	8640	2,00	1,80	-10,00	1,80	-10,00
13-A	Plant(IV)-A	4	34560	1,00	0,95	-5,00	0,90	-10,00
13-B	Plant(IV)-B	2	17280	1,00	0,65	-35,00	1,15	15,00
13-C	Plant(IV)-C	1	34560	0,25	0,15	-40,00	0,30	20,00
20	Plant(V)	8	5317	13,00	10,90	-16,15	11,05	-15,00

Where:

No.	:	The plant identification number.
Mod.	:	The number of modules in the plant.
MTBF	:	The Mean Time Between Failure of the modules (hour).
No. Fail Real	:	The real-world number of failures that occur during a one year period (calculated with the real-world MTBF).
No. Fail Ar	:	The mean number of failures created by the Arena simulation model during a simulated time period of one year.
Dev-Ar	:	The deviation of the mean number of failures created by the Arena simulation model from the real-world number of failures that occur (%).
No. Fail S8	:	The mean number of failures created by the Simul8 simulation model during a simulated time period of one year.

Dev-S8 : The deviation of the mean number of failures created by the Simul8 simulation model from the real-world number of failures that occur (%).

A scrutiny of Column 7 of Table 3.3 reveals that the deviations of the number of failures created by the Arena simulation model, from the real-world number of failures that occur, vary in a range from a deviation as small as -0,07% (Oxygen-C) to a deviation as large as -40,00% (Plant(IV)-C). A deviation of -40,00% seems excessive but it could still be acceptable if the large MTBF value (or conversely the low failure rate) of Plant(IV)-C is taken into account. The MTBF of Plant(IV)-C is 34560 hours and that translates into approximately one failure every four years. Such a low failure rate could easily lead to a large deviation from the real-world number of failures that occur because the simulated time period of one year is considerably shorter than the MTBF of four years. This implies that the number of failures created by the Arena simulation model is small and therefore the randomness of the failures is accentuated. However, it is still good simulation modelling practice to thoroughly investigate any large deviations. Even though some of the deviations in Column 7 of Table 3.3 assume large values, the overall impression is that the Arena simulation model operates as intended, insofar as the number of failures created is concerned.

A scrutiny of Column 9 of Table 3.3 reveals that the deviations of the number of failures created by the Simul8 simulation model, from the real-world number of failures that occur, vary in a range from a deviation as small as 0,07% (Oxygen-C) to a deviation as large as 25,00% (Plant(II)-B). The same arguments as those stated in the previous paragraph about the Arena simulation model deviations is applicable to the Simul8 simulation model deviations. Even though some of the deviations in Column 9 of Table 3.3 assume large values, the overall impression is that the Simul8 simulation model operates as intended, insofar as the number of failures created is concerned.

A simulation model is usually validated by comparing the behaviour of the simulation model in a known scenario with the behaviour of the real-world system in the known scenario. In this instance the mean output throughput values of the Arena and Simul8 simulation models in a known scenario are compared to the real-world mean output throughput value of the Synthetic Fuel plant in the known scenario. The known scenario is the 1993 production year of the Synthetic Fuel plant and the mean raw gas output throughput value of the Gas Production plant is used as the variable of comparison. The monthly mean output throughput values of the Gas Production plant during the 1993 production year are indicated in Table M1: *Gas Production Plant Output Throughput -1993* (see Appendix M). From Table M1 it follows that the mean output throughput value of the Gas Production plant during the 1993 production year was 1332234,2 nm^3/h.

In Table 3.4: *Validation of the Simulation Models* the Arena and Simul8 simulation models are validated by comparing the mean output throughput values of the Gas Production plant that are generated by their respective simulation runs with the mean output throughput value of the Gas Production plant during the 1993 production year.

Table 3.4: Validation of the Simulation Models

Simulation Model	ITI (hour)	n_{Rep}	Runtime (min)	GasPro (nm^3/h)	StdDev (nm^3/h)	n_{Sam}	Deviation (%)
Arena	1	20	24,0	1326773,7	8066,6	14	-0,410
Simul8	1	20	17,0	1331462,8	7154,9	12	-0,058

Where:

ITI : The iteration time interval (hour).

n_{Rep} : The number of replications completed.

Runtime : The simulation runtime for n_{Rep} replications (minute).

GasPro : The mean output throughput value of the Gas Production plant, calculated from n_{Rep} replications (nm^3/h).

StdDev : The standard deviation from the mean output throughput value (nm^3/h).

n_{Sam} : The minimum sufficient sample size.

Deviation : The deviation of the specific mean output throughput value from the mean output throughput value of the Gas Production plant during the 1993 production year (%).

Simulation runs consisting of 20 replications of a simulated time period of one year (see Appendix L) and with an iteration time interval of one hour were completed with the Arena and Simul8 simulation models. The input values for the services and failures that were used are those that are represented in Table A2 (service schedules and failure characteristics) and Appendix E (start times of service cycles).

The means and the standard deviations from the means of the output throughput values of the Gas Production plant are calculated from the results of the 20 replications. The standard deviations are used to calculate the corresponding minimum sufficient sample sizes with an allowance for a 0,5% deviation from the real-world mean output throughput value of the Gas Production plant (see Appendix M) and a 99% confidence interval. Section 3.5 provides a detailed explanation of the determination of minimum sufficient sample size. The number of replications completed in both instances should be more than, or equal to, the calculated minimum sufficient sample sizes for the answers to be taken as representative of the simulated scenario. A scrutiny of Columns 3 and 7 of Table 3.4 indicates that this constraint is adhered to.

From Table 3.4 it follows that the mean output throughput values of the Gas Production plant of the Arena and Simu8 simulation models deviate only -0,410% and -0,058% respectively from the mean output throughput value of the Gas Production plant during the 1993 production year.

These results (deviations of less than 1% for the Arena and Simul8 simulation models) indicate that it can be accepted that the Arena and Simul8 simulation models with an iteration time interval of one hour are valid representations of the Synthetic Fuel plant.

These results correlate closely with the *Magister* dissertation (Albertyn, 1995:76) which indicates that the original simulation model with an iteration time interval of one hour also deviates less than 1% (0,59%) from the real-world situation for the same known scenario.

The sensitivity of the Arena and Simul8 simulation models, with regard to the input values for the services and failures that are used, is also worthy of consideration. The only input values that are "variable" in the strict sense of the word are the start times of the service cycles (see Appendix E). The input values for the cycle times, services times, failure rates and repair times are "fixed" in terms of the system description of the Synthetic Fuel plant that is provided in Section 1.2 (see Table A2).

Table 3.5: *Sensitivity of the Simulation Models* provides an indication of the sensitivity of the Arena and Simul8 simulation models in terms of variation in the start times of the service cycles. Three different scenarios for the start times of the service cycles are considered for both simulation models.

The three different scenarios are the following:

a) Scenario 1 - at the start of the simulation run, every service cycle (excluding the "phase" services) is considered to start just after the completion of the last service of a sequence of services.

b) Scenario 2 - at the start of the simulation run, every service cycle (excluding the "phase" services) is considered to start exactly halfway through the service cycle (see Appendix E).

c) Scenario 3 - at the start of the simulation run, every service cycle (excluding the "phase" services) is considered to start with the first service of a sequence of services.

Table 3.5: Sensitivity of the Simulation Models

Simulation Model	ITI (hour)	n_{Rep}	Runtime (min)	GasPro (nm^3/h)	StdDev (nm^3/h)	n_{Sam}	Deviation (%)
Arena (Scenario 1)	1	20	24,0	1340731,2	7220,3	12	0,638
Arena (Scenario 2)	1	20	24,0	1326773,7	8066,6	14	-0,410
Arena (Scenario 3)	1	20	24,0	1320225,4	7863,5	14	-0,901
Simul8 (Scenario 1)	1	20	17,0	1343426,6	6887,4	11	0,840
Simul8 (Scenario 2)	1	20	17,0	1331462,8	7154,9	12	-0,058
Simul8 (Scenario 3)	1	20	17,0	1322135,6	7015,2	12	-0,758

Where:

ITI	:	The iteration time interval (hour).
n_{Rep}	:	The number of replications completed.
Runtime	:	The simulation runtime for n_{Rep} replications (minute).
GasPro	:	The mean output throughput value of the Gas Production plant, calculated from n_{Rep} replications (nm^3/h).
StdDev	:	The standard deviation from the mean output throughput value (nm^3/h).
n_{Sam}	:	The minimum sufficient sample size.
Deviation	:	The deviation of the specific mean output throughput value from the mean output throughput value of the Gas Production plant during the 1993 production year (%).

Scenario 2 represents the input values for the start times of the service cycles (see Appendix E) that are used for all the other simulation runs in this document because they represent a good middle-of-the-road option.

Simulation runs consisting of 20 replications of a simulated time period of one year (see Appendix L) and with an iteration time interval of one hour were completed for the previously mentioned three different scenarios with the Arena and Simul8 simulation models (*i.e.* a total of six simulation runs was completed). The input values for the services and failures that were used are those that are represented in Table A2 (service schedules and failure characteristics). The input values for the start times of the service cycles are those that are described above for the three different scenarios.

The means and the standard deviations from the means of the output throughput values of the Gas Production plant are calculated from the results of the 20 replications. The standard deviations are used to calculate the corresponding minimum sufficient sample sizes with an allowance for a 0,5% deviation from the real-world mean output throughput value of the Gas Production plant (see Appendix M) and a 99% confidence interval. Section 3.5 provides a detailed explanation of the determination of minimum sufficient sample size. The number of replications completed in all instances should be more than, or equal to, the calculated minimum sufficient sample sizes for the answers to be taken as representative of the simulated scenario. A scrutiny of Columns 3 and 7 of Table 3.5 indicates that this constraint is adhered to.

From Table 3.5 it follows that none of the mean output throughput values of the Gas Production plant of the Arena and Simu8 simulation models deviate more than 1% from the mean output throughput value of the Gas Production plant during the 1993 production year. The maximum delta between the deviations of the Arena simulation model is between Scenario 1 and 3 and it is 1,539% (0,638% minus -0,901%). The maximum delta between the deviations of the Simul8 simulation model is between Scenario 1 and 3 and it is 1,598% (0,840% minus -0,758%).

These results indicate that the maximum bandwidth of variation of the mean output throughput values of the Gas Production plant of the Arena and Simu8 simulation models is less than 2% of the mean output throughput value of the Gas Production plant during the 1993 production year. It can therefore be deducted that the Arena and Simul8 simulation models are not overly sensitive to variation if the input values for the start times of the service cycles are varied between the extremes of Scenario 1 and 3.

Another concept that has to be introduced is the confidence interval for a population mean. Various sources (Miller *et al.*, 1990:210-214; Pegden *et al.*, 1995:36-38; *Simul8®: Manual and Simulation Guide*, 1999:39-48) detail the theoretical background for the determination of a confidence interval for a population mean (see Appendix N: *Determination of the Confidence Interval*).

Table 3.6: *99% Confidence Intervals for the Output Throughput* provides the 99% confidence intervals for the mean output throughput values of the six scenarios that are under scrutiny. The

mean output throughput values of the Gas Production plant are used.

Table 3.6: 99% Confidence Intervals for the Output Throughput

Simulation Model	GasPro (nm^3/h)	StdDev (nm^3/h)	ConInt (nm^3/h)	Lower ConLmt (nm^3/h)	Upper ConLmt (nm^3/h)
Arena (Scenario 1)	1340731,2	7220,3	9238,2	1336112,1	1345350,3
Arena (Scenario 2)	1326773,7	8066,6	10321,0	1321613,2	1331934,2
Arena (Scenario 3)	1320225,4	7863,5	10061,2	1315194,8	1325256,0
Simul8 (Scenario 1)	1343426,6	6887,4	8812,3	1339020,5	1347832,7
Simul8 (Scenario 2)	1331462,8	7154,9	9154,5	1326885,5	1336040,1
Simul8 (Scenario 3)	1322135,6	7015,2	8975,8	1317647,7	1326623,5

Where:

GasPro : The mean output throughput value of the Gas Production plant, calculated from n_{Rep} replications (nm^3/h).

StdDev : The standard deviation from the mean output throughput value (nm^3/h).

ConInt : The confidence interval (nm^3/h).

ConLmt : The confidence limit (nm^3/h).

Pegden *et al.* (1995:36-38) and the *Simul8®: Manual and Simulation Guide* (1999:39-48) indicate that the confidence intervals should be taken into consideration when alternatives are compared. If the confidence intervals for the mean output throughput values of two scenarios overlap, the two scenarios cannot be differentiated in terms of representing two different outcomes.

A scrutiny of Columns 5 and 6 of Table 3.6 reveals that the 99% confidence intervals for the mean output throughput values of the Scenario 1 and 3 Arena simulation models do not overlap and therefore the two scenarios can be assumed to represent two different outcomes. This implies that it is valid to determine and use the delta between the deviations of Scenario 1 and 3 of the Arena simulation model during the sensitivity analysis (see Table 3.5). Furthermore, the 99% confidence intervals for the mean output throughput values of the Scenario 1 and 3 Simul8 simulation models also do not overlap and therefore the two scenarios can be assumed to represent two different outcomes. This implies that it is valid to determine and use the delta between the deviations of Scenario 1 and 3 of the Simul8 simulation model during the sensitivity analysis (see Table 3.5).

Summary

Some of the verification and validation concepts of the Arena and Simul8 simulation models are discussed and demonstrated with examples in this section. The example that demonstrates the verification of the simulation models indicates that the simulation models operate as intended, insofar as the number of failures created is concerned. The validation example compares the mean output throughput values of the Gas Production plant of the simulation models with the mean output throughput value of the Gas Production plant during the 1993 production year. The results (deviations of less than 1% from the 1993 production year) indicate that the simulation models can be accepted as valid representations of the Synthetic Fuel plant. The sensitivity of the simulation models in terms of variation in the start times of the service cycles is investigated and the conclusion is reached that the simulation models are not overly sensitive for variation in the start times of the service cycles. Confidence intervals for the mean output throughput values of the simulation models are also determined.

* * * * *

3.7 SIMULATION MODEL ENHANCEMENT

The original, Arena and Simul8 simulation models use a fixed time interval (*i.e.* an iteration time interval) to advance the simulation models in time. This concept is explained, developed and detailed in Sections 1.4, 1.6, 1.7, 2.2 and 3.4. If an iteration time interval concept is used to advance a simulation model in time, it will be referred to as an iteration time interval (ITI) evaluation method in the rest of this document.

However, another possibility to advance the original, Arena and Simul8 simulation models in time, does exist. The event-driven evaluation concept advances a simulation model in time by evaluating the simulation model only when an event takes place. If an event-driven evaluation concept is used to advance a simulation model in time, it will be referred to as an event-driven (ED) evaluation method in the rest of this document.

A summary of the most salient points of the ITI evaluation method is provided here for the sake of continuity and to provide an introduction to the arguments that support the development of the ED evaluation method.

The basic principles of the ITI evaluation method are based on the methods of classical mathematics. In classical mathematics the behaviour of a continuous system over a period of time is usually modelled with the help of differential equations. Unfortunately, analytical solutions are only available for rather simplistic differential equations. As soon as more complex differential equations are encountered, numerical methods seem to be the only viable solution. One such method involves the discretisation (division into discrete elements) of the continuous behaviour of the system over the time period into behaviour during specific time intervals. The behaviour of the system is evaluated at the start of every time interval and is assumed to remain constant for the duration of the time interval. The total behaviour of the system over the time period is then found by the summation of the behaviour during the specific time intervals. If the time interval between evaluations is chosen correctly in accordance with the dynamic response characteristics of the system that is modelled the results that are obtained can be a very close approximation of the real-world situation that is modelled. It is common practice to use a fixed time interval (*i.e.* an iteration time interval) between evaluations.

The ED evaluation method works on the principle that the behaviour of a system over a period of time can only change when an event takes place and assumes that the behaviour is constant between events. Therefore the behaviour of the system will remain constant until an event takes place that necessitates the re-evaluation of the system to determine the new behaviour. The total behaviour of the system over the time period is then found by the summation of the behaviour between the different points in time that the events took place.

The basic difference between the two evaluation methods is that the ITI evaluation method evaluates a simulation model with a time interval that is of constant (*i.e.* fixed) length, while the ED evaluation method evaluates a simulation model with a time interval that is of variable length, depending on the events that take place.

The flexibility of the generic simulation modelling methodology and therefore also the flexibility of the Arena and Simul8 simulation models, can be greatly enhanced by the inclusion of an ED evaluation method option. The reason why an ED evaluation method option can be incorporated into the generic methodology, is because the generic methodology does not make provision for the inclusion of transient behaviour. It is assumed that the changes in the state of the system occur at isolated (specific) points in time. The reasons for this assumption are provided in Section 1.7 and its validity is provided in Section 3.6.

The following six different types of events that take place in simulation models that are developed with the generic simulation modelling methodology can be identified:

a) The beginning and end of each replication of the simulation run.

b) The beginning and end of each service of the modules.

c) The beginning and end of each failure of the modules.

In order to explain one of the possible benefits of using an ED evaluation method in simulation models that are developed with the generic simulation modelling methodology, it is necessary to introduce the concept of event density. In this context event density may be defined as the number of events per time unit (see Equation 3.3).

$$Density_{Evt} = n_{Evt} \,/\, Time \quad (event/hour) \qquad (Eq.:3.3)$$

Where:

$Density_{Evt}$: The event density, in events per hour.

n_{Evt} : The number of events.

The event density value of a simulation model can be used to determine which of the two evaluation methods (*i.e.* the ITI or ED evaluation method) is appropriate for that specific application. Of course, the event density value of a simulation model cannot be calculated before a simulation run consisting of a number of replications has been completed. During a simulation run the number of events that take place during each replication can be counted and consequently the mean number of events and the event density value of the simulation model can be calculated. Paradoxically, this implies that the simulation model should already exist before it can be determined which of the two evaluation methods is appropriate for a specific application. This problem is circumvented by making a first-order estimate of the number of events that should take place per replication.

Table O1: *Number of Services and Failures (8640-hour year)* of Appendix O: *First-order Estimate of the Number of Services and Failures* provides a first-order estimate of the number of services and failures that should take place in the Arena and Simul8 simulation models over a simulated time period of one year. From Table O1 it follows that the estimated number of events in the simulation models over a simulated time period of one year is 4024 events. That is two events for the beginning and end of each replication, 2132 events that are related to the beginning and end of each service (1066 services multiplied by 2) and 1890 events that are related to the beginning and end of each failure (945 failures multiplied by 2). That gives an estimated event density value of 0,47 events per hour (4024 events divided by 8640 hours) for the simulation models.

An ED evaluation method option was incorporated into the Arena and Simul8 simulation models and a simulation run was completed with the ED evaluation method option for both the simulation models. In Table 3.7: *Validation of the ED Evaluation Method Option Simulation Models* the ED evaluation method option Arena and Simul8 simulation models are validated and a comparison with the validation of their ITI evaluation method option counterparts with an iteration time interval of one hour (see Table 3.4) is provided.

It is imperative to note that both the Arena and Simul8 simulation models incorporate an ITI evaluation method option and an ED evaluation method option in the same simulation model. The original simulation model, on the other hand, only incorporates an ITI evaluation method option. All the results that are shown up to this point were generated with the ITI evaluation method option of the Arena and Simul8 simulation models. Even though both evaluation method options are available in the Arena simulation model and it is essentially exactly the same simulation model that is used, the simulation model will be referred to as the ITI evaluation method option Arena simulation model when the ITI evaluation method option is used and as the ED evaluation method option Arena simulation model when the ED evaluation method option is used. The same logic applies to the Simul8 simulation model.

Table 3.7: Validation of the ED Evaluation Method Option Simulation Models

Simulation Model	ITI (hour)	n_{Evt}	D_{Evt} (e/h)	n_{Rep}	Runtime (min)	GasPro (nm^3/h)	StdDev (nm^3/h)	n_{Sam}	Deviation (%)
Arena (ITI)	1	-	-	20	24,0	1326773,7	8066,6	14	-0,410
Arena (ED)	-	3242,3	0,38	20	8,6	1332471,8	6620,5	11	0,018
Simul8 (ITI)	1	-	-	20	17,0	1331462,8	7154,9	12	-0,058
Simul8 (ED)	-	3259,6	0,38	20	6,8	1332253,3	7462,5	13	0,001

Where:

ITI	:	The iteration time interval (hour).
n_{Evt}	:	The mean number of events (simulation model evaluations), calculated from n_{Rep} replications.
D_{Evt} (e/h)	:	The event density value (event/hour).
n_{Rep}	:	The number of replications completed.
Runtime	:	The simulation runtime for n_{Rep} replications (minute).

GasPro	:	The mean output throughput value of the Gas Production plant, calculated from n_{Rep} replications (nm^3/h).
StdDev	:	The standard deviation from the mean output throughput value (nm^3/h).
n_{Sam}	:	The minimum sufficient sample size.
Deviation	:	The deviation of the specific mean output throughput value from the mean output throughput value of the Gas Production plant during the 1993 production year (%).

Simulation runs consisting of 20 replications of a simulated time period of one year (see Appendix L) were completed with the ED evaluation method option Arena and Simul8 simulation models. The input values for the services and failures that were used are those that are represented in Table A2 (service schedules and failure characteristics) and Appendix E (start times of service cycles).

The mean number of events and the event density values, as well as the means and the standard deviations from the means of the output throughput values of the Gas Production plant, are calculated from the results of the 20 replications. The standard deviations are used to calculate the corresponding minimum sufficient sample sizes with an allowance for a 0,5% deviation from the real-world mean output throughput value of the Gas Production plant (see Appendix M) and a 99% confidence interval. Section 3.5 provides a detailed explanation of the determination of minimum sufficient sample size. The number of replications completed in both instances should be more than, or equal to, the calculated minimum sufficient sample sizes for the answers to be taken as representative of the simulated scenario. A scrutiny of Columns 5 and 9 of Table 3.7 indicates that this constraint is adhered to.

From Table 3.7 it follows that the mean output throughput values of the Gas Production plant of the ED evaluation method option Arena and Simu8 simulation models deviate only 0,018% and 0,001% respectively from the mean output throughput value of the Gas Production plant during the 1993 production year.

These results (deviations of less than 1% for the ED evaluation method option Arena and Simul8 simulation models) indicate that it can be accepted that the ED evaluation method option Arena and Simul8 simulation models are valid representations of the Synthetic Fuel plant.

Section 3.6 indicates that the ITI evaluation method option Arena and Simul8 simulation models with an iteration time interval of one hour are also valid representations of the Synthetic Fuel plant and therefore it is clear that the ITI (with an iteration time interval of one hour) and ED evaluation method option Arena and Simul8 simulation models (*i.e.* two instances of the Arena simulation model and two instances of the Simul8 simulation model) are all valid representations of the Synthetic Fuel plant.

From Table 3.7 it follows that the calculated event density value of the ED evaluation method option Arena and Simul8 simulation models is 0,38. This differs significantly from the estimated event density value of 0,47. This deviation can be attributed to the fact that the n_{Evt} values in Table 3.7 represent the mean number of simulation model evaluations and not, in the strict sense of the word, the exact mean number of events. The mean number of simulation model evaluations differs from the mean number of events because some of the events are concurrent. For instance, more than one module can start a service at exactly the same time. This implies that one evaluation can capture more than one event and therefore the mean number of simulation model evaluations is generally less than the mean number of events in an ED evaluation method simulation model.

The difference in the simulation runtimes of the ITI and ED evaluation method option Arena and Simul8 simulation models are of special significance. The simulation runtime of the ITI evaluation method option Arena simulation model with an iteration time interval of one hour is 24,0 minutes and that of the ED evaluation method option Arena simulation model is 8,6 minutes. That is an improvement of more than 50% in terms of simulation runtime for the Arena simulation model, if the ED evaluation method option is used. The simulation runtime of the ITI evaluation method option Simul8 simulation model with an iteration time interval of one hour is 17,0 minutes and that of the ED evaluation method option Simul8 simulation model is 6,8 minutes. That is an improvement of more than 50% in terms of simulation runtime for the Simul8 simulation model, if the ED evaluation method option is used.

These results could be expected because the event density value of the ITI evaluation method option Arena and Simul8 simulation models with an iteration time interval of one hour is 1,00 (8641 events or evaluations divided by 8640 hours - the extra event or evaluation is the beginning of each replication). In the instance of the ITI evaluation method option Arena and Simul8 simulation models the events are, of course, the simulation model evaluations that take place every iteration time interval. It can therefore be concluded that a low event density value leads to a shorter simulation runtime.

In Section 3.4 the effect of the iteration time interval on the accuracy of the ITI evaluation method option Simul8 simulation model is investigated. The results indicate that if a deviation of 0,5% from the perceived correct answer (*i.e.* the one that is generated by the simulation run with the shortest iteration time interval) is taken as an acceptable deviation, all iteration time intervals up to and including six hours seem acceptable. From Table 3.2 it follows that the simulation runtime for the ITI evaluation method option Simul8 simulation model with an iteration time interval of six hours is only 3,1 minutes. That is considerably shorter than the simulation runtime of 6,8 minutes for the ED evaluation method option Simul8 simulation model. It therefore seems tempting to use the ITI evaluation method option Simul8 simulation model with an iteration time interval of six hours if a short simulation runtime is a prerequisite. Even though the cold figures suggest that it is a valid option, intuitively it seems a better option to avoid the possible risk of deviation from the correct answer, by rather using the ED evaluation method option Simul8 simulation model with the still very acceptable simulation runtime of 6,8 minutes.

The ITI and ED evaluation methods are compared and their strengths and weaknesses are discussed in a conference paper by Albertyn (2000 Summer Computer Simulation Conference, 2000:129-134). Only the most pertinent points of discussion in the paper will be touched upon here to provide some insight into the characteristics of the two evaluation methods. The ITI and ED evaluation methods can be compared in terms of accuracy, complexity of simulation model construction, ease of use and simulation runtimes.

In terms of accuracy there is no discernible distinction between the two evaluation methods, provided that an appropriate iteration time interval is used by the ITI evaluation method (see Section 3.4 and Table 3.7). Both evaluation methods can render extremely accurate results.

As far as complexity of simulation model construction is concerned, an ITI evaluation method simulation model is more straightforward and less complex than an ED evaluation method simulation model. An ED evaluation method simulation model needs additional logic to identify when the next event will take place and consequently the complexity of simulation model construction increases. In the instance of the Arena and Simul8 simulation models it proved to be extremely difficult to incorporate an ITI and ED evaluation method option into the same simulation model. The basic concepts of the ITI and ED evaluation methods differ substantially and therefore they do not lend themselves to easy integration and synergism.

There is no difference in the ease of use of the two evaluation methods. The ITI and ED evaluation method option Arena simulation models use exactly the same input and output files and the ITI and ED evaluation method option Simul8 simulation models use exactly the same spreadsheet variables as input and output mechanisms (see Section 3.3). The input and output files of the Arena simulation models and the spreadsheet variables of the Simul8 simulation models enhance user-friendliness.

The simulation runtimes of ITI and ED evaluation method simulation models depend on the computer hardware configuration and simulation software package that are used as well as the size and complexity of the simulation model. In addition, the simulation runtime of an ITI evaluation method simulation model also depends on the iteration time interval that is used (see Section 3.4). The simulation runtimes of the ITI and ED evaluation method option Arena and Simul8 simulation models have already been discussed in this section. It will suffice to summarise by stating that, for the computer hardware configuration and simulation software packages that were used for the simulation experiments that are discussed in this document, the simulation runtimes of the ED evaluation method option Arena and Simul8 simulation models are about 50% of those of the ITI evaluation method option Arena and Simul8 simulation models with an iteration time interval of one hour.

The principal features of the hardware configuration of the computer that was used for all the simulation experiments that are discussed in this document are an 800-megahertz processor and 128 megabytes of RAM.

The **strengths** of the ITI evaluation method are accuracy (if an appropriate iteration time interval is used), straightforward and less complex simulation model construction and ease of use (if input and output files or spreadsheet variables are used). Short simulation runtimes can also be achieved by increasing the iteration time interval up to the acceptable limit.

The **weakness** of the ITI evaluation method is that a bandwidth of iteration time intervals that render valid results has to be determined before the simulation model can be used. This is a somewhat cumbersome exercise (see Section 3.4).

The **strengths** of the ED evaluation method are accuracy and ease of use (if input and output files or spreadsheet variables are used). There is also no need to determine a bandwidth of iteration time intervals that render valid results.

The **weaknesses** of the ED evaluation method are a more complex simulation model construction and the fact that the simulation runtime for a specific simulation model in a specific simulation software package is a given that depends on the computer hardware configuration.

Summary

In this section the Arena and Simul8 simulation models are enhanced by the inclusion of an additional evaluation method option. The ED evaluation method option evaluates the simulation models only when an event takes place. The concept of event density is introduced and it is indicated that the event density value of a simulation model can be used to determine which of the ITI or ED evaluation method options is appropriate for that specific application. Simulation runs are completed with the ED evaluation method option simulation models and the results are validated. The results (deviations of less than 1% from the 1993 production year) indicate that the ED evaluation method option simulation models can be accepted as valid representations of the Synthetic Fuel plant. The ITI and ED evaluation methods are also compared and their strengths and weaknesses are discussed.

* * * * *

3.8 COMPARISON OF THE SIMULATION MODELS AND THE SIMULATION SOFTWARE PACKAGES

In Section 3.6 the ITI evaluation method option Arena and Simul8 simulation models with an iteration time interval of one hour are validated and in Section 3.7 the ED evaluation method option Arena and Simul8 simulation models are validated. Table 3.7 indicates that the simulation runtimes of the ED evaluation method option Arena and Simul8 simulation models are approximately 50% of those of their ITI evaluation method option counterparts with an iteration time interval of one hour. These results follow from the fact that the event density value of the ED evaluation method option Arena and Simul8 simulation models is only 0,38 (see Table 3.7) while the event density value of the ITI evaluation method option Arena and Simul8 simulation models is 1,00. It therefore stands to reason that the ED evaluation method option Arena and Simul8 simulation models are the preferred options when scenario analysis is conducted because of their shorter simulation runtimes. From this point onward, only the ED evaluation method option Arena and Simul8 simulation models are used and discussed.

An introductory comparison of the ED evaluation method option Arena and Simul8 simulation models and the Arena and Simul8 simulation software packages are provided in a conference paper by Albertyn and Kruger (16th European Simulation Multiconference, 2002:29-36) and a more detailed version thereof is provided in a published article by Albertyn and Kruger (2003:57-60). The comparisons provided in the conference paper and the published article are repeated here and expanded upon for the sake of continuity and completeness.

Table 3.8: *Comparison of the Simulation Models* provides a comparison between the ED evaluation method option Arena and Simul8 simulation models. The values that are presented in Table 3.8 are mostly taken from Table 3.7 (*i.e.* for a simulated time period of one year) but a few other values are also added. This might seem like an unnecessary repetition but the discussion in Section 3.7 compares the ITI and ED evaluation methods and the way that they manifest themselves in the Arena and Simul8 simulation model environments, while the discussion here compares the ED evaluation method option Arena and Simul8 simulation models.

Table 3.8: Comparison of the Simulation Models

Attribute	ED Evaluation Method Option Arena Simulation Model	ED Evaluation Method Option Simul8 Simulation Model
n_{Evt}	3242,3	3259,6
$Density_{Evt}$ (event/h)	0,38	0,38
n_{Rep}	20	20
Runtime (min)	8,6	6,8
GasPro (nm^3/h)	1332471,8	1332253,3
StdDev (nm^3/h)	6620,5	7462,5
n_{Sam}	11	13
Deviation (%)	0,018	0,001
Size (KB)	2438	937

Where:

n_{Evt}	:	The mean number of events (simulation model evaluations), calculated from n_{Rep} replications.
$Density_{Evt}$	:	The event density value (event/hour).
n_{Rep}	:	The number of replications completed.
Runtime	:	The simulation runtime for n_{Rep} replications (minute).
GasPro	:	The mean output throughput value of the Gas Production plant, calculated from n_{Rep} replications (nm^3/h).
StdDev	:	The standard deviation from the mean output throughput value (nm^3/h).

n_{Sam}	:	The minimum sufficient sample size.
Deviation	:	The deviation of the specific mean output throughput value from the mean output throughput value of the Gas Production plant during the 1993 production year (%).
Size	:	The simulation model size (kilobyte).

The mean number of events (*i.e.* the mean number of simulation model evaluations) of the ED evaluation method option Arena and Simul8 simulation models correlates closely and are 3242,3 and 3259,6 respectively. That gives an identical event density value of 0,38 for both simulation models. The simulation runtime of the Arena simulation model is 8,6 minutes and that of the Simul8 simulation model is slightly less at 6,8 minutes. The minimum sufficient sample size of the Arena simulation model is 11 and that of the Simul8 simulation model is 13 because of the slightly larger standard deviation value of the Simul8 simulation model. Both simulation models render extremely accurate results with deviations of only 0,018% (Arena simulation model) and 0,001% (Simul8 simulation model) from the mean output throughput value of the Gas Production plant during the 1993 production year. The size of the Arena simulation model is 2438 kilobytes while the Simul8 simulation model is considerably smaller at only 937 kilobytes.

The simulation runtimes of 8,6 and 6,8 minutes for the ED evaluation method option Arena and Simul8 simulation models respectively, represent an approximate fortyfold improvement in simulation runtime over the 5,7 hour simulation runtime of the original simulation model (see Section 3.4).

Table 3.9: *Comparison of the Simulation Software Packages* provides a comparison between the *Arena Standard Edition* and *Simul8 Standard* simulation software packages. It should be noted that some of the statements in Table 3.9 are subjective perceptions and not scientifically deduced conclusions. These perceptions follow from the use of the two simulation software packages during the development of the Arena and Simul8 simulation models.

The Arena acquisition cost and annual licencing fees are given as values normalised to the acquisition cost of Simul8. The acquisition cost and annual licencing fees of the simulation software packages change over time because the developers adjust prices to accommodate software upgrades and inflation. Therefore the values for acquisition cost and annual licencing fees that are presented in Table 3.9 are only representative and not absolute.

Table 3.9: Comparison of the Simulation Software Packages

Attribute	Arena	Simul8
Acquisition cost	13,6	1
Annual licencing fees	2,0	None
Graphics capability	More advanced	More basic
Modelling environment complexity (familiarisation, use, *etc.*)	More complex	More simplistic
Simulation modelling capability	More capable	Adequate
Simulation model ease of use (variable manipulation, input and output mechanisms, *etc.*)	More difficult	More easy
Numerical accuracy	15 decimal digits	10 decimal digits
Logic programming language accessibility	Less accessible (VBA is accessible but not integral part of software)	More accessible (Visual Logic is integral part of software)
Simulation model size	Larger	Smaller
Simulation runtime	Longer	Shorter
Random number generation test	Pass	Pass

Where:

VBA : Visual Basic for Applications

It should be noted that a less expensive version of Arena, called *Arena Basic Edition*, is also available. The acquisition cost of *Arena Basic Edition* is about a third of that of Simul8 and it has no annual licencing fee. It does, however, only allow modelling with the *Basic Process* template. The *Basic Process* template contains only the most basic simulation software package building blocks and a vital omission is the ability to read data from, or write data to, an external file. The *ReadWrite* building block of Arena is contained in the *Advanced Process* template that is not available in *Arena Basic Edition*. A basic design philosophy of the generic simulation modelling methodology is to use the most basic of the standard simulation software package building blocks (in the respective simulation software packages) whenever possible. This approach supports the design criteria of compact simulation model size and short simulation runtimes (see Section 1.5). The ability to read input variables from, or to write output variables to, an external file is seen as one of the basic capabilities that is needed to support the user-friendliness design criterion of the generic methodology. Apparently the capability to read input variables from, or to write output variables to, an external file can be achieved in *Arena Basic*

Edition through the use of VBA code. This possibility, however, violates the single software application design criterion of the generic methodology and it was thus not considered a viable option.

A variable in Arena is accurate to 15 decimal digits (that is comparable with a Double Precision or Real*8 variable defined in FORTRAN) and in Simul8 a variable is accurate to 10 decimal digits. This difference should not be of concern to a modeller in the normal applications of this type of simulation software package. Operations where floating-point errors tend to accumulate, however, will need extra consideration (see Section 2.3 for a discussion about the effect of floating-point errors on the service schedules).

Section 1.5 shows that the generic simulation modelling methodology presents a structured approach that renders simulation models with the following characteristics: short development time, short maintenance time, user-friendliness, short simulation runtimes, compact size, robustness, accuracy and preferably a single software application. Both the Arena and Simul8 simulation software packages conform to all these characteristics. In both packages short development and maintenance times are achieved through the use of the high-level building blocks. Both packages allow hierarchical modelling (through the use of *submodels* in the Arena environment and *sub-windows* in the Simul8 environment) and support user-friendliness with their input and output mechanisms (through the use of input and output files in the Arena environment and spreadsheet variables in the Simul8 environment). These input mechanisms allow fast and easy access to input and output variables. Acceptable simulation runtimes and compact simulation model sizes are achievable with both packages. The robustness of the generic methodology and both packages are proved by the ease of simulation model construction in both instances. Both packages produce accurate simulation models (proved through verification and validation) and allow the whole simulation model to be accommodated in a single software application.

The **strengths** of the Arena simulation software package are a more advanced graphics capability and additional modelling capabilities, like transporters, conveyors, *etc.* These additional capabilities do not feature in the generic simulation modelling methodology, but could be important for users when seen in the broader perspective of general simulation modelling applications. Arena is also more widely accepted as an "industry standard" among simulation software packages. According to marketing material of Arena more than 75% of the top 30 companies in Fortune's Global 500 use Arena. The use of input and output files as input and output mechanisms enhance user-friendliness and therefore the ease of use of Arena simulation models is also perceived as a strength of Arena, even though the ease of use is described as "more difficult" in Table 3.9.

The **weaknesses** of the Arena simulation software package are higher acquisition cost, annual licencing fees, more complex modelling environment (and thus more difficult to learn and use), no internal logic programming language, larger simulation model size and longer simulation runtime.

The **strengths** of the Simul8 simulation software package are lower acquisition cost, no annual licencing fees, more simplistic modelling environment (and thus easier to learn and use), inclusion of an internal logic programming language, smaller simulation model size and shorter simulation runtime. The use of spreadsheet variables as input and output mechanisms enhance user-friendliness and therefore the ease of use of Simul8 simulation models is also a strength of Simul8.

The **weaknesses** of the Simul8 simulation software package are a more basic graphics capability and less modelling capabilities. The *Simul8 Standard* package only provides five building blocks but the inclusion of Visual Logic allows great modelling freedom and creativity.

The random number generation functionality of the Arena and Simul8 simulation software packages was also investigated. A string of random numbers was generated with both packages and then subjected to a statistical random number test. The random number generation test and the results are detailed in Appendix P: *Random Number Generation Test*. Both packages passed the test of randomness with a significance level of 95%.

Summary

In this section the ED evaluation method option Arena and Simul8 simulation models and the Arena and Simul8 simulation software packages are compared. It is indicated that the simulation runtimes of the ED evaluation method option simulation models represent an approximate fortyfold improvement over the simulation runtime of the original simulation model. The strengths and weaknesses of the simulation software packages are also discussed.

* * * * *