CHAPTER 2

METHODOLOGY CONCEPTUALISATION

INTRODUCTION

Simply stated, the purpose of this chapter is to conceptualise the generic simulation modelling methodology. It is imperative to have a clear understanding of precisely what has to be achieved and how it should be attained, before any attempt is made to begin with the physical process by which the desired goal has to be achieved.

A simulation modelling method or methodology is usually developed with a specific class or type of system in mind. Therefore the first section identifies the characteristics of the class or type of system that is considered in this document. The key characteristics of these systems are the following: continuous processes, two types of discrete events (*i.e.* the services and failures) and complex interrelationships.

In the second section the implications of these characteristics on a simulation model are explored. Different techniques are considered and two possible candidates emerge, namely: a technique that uses variables to represent the process flow in a simulation model and a technique that uses a fixed time interval to advance the simulation model in time. Equations are developed to determine the maximum possible throughput of the Synthetic Fuel plant, as a function of time, and also the number of modules that is switched on or off in each of the smaller plants to achieve that throughput, as a function of time. The determination of the maximum possible throughput is no arbitrary task because of the presence of feedback-loops, the division of the output of the Steam and Oxygen plants and the fact that the number of available modules in each of the smaller plants is a function of time.

The Entity-represent-module (ERM) method is described in the third section. The ERM method was originally developed as part of the *Magister* research and is used by both the original simulation modelling method and the generic simulation modelling methodology. It is an innovative method that determines the number of available modules in each of the smaller plants at any given moment in time. The concept of the ERM method is counter-intuitive because it uses entities to represent the modules rather than the cumbersome *Servers* or *Work Centers* that are usually used in simulation software packages. It leads to a compact simulation model size, total control over all the aspects of the services and accuracy. Each of the smaller plants is represented by three separate parts (*i.e.* the Availability, Service and Failure parts) that are combined to form a high-level building block. Four types of smaller plants are represented in the ERM method by high-level building blocks (*i.e.* a smaller plant with a multiple service cycle and failures of the modules, a smaller plant with a service cycle and failures of the modules, a smaller plant with a service cycle of the modules and a smaller plant with failures of the modules). The advanced version of the ERM method (*i.e.* the one used by the generic methodology) is more compact and accurate than the original version (*i.e.* the one used by the original method).

The Fraction-comparison (FC) method is detailed in the fourth section. The FC method is the most important innovation of the generic simulation modelling methodology and can be considered as the "jewel in the crown" of the generic methodology. It is an elegant method that identifies the momentary "bottleneck" in a complex system at any given moment in time. The FC method is based on the fact that the actual output throughput values of the possible "bottleneck" points at any given moment in time are in fixed relations in terms of one another for all possible throughput options of the system that is under scrutiny. The fixed relations are expressed as the steady state actual output throughput values of the possible "bottleneck" points and are referred to as the FC method parameter set. The parameter set is unique for every specific system description of the system that is under scrutiny. The FC method provides a solution to one of the major problem areas of the generic methodology.

The determination of the governing parameters is detailed in the fifth section. The governing parameters are the gas-feedback-loop-fraction, steam-division-ratio, oxygen-division-ratio and the FC method parameter set. An iterative-loop technique is detailed that uses a FORTRAN software programme called PSCALC.FOR to determine the governing parameters of the Synthetic Fuel plant for the system description that is provided in Section 1.2.

The sixth section considers techniques to identify the "bottleneck" smaller plants in the system that is under scrutiny. The original simulation modelling method uses the throughput utilisation values of the smaller plants to identify the "bottleneck" smaller plants. A distinction is made between primary and secondary "bottlenecks". Two techniques are introduced to identify the primary "bottlenecks". The first technique identifies the primary "bottlenecks" based on the time that the smaller plant is the "bottleneck" and the second technique identifies the primary "bottlenecks" based on the production that is lost due to the smaller plant. Flared throughput indicates the existence of a secondary "bottleneck".

The last section conceptualises the structure of the generic simulation modelling methodology. The seven methods and techniques that are developed in the previous sections are integrated to form the generic methodology. The generic methodology is divided into two separate parts. The iterative-loop technique part determines the governing parameters before the start of a simulation run and the simulation model part uses the six other methods and techniques continuously during the simulation run. The simulation model itself is divided into a "virtual" part that deals with the continuous processes and the functioning of the simulation model and a "real" part that deals with the behaviour of the modules. The "virtual" part is represented in the simulation model by the logic engine high-level building block and the "real" part is represented by the four different high-level building blocks of the ERM method. The five high-level building blocks can be used to construct simulation models of stochastic continuous systems. Simulation models that are developed with the generic methodology do not need a warm-up period and the advantages of this feature are also highlighted.

* * * * *

2.1 SYSTEM CHARACTERISTICS

In most instances a simulation modelling method or methodology is developed with a specific class or type of system in mind. The term "class" implies a collection of objects that share the same characteristics. A simulation modelling method or methodology therefore usually makes provision for systems with the same characteristics. This concept is also applicable to the generic simulation modelling methodology. It is therefore of cardinal importance to fully understand the characteristics of the Synthetic Fuel plant, as well as their impact on a simulation model, before a generic methodology can be conceptualised and developed.

Although the discussions in the rest of this chapter use the Synthetic Fuel plant as an example, it is important to realise that all the concepts are equally applicable to all systems of the class or type of system that is considered in this document.

From the system description of the Synthetic Fuel plant that is provided in Section 1.2, the following key characteristics of systems that belong to the class or type of system that is considered in this document, can be identified:

a) The systems are continuous process systems.
b) The systems are subject to two types of discrete events:
   i) Chronological events (services).
   ii) Stochastic events (failures).
c) The systems have complex interrelationships.

The following three paragraphs provide more detail about the characteristics of this class or type of system. Such systems are commonly referred to as stochastic continuous systems to accentuate their two most important characteristics.

Section 1.2 indicates that the motion of the "commodities" (coal, gases and liquids) in the Synthetic Fuel plant can be characterised as flow and therefore the process of the plant is characterised as continuous.

The Synthetic Fuel plant is subject to chronological and stochastic events. The services of the modules are strictly chronological events and are characterised by the service cycles of the modules (see Section 1.2 and Table A2). The failures of the modules are stochastic events and are characterised by the failure characteristics of the modules (see Section 1.2 and Table A2).

The complex interrelationships of the Synthetic Fuel plant are manifested in both the process flow and the process logic of the plant. The system description of the process flow indicates that there are several feedback-loops and that the output of both the Steam and Oxygen plants is divided (see Section 1.2 and Table A1). The process logic (rules of operation) of the plant indicates the complexity of the interrelationships between the smaller plants (see Section 1.2 and Appendix B). The continuous nature of the process of the plant implies that all 147 modules are, in a way, intrinsically interlinked as far as the effect of the service or failure of a module is concerned. Any breakdown in the processing capacity at one point because of the service or failure of a module, does have an immediate effect on upstream and downstream operations.

The fact that these characteristics have to be accommodated in a simulation model that conforms to the design criteria that are stated in Section 1.5 poses the main problem of the generic simulation modelling methodology.

The complexity of the main problem, when viewed in its entirety, seems overwhelming. This challenge, however, can be approached in a meaningful way by segregating the main problem into appropriate smaller manageable units or subproblems and then solving each of them individually. The rest of this chapter identifies the subproblems through the process of logical deduction and then identifies and develops methods and techniques that solve the various problems that are posed by the subproblems.

Leedy (1993:71) postulates that the main research problem usually consists of two to six subproblems and advocates that subproblems should not be confused with pseudo-subproblems. He defines pseudo-subproblems as procedural indecisions and indicates, for example, that the problem to determine the correct sample size is a pseudo-subproblem, because there are various techniques available to determine sample sizes and it is only necessary to identify the correct one to use for each specific application.

In this chapter the terms "method" and "technique" are also used in accordance with the convention that is explained in Section 1.1 concerning the hierarchy of terminologies that are proposed by van Dyk (2001:2-4). According to the convention the term "method" is perceived to be indicative of a higher order terminology, while the term "technique" is perceived to be indicative of a lower order terminology. Hence, the term "method" is used to indicate a "tool" that is used to solve a more complex subproblem and the term "technique" is used to indicate a "tool" that is used to solve a less complex subproblem.

Summary

The characteristics of the class or type of system that is considered in this document are identified in this section. The key characteristics of these systems are continuous processes, two types of discrete events (chronological and stochastic) and complex interrelationships.

* * * * *

2.2 IMPLICATIONS OF THE CHARACTERISTICS

Section 1.6 indicates that some authors propose that continuous phenomena can be accommodated by using discrete-event modelling techniques. Harrell and Tumay (1999:35) propose two possible techniques that both use discrete-event modelling techniques to deal with continuous phenomena. The first technique suggests that continuously flowing "commodities" can be converted into discrete entities or "packages" for the purpose of a simulation model. For example, the maximum possible raw gas output throughput of the Gas Production plant is 1596000 nm^3/h (40 modules with an output capacity of 39900 nm^3/h each). This can be converted into 100 discrete raw gas "packages" of 15960 nm^3 each for the purpose of a simulation model, if it is assumed that each raw gas "package" represents 1% of the maximum possible raw gas output throughput. If each raw gas "package" is delayed in a simulation model for 36 seconds (one hour divided by 100) as it leaves the Gas Production plant, then the simulation model simulates a raw gas output throughput of 1596000 nm^3/h (100 "packages" of 15960 nm^3 each leaves the Gas Production plant in one hour).

The following two major concerns immediately become apparent if the example that is mentioned in the previous paragraph is implemented in a simulation model:

a) The first concern is that the maximum possible accuracy with which the raw gas output throughput of the Gas Production plant can be determined, has been reduced to the size of a raw gas "package" per hour (*i.e.* 15960 nm^3/h) or alternatively 1% of the maximum possible raw gas output throughput. The resolution of an answer that indicates the raw gas output throughput therefore cannot be any better than the size of a raw gas "package".
b) The second concern is that 100 entities (raw gas "packages") leave the Gas Production plant during one hour of simulated time. This implies that 100 events (delays of raw gas "packages") occur at that point in the simulation model during one hour of simulated time. It also implies that over a simulated time period of one year a staggering 864000

events (assume an 8640-hour simulation model year - see Appendix L: *Synthetic Fuel Plant Simulation Model Year*) occur at that point in the simulation model.

The accuracy can obviously be improved by converting the maximum possible raw gas output throughput into more discrete "packages". For instance, a conversion into 200 discrete "packages" will result in an accuracy resolution of ½% of the maximum possible raw gas output throughput. Paradoxically, this implies that the number of events at that point in the simulation model now doubles. This clearly represents a Scylla and Charybdis situation where the choice lies between "*two dangers such that avoidance of one increases the risk from the other.*" (*The Oxford Compact English Dictionary*, 1996:917; Macrone, 1999:20-21).

Kelton *et al.* (1998:353) also propose a variation on this technique and they indicate that it is usually preferred because it results in fewer entities in the simulation model. The variation on the technique uses a single entity that is looped through a time delay and increases a variable that represents the raw gas output throughput with a fixed amount (*i.e.* the discrete "package" size) each time a loop is completed. The problem is that this variation on the technique does not address the accuracy and huge number of events in the simulation model concerns that are detailed in the previous paragraphs.

The diminished accuracy and huge number of events that characterise this technique clearly violate some of the design criteria of the generic simulation modelling methodology that is stated in Section 1.5. The concession on accuracy obviously impacts negatively on the accurate modelling ability design criterion. The huge number of events in a simulation model that uses this technique affects the short simulation runtime criterion directly and the short development and maintenance times criteria indirectly, because longer simulation runtimes impact negatively on simulation model development, maintenance and use. The violation of the design criteria leads to an untenable situation. It emphatically disqualifies this technique as a contender to feature in the generic methodology.

The second of the two techniques that are proposed by Harrell and Tumay (1999:35) holds more promise. The technique simply updates "*a variable* [like the raw gas output throughput of the Gas Production plant] *at regular time intervals that accounts for a constant rate of change that occurred over the interval.*" The second technique updates the variable with a real number amount as opposed to the first technique that updates the variable with an amount that is a multiple of the size of the discrete "package" that is used. It is therefore quite obvious that the second technique is much more accurate than the first one. With the second technique there is also only one event every time interval to update the variable. That implies that if a time interval of one hour is used there are only 8640 events at that point in the simulation model over a simulated time period of one year. That is a hundredfold improvement on the 864000 events at that point in the simulation model if the first technique is used.

Pegden *et al.* (1995:431-464) indicate that continuous behaviour can also be represented in a simulation model by algebraic, differential and difference equations that describe the behaviour of the system in terms of states and rates (see Section 1.7). The behaviour of the system is obtained by solving these equations over time. Unfortunately differential equations are very difficult to solve mathematically and numerical techniques are usually used to obtain solutions. If a numerical technique is used, the simulation model is advanced in time by a series of small time intervals. The size of each small time interval is calculated individually and is determined by the required accuracy. The numerical technique actually divides the continuous behaviour of the system into behaviour at discrete points in time. The state of the system is calculated at each of these discrete points in time and the total behaviour of the system over a period of time follows from the summation of the behaviour at the discrete points in time.

An interesting variation on the numerical technique described in the previous paragraph uses a fixed time interval to advance in time. The size of the fixed time interval depends on the required accuracy and is usually chosen in accordance with the dynamic response characteristics of the system that is modelled. The advantage of this variation is that the size of each time interval does not have to be calculated and therefore a lot of processing time is saved during a simulation run. The simulation model is also simpler because some of the numerical techniques are rather cumbersome to implement in a simulation model. For example, the 3- and 6-degree-of-freedom simulation models that are used to investigate the performance of aircraft and missile systems use this technique. If the time interval is chosen prudently and correctly in relation to the dynamic response characteristics of the system that is modelled, the result that is obtained can be a very close approximation of the real-world situation that is modelled. This is also the technique that is used by Forrester (*c.*1961:73) in *Industrial Dynamics* where he describes the use of a fixed time interval and how its size is determined.

> "*The equations of the model are evaluated repeatedly to generate a sequence of steps equally spaced in time.*"

> "*The interval of time between solutions must be relatively short, determined by the dynamic characteristics of the real system that is being modeled* [*sic*]."

To summarise, the first part of this section clearly shows that the first of the two techniques that are proposed by Harrell and Tumay leads to low accuracy and a huge number of events and therefore the first technique disqualifies itself. The second technique leads to high accuracy and fewer events and therefore qualifies as an excellent possible candidate for further use. The part that follows indicates that differential equations in simulation models are solved with numerical techniques that advance in time with small time intervals. If a fixed time interval is used, the need to calculate the size of each time interval falls away but care should be taken to ensure that accuracy requirements are not violated.

Section 1.7 shows that the generic simulation modelling methodology does not use differential equations to represent continuous behaviour. The reasons for this omission are also discussed. Even though the generic methodology does not accommodate differential equations, it still stands to reason that the continuous processes of the Synthetic Fuel plant can also be modelled by a simulation model that uses a fixed time interval to advance in time. The size of the fixed time interval should be chosen in accordance with the dynamic response characteristics of the Synthetic Fuel plant.

The following two possible candidates thus emerge as techniques for inclusion into the generic simulation modelling methodology:

a) The one technique proposes the use of variables to represent process flow, like the raw gas output throughput of the Gas Production plant, as real numbers. These variables are updated with real number amounts at regular time intervals to ensure high accuracy.
b) The other technique proposes the use of a fixed time interval to advance the simulation model in time. The size of the fixed time interval depends on the required accuracy and dynamic response characteristics of the system that is modelled.

Section 2.1 indicates that the class or type of system that is considered in this document is subject to two types of discrete events, namely: services and failures. Section 1.6 explains that the development of simulation software packages has traditionally focused primarily on the ability to model discrete-event systems. This implies that there is a plethora of techniques available in various simulation software packages that allow the easy incorporation of discrete events into simulation models.

The complex interrelationship characteristic of the class or type of system that is considered poses a much more formidable problem. The complex interrelationship characteristic is manifested in both the process flow and the process logic. The system description of the Synthetic Fuel plant in Section 1.2 reveals that there are feedback-loops in the plant. Crowe *et al.* (1971:1) provide some insights into the problems that are posed by the recycling (*i.e.* the feedback-loops) of either heat or matter in chemical plants.

> "*... an evaluation can be anything but exiting when it involves the tedious task of long and repetitious calculations caused by the recycle of energy or material.*"

> "*Recycle occurs frequently in chemical plants to conserve material and to improve the overall efficiency. Such recycle, however, introduces calculational* [*sic*] *difficulties.*"

The feedback-loops in the Synthetic Fuel plant are detailed in Point k) of the rules of operation of the plant in Appendix B, but it is important to repeat it here *verbatim* for the sake of the continuity of the argument. Plant(II)-A receives input from three other plants. Plant(II)-A receives pure gas directly from Plant(I), H_2 from the Division Process plant and recycled gas from the Recycling plant. From the Division Process plant there is a direct feedback-loop to Plant(II)-A and there is also an indirect feedback-loop from the Division Process plant through the Recycling plant to Plant(II)-A. The primary input of Plant(II)-A is the pure gas from Plant(I) and it is supplemented by the secondary input that consists of the H_2 and recycled gas from the Division Process and Recycling plants respectively. The volumes of H_2 and recycled gas that are supplied to Plant(II)-A obviously depends on the volume of pure gas that is supplied to Plant(II)-A from Plant(I). The ratio of the pure gas to the pure gas plus the H_2 and the recycled gas is referred to as the gas-feedback-loop-fraction. The gas-feedback-loop-fraction assumes a fixed value for a specific system description.

The system description of the Synthetic Fuel plant also indicates that the output of both the Steam and Oxygen plants is divided. The division of the output of the Steam and Oxygen plants is detailed in Points g) and i) respectively of the rules of operation of the Synthetic Fuel plant in Appendix B. Once again these points are repeated here *verbatim* for the sake of the continuity of the argument.

The output of the Steam plant is divided between three of the smaller plants. Steam is supplied to the Gas Production, Oxygen and Electricity Generation plants. Steam will only be supplied to the Electricity Generation plant once the Gas Production and Oxygen plants have been supplied. The primary function of the Steam plant is to supply steam to the Gas Production and Oxygen plants and the secondary function is to supply steam to the Electricity Generation plant.

The ratio of steam that is supplied to the Gas Production plant to steam that is supplied to the Oxygen plant is referred to as the steam-division-ratio. The steam-division-ratio is a fixed ratio for a specific system description.

The output of the Oxygen plant is divided between two of the smaller plants. Oxygen is supplied to both the Gas Production and Recycling plants. The ratio of oxygen that is supplied to the Gas Production plant to oxygen that is supplied to the Recycling plant is referred to as the oxygen-division-ratio. The oxygen-division-ratio is a fixed ratio for a specific system description.

The previous paragraphs clearly indicate that the gas-feedback-loop-fraction, steam-division-ratio and oxygen-division-ratio assume fixed values for a specific system description. This aspect of the complex interrelationship characteristic therefore implies that a method has to be devised that can render the gas-feedback-loop-fraction, steam-division-ratio and oxygen-division-ratio for every specific system description.

The complex interrelationship characteristic also manifests itself in the operation of the Synthetic Fuel plant. The first rule of operation in Appendix B states that the Synthetic Fuel plant always strives to maintain the maximum possible rate of production or throughput. In their book *The Goal*, Goldratt and Cox (1992:294) stress the importance of the throughput as the definitive measurement of plant performance.

> "*But the important thing is that we, in our plant, have switched to* ***regard throughput as the most important measurement****.* ***Improvement*** *for us is not so much to reduce costs but* ***to increase throughput****.*" [Bold typeface added for emphasis]

The maximum possible throughput of the Synthetic Fuel plant varies over time (*i.e.* it is a function of time) because the modules in the smaller plants are subject to services and failures. It therefore follows that the maximum possible throughput of the Synthetic Fuel plant, as a function of time, needs to be determined by the simulation model. The maximum possible throughput of the total Synthetic Fuel plant, as a function of time, can only be determined once the maximum possible throughput of each of the smaller plants, as a function of time, has been determined.

The maximum possible throughput of each of the smaller plants is a function of time because the modules in the smaller plants are subject to services and failures. The maximum possible throughput of each of the smaller plants, as a function of time, is the number of available modules in the smaller plant, as a function of time, multiplied by the capacity of a module in the smaller plant, as a constant.

$$Throughput_{PltMaxPos}(t) = (n_{PltModAvl}(t))(Capacity_{PltMod}) \quad (ton,m^3,nm^3/h) \qquad (Eq.:2.1)$$

Where:

$Throughput_{PltMaxPos}(t)$	:	The maximum possible throughput of the smaller plant, as a function of time, in ton/h, m^3/h or nm^3/h.
$n_{PltModAvl}(t)$	:	The number of available modules in the smaller plant, as a function of time.
$Capacity_{PltMod}$	:	The input or output capacity of a module in the smaller plant, as a constant, in ton/h, m^3/h or nm^3/h.

The input and output capacities of the modules in each of the smaller plants usually differ (*i.e.* usually the input to output ratios are not equal to one), depending on the chemical processes that are involved. The maximum possible throughput of each of the smaller plants can therefore be expressed as either a maximum possible input throughput (*i.e.* the maximum possible upstream throughput) that depends on the input capacity or a maximum possible output throughput (*i.e.* the maximum possible downstream throughput) that depends on the output capacity.

The number of available modules in each of the smaller plants is a function of time because the modules in the smaller plants are subject to services and failures, both of which display time-dependent behaviour.

$$n_{PltModAvl}(t) = f(Service(t),Failure(t)) \quad (number) \qquad (Eq.:2.2)$$

More specifically, the number of available modules in each of the smaller plants, as a function of time, is the number of modules in the smaller plant, as a constant, minus the number of modules in the smaller plant that is being serviced, as a function of time, and the number of modules in the smaller plant that is being repaired after failure, as a function of time.

$$n_{PltModAvl}(t) = n_{PltMod} - (n_{PltModServ}(t) + n_{PltModFail}(t)) \quad (number) \qquad (Eq.:2.3)$$

Where:

n_{PltMod}	:	The number of modules in the smaller plant, as a constant.

$n_{PltModServ}(t)$	:	The number of modules in the smaller plant that is being serviced, as a function of time.
$n_{PltModFail}(t)$	:	The number of modules in the smaller plant that is being repaired after failure, as a function of time.

The maximum possible throughput of the Synthetic Fuel plant is a function of the maximum possible throughput of each of the smaller plants and therefore also a function of time.

$$Throughput_{SFPltMaxPos}(t) = f(Throughput_{PltMaxPos}(t) \text{ for No.1 ... } n_{Plt}) \quad (ton,m^3,nm^3/h) \quad (Eq.:2.4)$$

Where:

$Throughput_{SFPltMaxPos}(t)$	:	The maximum possible throughput of the Synthetic Fuel plant, as a function of time, in ton/h, m^3/h or nm^3/h.
n_{Plt}	:	The number of smaller plants, as a constant.

The determination of the maximum possible throughput of the Synthetic Fuel plant, as a function of time, is no arbitrary task because of the presence of feedback-loops, the division of the output of the Steam and Oxygen plants and the fact that the number of available modules in each of the smaller plants is a function of time. There is one consolation though. The second rule of operation in Appendix B states that only the smaller plants that form part of the main-gas-cycle can act as “bottlenecks” that influence the maximum possible throughput of the Synthetic Fuel plant. There are 10 smaller plants in the main-gas-cycle and they are sometimes referred to as the “heart” of the Synthetic Fuel plant. Two of the 10 smaller plants consist of groupings of different types of modules. The Oxygen plant consists of three groupings of different types of modules and Plant(II) consists of two groupings of different types of modules. The 10 smaller plants of the main-gas-cycle therefore represent 13 possible separate points, any one of which can be the “bottleneck” that determines the maximum possible throughput of the Synthetic Fuel plant at any given moment in time. The 13 possible “bottleneck” points in the main-gas-cycle are the following: 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 possible “bottleneck” points determine the maximum possible throughput of the Synthetic Fuel plant at any given moment in time. The possible “bottleneck” point that is the “bottleneck” in the main-gas-cycle of the Synthetic Fuel plant at a specific moment in time is referred to as the momentary “bottleneck”. The throughput of the Synthetic Fuel plant at any given moment in time is adjusted to coincide with the maximum possible throughput of the momentary “bottleneck” at that specific moment in time.

More than one of the 13 possible "bottleneck" points can simultaneously be the "bottleneck" at any given moment in time. Such an occurrence is referred to as a multiple momentary "bottleneck". The effect of these multiple momentary "bottleneck" occurrences is taken into account when the "bottleneck" smaller plants in the Synthetic Fuel plant are identified (see Section 2.6). The identification of the momentary "bottleneck" should not be confused with the identification of the "bottleneck" smaller plants. The identification of the momentary "bottleneck" is necessary to determine the maximum possible throughput of the Synthetic Fuel plant at a specific moment in time, while the identification of the "bottleneck" smaller plants are necessary to determine which of the smaller plants are "bottlenecks" over a period of time, typically a year or more.

The maximum possible throughput of the Synthetic Fuel plant at any given moment in time is defined by a "throughput vector" that comprises the actual throughput of each of the smaller plants. The actual throughput of the momentary "bottleneck" at that specific moment in time is, of course, exactly the same as the maximum possible throughput of the momentary "bottleneck" because the Synthetic Fuel plant always strives to maintain the maximum possible throughput. The momentary "bottleneck" represents one of the 13 possible "bottleneck" points in the main-gas-cycle and from there the actual throughput of the other 12 possible "bottleneck" points in the main -gas-cycle at that specific moment in time can be determined, depending on the input and output capacities of the modules in the smaller plants and provided that the gas-feedback-loop-fraction, steam-division-ratio and oxygen-division-ratio are known for that specific system description. If the actual throughput of each of the 13 possible "bottleneck" points in the main-gas-cycle is known, the actual throughput of the rest of the Synthetic Fuel plant at that specific moment in time can be determined. Point c) of the rules of operation in Appendix B indicates 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 that they are referred to as the peripheral plants. The actual throughput of each of the peripheral plants depends on the rules of operation of that specific peripheral plant. For example, Point d) of the rules of operation 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 in the Synthetic Fuel plant, then the portions of the throughput that cannot be processed are flared. This example indicates that the complex interrelationship characteristic is even manifested in the determination of the actual throughput of the peripheral plants.

It is a common convention to express the actual throughput of each of the smaller plants as the actual output throughput (*i.e.* the actual downstream throughput) and not as the actual input throughput (*i.e.* the actual upstream throughput). If this convention is followed, it is only necessary to add the actual input throughput of the total Synthetic Fuel plant (*i.e.* the coal that is supplied to the Coal Processing plant and the water that is supplied to the Water Treatment plant from external sources) to give a complete description of the maximum possible throughput (*i.e.* the "throughput vector") of the Synthetic Fuel plant at any given moment in time.

To summarise, the previous paragraphs indicate that the determination of the maximum possible throughput of the Synthetic Fuel plant, as a function of time, primarily depends on the identification of the momentary"bottleneck" in the main-gas-cycle at any given moment in time. The identification of the momentary "bottleneck" at any given moment in time poses a significant challenge due to the presence of feedback-loops, the division of the output of the Steam and Oxygen plants and the fact that the number of available modules in each of the 13 possible "bottleneck" points is a function of time. This challenge represents one of the significant problem areas of the generic simulation modelling methodology. An elegant solution to this problem is detailed in Section 2.4.

The number of modules that is switched on or off in each of the smaller plants, as a function of time, also has to be determined by the simulation model. All the available modules in the momentary "bottleneck" are switched on and operating at 100% of the module capacity at any given moment in time because of the philosophy of operating the Synthetic Fuel plant at the maximum possible throughput. However, it may not be necessary to switch on all the available modules in the other 12 possible "bottleneck" points for that specific maximum possible throughput of the Synthetic Fuel plant at that specific moment in time.

The number of modules that is switched on in each of the smaller plants, as a function of time, is the actual output throughput of the smaller plant, as a function of time, divided by the output capacity of a module in the smaller plant, as a constant.

$$n_{PltModOn}(t) = (Throughput_{PltActOut}(t)) \,/\, (Capacity_{PltModOut}) \quad (number)$$

$$if \qquad (Throughput_{PltActOut}(t)) \,/\, (Capacity_{PltModOut}) = Integer \quad (number)$$

$$or \qquad (Eq.:2.5)$$

$$n_{PltModOn}(t) = Truncate((Throughput_{PltActOut}(t)) \,/\, (Capacity_{PltModOut})) + 1 \quad (number)$$

$$if \qquad (Throughput_{PltActOut}(t)) \,/\, (Capacity_{PltModOut}) = Real \quad (number)$$

Where:

$n_{PltModOn}(t)$	:	The number of modules that is switched on in the smaller plant, as a function of time.
$Throughput_{PltActOut}(t)$	:	The actual output throughput of the smaller plant , as a function of time, in ton/h, m^3/h or nm^3/h.
$Capacity_{PltModOut}$	:	The output capacity of a module in the smaller plant, as a constant, in ton/h, m^3/h or nm^3/h.

The number of modules that is switched off in each of the smaller plants, as a function of time, is the number of modules that is available in each of the smaller plants, as a function of time, minus the number of modules that is switched on in each of the smaller plants, as a function of time.

$$n_{PltModOff}(t) = n_{PltModAvl}(t) - n_{PltModOn}(t) \quad (number) \qquad (Eq.:2.6)$$

Where:

$n_{PltModOff}(t)$	:	The number of modules that is switched off in the smaller plant, as a function of time.

Even though Equations 2.1 to 2.6 use the term "smaller plant", they are equally applicable when the term "smaller plant" is replaced with the term "possible "bottleneck" point" to accommodate instances where some of the smaller plants consist of groupings of different types of modules.

The following example serves to illustrate what the impact of the complex interrelationship characteristic is on the operation of the Synthetic Fuel plant. Consider an imaginary two-plant system that only involves the Gas Production and Temperature Regulation plants at a specific moment in time. If two of the modules in the Gas Production plant are being repaired after failure and one module in the Temperature Regulation plant is being serviced, then the maximum possible raw gas output throughput of the Gas Production plant is 1516200 nm^3/h (38 of the 40 modules with a raw gas output capacity of 39900 nm^3/h each are available) and the maximum possible raw gas input throughput of the Temperature Regulation plant is 1470000 nm^3/h (seven of the eight modules with a raw gas input capacity of 210000 nm^3/h each are available). The smaller one of the maximum possible raw gas output throughput of the Gas Production plant and the maximum possible raw gas input throughput of the Temperature Regulation plant determines the maximum possible throughput of the imaginary two-plant system. It is obvious that the momentary "bottleneck" in the imaginary two-plant system is the Temperature Regulation plant and that the maximum possible raw gas input throughput of the momentary "bottleneck" is 1470000 nm^3/h. (Assume that the imaginary two-plant system always strives to maintain the maximum possible throughput.)

The maximum possible throughput of the imaginary two-plant system at that specific moment in time, according to the convention previously described, is defined by a "throughput vector" that comprises the actual output throughput (*i.e.* the actual downstream throughput) of each of the two smaller plants as well as the actual input throughput of the imaginary two-plant system. The actual raw gas output throughput of the Gas Production plant is 1470000 nm^3/h. The actual output throughput of the Temperature Regulation plant consists of an actual raw gas output throughput and an actual gas-water output throughput. The actual raw gas output throughput is also 1470000 nm^3/h because the raw gas input and output capacities of the Temperature Regulation modules are identical (1470000 nm^3/h multiplied by the output to input ratio of the raw gas - 210000 nm^3/h divided by 210000 nm^3/h). The actual gas-water output throughput is 940,8 m^3/h (1470000 nm^3/h multiplied by the output to input ratio of the gas-water - 134,4 m^3/h divided by 210000 nm^3/h). The actual input throughput of the imaginary two-plant system is determined in a similar manner. The actual steam input throughput is 954,2 ton/h (1470000 nm^3/h multiplied by the input to output ratio of the steam - 25,9 ton/h divided by 39900 nm^3/h). The actual oxygen input throughput is 203736,8 nm^3/h (1470000 nm^3/h multiplied by the input to output ratio of the oxygen - 5530 nm^3/h divided by 39900 nm^3/h). The actual coarse coal input throughput is 937,6 ton/h (1470000 nm^3/h multiplied by the input to output ratio of the coarse coal - 25,45 ton/h divided by 39900 nm^3/h).

The number of modules that is switched on or off in each of the smaller plants at that specific moment in time depends on the actual output throughput of each of the smaller plants (see Equations 2.5 and 2.6). The Temperature Regulation plant is the momentary "bottleneck" and consequently all seven available modules in the Temperature Regulation plant are switched on and operating at 100% of the module capacity (all the available modules in the momentary "bottleneck" are switched on to ensure that the maximum possible throughput of the imaginary two-plant system is realised). Or alternatively, the actual raw gas output throughput of the Temperature Regulation plant, divided by the raw gas output capacity of a module in the Temperature Regulation plant, gives the number of modules that is switched on in the Temperature Regulation plant. That also gives exactly seven modules in the Temperature Regulation plant that are switched on (1470000 nm^3/h divided by 210000 nm^3/h). The number of modules that is switched on or off in the Gas Production plant is determined in a similar manner. The actual raw gas output throughput of the Gas Production plant, divided by the raw gas output capacity of a module in the Gas Production plant, gives the number of modules that is switched on in the Gas Production plant. That gives an answer of 36,8 modules in the Gas Production plant that are switched on (1470000 nm^3/h divided by 39900 nm^3/h). It is, however, impossible to switch on 36,8 modules and therefore 37 of the 38 available modules in the Gas Production plant are switched on and one is switched off. In reality the workload (*i.e.* the actual raw gas output throughput of the Gas Production plant) is evenly distributed among the 37 modules in the Gas Production plant that are switched on. There will not be 36 modules operating at 100% of the raw gas output capacity of a module and one module operating at 80% of the raw gas output capacity of a module.

To summarise, the example shows that the operation of the imaginary two-plant system at that specific moment in time is described by the following:

a) The number of available modules in each of the smaller plants is (use Equation 2.3):
   - i) Gas Production : 38 modules of a possible 40 modules
   - ii) Temperature Regulation : 7 modules of a possible 8 modules

b) The maximum possible throughput (input or output) of each of the smaller plants is (use Equation 2.1):
   - i) Gas Production : 1516200 nm^3/h raw gas (output)
   - ii) Temperature Regulation : 1470000 nm^3/h raw gas (input)

c) The momentary "bottleneck" of the two-plant system is:
   - i) Temperature Regulation

d) The maximum possible throughput (*i.e.* the "throughput vector") of the two-plant system is (use Equation 2.4):
   - i) Actual input throughput:
     - Steam : 954,2 ton/h
     - Oxygen : 203736,8 nm^3/h
     - Coarse coal : 937,6 ton/h
   - ii) Actual output throughput of the Gas Production plant:
     - Raw gas : 1470000 nm^3/h
   - iii) Actual output throughput of the Temperature Regulation plant:
     - Raw gas : 1470000 nm^3/h
     - Gas-water : 940,8 m^3/h

e) The number of modules that is switched on in each of the smaller plants is (use Equation 2.5):
   - i) Gas Production : 37 modules of the 38 available modules
   - ii) Temperature Regulation : 7 modules of the 7 available modules

f) The number of modules that is switched off in each of the smaller plants is (use Equation 2.6):
   - i) Gas Production : 1 module of the 38 available modules
   - ii) Temperature Regulation : 0 modules of the 7 available modules

This example clearly illustrates the complexities that are involved to determine the maximum possible throughput (*i.e.* the "throughput vector") and the number of modules that is switched on or off to achieve that throughput, for a very simple imaginary two-plant system at a specific moment in time. Therefore, the determination of the maximum possible throughput and the number of modules that is switched on or off to accomplish that throughput, as functions of time, for the entire Synthetic Fuel plant is not a straightforward matter.

The maximum possible throughput (input or output) of each of the smaller plants and the maximum possible throughput of the system (that consists of the actual input throughput of the system and the actual output throughput of each of the smaller plants) are determined as real numbers. In contrast to this, the number of modules that is available in each of the smaller plants and the number of modules that is switched on or off in each of the smaller plants are determined as integer numbers. The representation of the maximum possible throughput of the smaller plants and the maximum possible throughput of the system as real numbers already presupposes the notion of representing continuous processes with variables (*i.e.* the variables technique). It is obvious that the variables technique is more accurate than the techniques that represent continuous processes by converting the continuously flowing "commodities" into discrete entities or "packages" (see the discussion in the first part of this section).

Summary

This section investigates the implications of the characteristics of stochastic continuous systems on a simulation model. The continuous process characteristic leads to two techniques that qualify for possible inclusion into the generic simulation modelling methodology, namely: the use of variables to represent processes and the use of a fixed time interval to advance the simulation model in time. The characteristic of the two types of discrete events (*i.e.* the services and failures) does not represent a significant problem. The complex interrelationship characteristic, however, poses a much more formidable problem. The gas-feedback-loop-fraction, steam-division-ratio and oxygen-division-ratio have to be determined for every specific system description. The complex interrelationship characteristic also influences the operation of the system and therefore the determination of the maximum possible throughput (*i.e.* the "throughput vector") of the system and the number of modules that is switched on or off to achieve that throughput, is no arbitrary matter (as demonstrated by the imaginary two-plant system example).

This section also provides equations for the determination of the maximum possible throughput and the number of modules that is switched on or off to achieve that throughput. However, there are still a few outstanding issues that have to be resolved. In the simple imaginary two-plant system example the number of modules that is being serviced and the number of modules that is being repaired after failure in each of the smaller plants at that specific moment in time is assumed to be known and the identification of the momentary "bottleneck" is very easy with only two possible candidates to choose from. The same does not apply when the entire Synthetic Fuel plant is considered. The identification of the momentary "bottleneck" from the 13 possible "bottleneck" points is not easy because there are feedback-loops, the output of the Steam and Oxygen plants is divided and the number of available modules in each of the smaller plants is a function of time.

The outstanding issues that require further consideration are the following:

a) The determination of the number of modules that is being serviced and the number of modules that is being repaired after failure in each of the smaller plants at any given moment in time. The services and failures are the discrete events and an innovative method to accommodate this characteristic is detailed in Section 2.3. This method is referred to as the Entity-represent-module (ERM) method.

b) The identification of the momentary "bottleneck" from the 13 possible "bottleneck" points at any given moment in time. An elegant method that identifies the momentary "bottleneck" in a complex system is detailed in Section 2.4. This method is referred to as the Fraction-comparison (FC) method

c) The determination of the governing parameters for every specific system description of the system that is under scrutiny. The governing parameters comprise the gas-feedback-loop-fraction, steam-division-ratio, oxygen-division-ratio and the FC method parameter set. The first three follows from the presence of feedback-loops and the fact that the output of the Steam and Oxygen plants is divided and the parameter set is necessary for the FC method to function. The determination of the governing parameters is detailed in Section 2.5.

* * * * *

2.3 THE ERM METHOD

The abbreviation ERM stands for Entity-represent-module. The ERM method is used by both the original simulation modelling method and the generic simulation modelling methodology. It was originally developed as part of the *Magister* work (Albertyn, 1995:42-47). However, the advanced version that is presented in this document, is considerably more refined than the original version. The ERM method is an innovative method that determines the state of the modules in the system that is under scrutiny at any given moment in time. The previous section indicates that the continuous processes can be represented by variables in a simulation model. However, the behaviour of the modules also has to be represented in the simulation model. The modules are subject to discrete events (*i.e.* the services and failures). The differences between the representation of the continuous processes and the representation of the behaviour of the modules lead to a natural division of the simulation model into two parts. One part deals with the continuous processes and the other deals with the behaviour of the modules. The part of the simulation model that deals with the continuous processes is referred to as the "virtual" part of the simulation model, because the actual processes are represented by variables and logical equations (*i.e.* the process flow and process logic or rules of operation are represented by variables and logical equations). The part that deals with the behaviour of the modules is referred to as the "real" part of the simulation model, because the actual modules are represented by standard simulation software package building blocks. This section is primarily concerned with the "real" part of the simulation model that deals with the behaviour of the modules.

The modules in the smaller plants represent the physical processing resources of the Synthetic Fuel plant that actually process the "commodities" (*i.e.* the coal, gases and liquids) that flow through the plant. The modules are subject to two types of discrete events, namely: the services and the failures of the modules (see Section 1.2 and Table A2). The groupings of components that are referred to as modules in this document are usually represented in simulation models by high-level simulation software package building blocks. A high-level building block is a conglomerate of basic building blocks that model a specific concept that occurs frequently in simulation models. For example, a high-level building block can be developed that represents a lathe in a machine shop. Most simulation software packages provide basic building blocks that allow the modeller the freedom to include unique concepts and high-level building blocks that facilitate the use of standardised concepts. The high-level building blocks that represent modules are different in different simulation software packages.

In the Arena simulation software package a module is represented by the *Server* high-level building block on the *Common* template. The *Server* high-level building block "... *defines a station corresponding to a physical or logical location where processing occurs.*" (according to the Arena help function). The *Server* high-level building block makes provision for services (called downtime) and failures. The services are defined by a cycle time and a service time, both of which can be defined by either constant values or theoretical probability distributions. A multiple service cycle can be accommodated, but the start time of a service cycle cannot be specified. The failures are defined by a failure rate and a repair time. The failure rate can be defined by either count (counting the number of occurrences of an event) or time (a constant value or a theoretical probability distribution). The repair time can be defined by a constant value or a theoretical probability distribution.

In the Simul8 simulation software package a module is represented by the *Work Center* high-level building block on the *Build Tools* template. "*A Work Center* [*sic*] *is a place where work takes place on Work Items.*" (according to the Simul8 help function). The *Work Center* high-level building block groups all unavailability (*i.e.* the services and failures) together under a single heading that is called *Efficiency*. The *Efficiency* is defined by a percentage value and an average repair time that is a constant value.

Simul8 is a registered trademark and is usually denoted by Simul8®. However, for the sake of simplicity it will be written simply as Simul8 in this document. Simul8 is a simulation software package from the Simul8 Corporation.

It is clear that the Arena representation of a module is more accomplished and that the Simul8 representation of a module is more basic. The Arena representation, however, still lacks the ability to specify the start time of a service cycle. It therefore seems as if none of the two simulation software packages can adequately represent a module. The services of the smaller plants are characterised by the service cycles of their modules (see Section 1.2 and Table A2). The start times of the service cycles are of critical importance, because the way that the different service cycles of the different smaller plants interact can have a pronounced effect on the throughput of the Synthetic Fuel plant. It is obvious that the two simulation software packages cannot accommodate all the required intricacies of the services.

This deficiency of the simulation software packages led to the development of the ERM method. The only logical solution is to use the basic simulation software package building blocks to develop a high-level building block that does accommodate all the required intricacies of the services. It also presents an opportunity to use the basic building blocks in an innovative manner.

A simulation model usually incorporates the processing resources and the "commodities" that are processed. In a discrete simulation modelling environment the processing resources are usually represented by *Servers* (Arena) or *Work Centers* (Simul8) and the "commodities", that move or flow through the system, are usually represented by entities. The word entity is "*... a generic term used to denote any person, object, or thing—whether real or abstract—whose movement through the system may cause changes in the state of the system.*" (according to the Arena help function). An entity is referred to as an *Entity* in Arena and as a *Work Item* in Simul8. Entities are usually created at specific points in a simulation model and then move or flow through the system while they are processed by various processing resources (*i.e.* the *Servers* or *Work Centers*).

The innovative aspect of the ERM method is that it uses entities to represent the modules. This is a counter-intuitive concept because *Servers* and *Work Centers* usually represent physical processing resources that are "fixed" in position, while the entities usually represent the "commodities" that move of flow through the system. All the relevant information about a module is stored in the attributes of the entity that represents the module. An attribute is referred to as an *Attribute* in Arena and as a *Label* in Simul8. For example, the relevant information about a module, such as the number of the smaller plant that the module belongs to, a grouping number (if the smaller plant consists of groupings of different types of modules), a module number that determines its position in the smaller plant, values that determine its next service and failure, *etc.* can all be stored in the attributes of the entity that represents the module.

The behaviour of a module is characterised by the following four different possible states:

a) On : Available (switched on)
b) Off : Available (switched off)
c) Service : Unavailable (being serviced)
d) Failure : Unavailable (failed and being repaired)

A module is either available or unavailable. An available module is either switched on or it is switched off. An unavailable module is either being serviced or it is being repaired after failure.

The four possible states of a module seem to imply that each of the smaller plants needs four separate parts to deal with the behaviour of the modules in that specific smaller plant. If the first two possible states are combined to form one part, the four separate parts are reduced to three separate parts. In such an instance the first part deals with all the available modules (irrespective of whether they are switched on or off) and the second and third parts deal with the modules that are being serviced and the modules that are being repaired after failure respectively.

The three separate parts of each of the smaller plants can easily be constructed from the basic building blocks of simulation software packages. The first part is very simplistic and consists of only a queue. All the available modules reside in the queue. The second and third parts are more complex and consist of queues, resources and other associated basic building blocks. The resources in the second and third parts represent the human resources of the Synthetic Fuel plant that are necessary to service and repair the modules. The human resources that service the modules are referred to as Service Teams and the human resources that repair the modules are referred to as Repair Teams. There is a dedicated Service Team for each of the smaller plants whose modules are subject to services and a dedicated Repair Team for each of the smaller plants whose modules are subject to failures.

It is important to note that in the ERM method of the original simulation modelling method each of the smaller plants consists of four separate parts, because the ERM method of the original method uses two separate queues to distinguish between the modules that are switched on and those that are switched off. The two queues provide statistics about the number of modules that is switched on or off in the smaller plant over a period of time. In the ERM method of the generic simulation modelling methodology, however, each of the smaller plants consists of only three separate parts because the two queues are combined to form one queue for the available modules. The statistics about the number of modules that is switched on or off in the smaller plant over a period of time is kept by variables. This change helps to support the compact simulation model size design criterion (see Point e) of the design criteria in Section 1.5) of the generic methodology by eliminating one of the queues that is used in each of the smaller plants in the ERM method of the original method. The ERM method of the original method uses four queues in each of the smaller plants, one in each of the four separate parts of each of the smaller plants while the ERM method of the generic methodology uses three queues in each of the smaller plants, one in each of the three separate parts of each of the smaller plants.

The aim of the ERM method is to determine the state of the modules in the system that is under scrutiny at any given moment in time. To reach that goal, it is necessary to construct three separate parts for each of the smaller plants. The first part is referred to as the Availability Part, the second as the Service Part and the third as the Failure Part.

Before the start of a simulation run, the first part (*i.e.* the Availability Part) of each of the smaller plants in the Synthetic Fuel plant is populated with the corresponding correct number of entities. The number of modules in each of the smaller plants is indicated in Column 3 of Table A1. The entities represent the modules in each of the smaller plants. Appropriate values are also assigned to the attributes of each of the modules. Each of the modules is uniquely identified by the number of the smaller plant that the module belongs to, a grouping number (if the smaller plant consists of groupings of different types of modules) and a module number that determines its position in the smaller plant. Values are also assigned to the next-service and next-failure attributes.

The next-service attribute determines when the module is decommissioned for a service. The start time of the service cycle of each of the smaller plants determines when the first module in that specific smaller plant is decommissioned for a service. The other modules in that specific smaller plant are then decommissioned in sequence until the service cycle is completed. The services of the modules are staggered in time to minimise the impact of the services on production. Before the start of a simulation run, the next-service attribute of the first module in each of the smaller plants is assigned the start time value of the service cycle of that specific smaller plant. The next-service attributes of the other modules in that specific smaller plant are then assigned values that are progressively the service time apart to ensure that the services are staggered in time and do not overlap. The start time of the service cycle in each of the smaller plants only controls when the first service cycle starts, from that point the service cycles follow in a regular pattern, one service cycle apart. The cycle times and service times of the smaller plants are indicated in Columns 3 and 4 respectively of Table A2. The start times are not indicated in Table A2 because they can vary significantly from scenario to scenario. A multiple service cycle can easily be accommodated by using different next-service attributes for the different service cycles of the multiple service cycle. The start time values of each of the different service cycles are then assigned to the corresponding next-service attributes before the start of a simulation run.

The next-failure attribute determines when the module is going to fail. Before the start of a simulation run, the next-failure attribute of each of the modules is assigned a value that is sampled randomly from a theoretical probability distribution. The theoretical probability distributions that are used represent the failure rates of the modules in each of the smaller plants (see Section 1.2). The failure rates of the modules in each of the smaller plants are characterised in Column 5 of Table A2.

To summarise, before the start of a simulation run the first part of each of the smaller plants is populated with the correct number of modules and the attributes of the modules are assigned appropriate values for identification purposes, next service, next failure, *etc.*

During a simulation run, at any given moment in time, each of the smaller plants is evaluated to determine the state of the modules in each of the smaller plants. The next-service attributes of all the modules in the first part (*i.e.* the Availability Part) of each of the smaller plants are first evaluated to determine if any of them are due for a service. If any of them are due for a service they are removed from the first part of the smaller plant and sent to the second part (*i.e.* the Service Part) of the smaller plant, provided that the Service Team of that specific smaller plant is available at that specific moment in time. To ensure the maximum possible throughput of the Synthetic Fuel plant a module will not be decommissioned for a service while another module is still being serviced. The services of the modules in each of the smaller plants do not overlap if they are assigned correctly, but this rule is necessary because if the service schedule of a smaller plant consist of a multiple service cycle, the services of the modules can overlap. The service cycles of a multiple service cycle are prioritised, with the service cycle having the longest service time, taking precedence. It is assumed that the service cycle with the longest service time is the most important service cycle. The next-failure attributes of all the modules in the first part (*i.e.* the Availability Part) of each of the smaller plants are then evaluated to determine if any of them have failed. If any of the modules in each of the smaller plants have failed, they are removed from the first part of the smaller plant and sent to the third part (*i.e.* the Failure Part) of the smaller plant. It is not necessary to determine if the Repair Team of that specific smaller plant is available at that specific moment in time because a failed module is immediately removed from the first part and placed in a queue to await repair if the Repair Team is still busy repairing another module at that specific moment in time.

Modules that arrive at the second part (*i.e.* the Service Part) of each of the smaller plants pass through a queue and are then delayed for a time period that is equal to the service time of that specific service. The service times of the services of each of the smaller plants are indicated in Column 4 of Table A2. The Service Team of that specific smaller plant is also engaged for that time period. Strictly speaking a queue is not necessary because a module is not removed from the first part if the Service Team is not available. However, the queue is advantageous because the statistics of the queue indicates whether modules had to wait in the queue for their services and therefore it can be used to verify that the simulation model works correctly. If modules had to wait in the queue for their services, the simulation model is obviously not working correctly. When the service is completed, the Service Team is disengaged, the number of services that is completed is incremented by one, the next-service attribute is assigned a value that corresponds to the cycle time of the appropriate service cycle, the next-failure attribute is assigned a value that is sampled randomly from the appropriate theoretical probability distribution and the module is returned to the first part of that smaller plant. The number of services that is completed is used for simulation model verification and validation purposes (see Section 3.6). The cycle times of the service cycles of the smaller plants are indicated in Column 3 of Table A2. The next-failure attribute is assigned a new value because it is assumed that the module is restored to an "approximately as good as new" configuration by the preventive maintenance of the service.

Modules that arrive at the third part (*i.e.* the Failure Part) of each of the smaller plants are placed in a queue if the Repair Team of that specific smaller plant is engaged, or pass through the queue if the Repair Team is available. If the Repair Team is available, the modules are delayed for a time period that is sampled randomly from a theoretical probability distribution. The theoretical probability distributions that are used represent the repair times of the modules in each of the smaller plants and are characterised in Columns 6, 7 and 8 of Table A2 (see Section 1.2). The Repair Team of that specific smaller plant is also engaged for that time period. When the repair is completed, the Repair Team is disengaged, the number of failures that is repaired is incremented by one, the next-failure attribute is assigned a value that is sampled randomly from the appropriate theoretical probability distribution and the module is returned to the first part of that smaller plant. The number of failures that is repaired is used for simulation model verification and validation purposes (see Section 3.6).

To summarise, during a simulation run, at any given moment in time, the Availability Part of each of the smaller plants is evaluated to determine the state of the modules in each of the smaller plants. Modules that are due for a service are removed from the Availability Part, sent to the Service Part, delayed for the service time, assigned new values to the appropriate attributes and returned to the Availability Part. Modules that have failed are removed from the Availability Part, sent to the Failure Part, delayed for the repair time, assigned new values to the appropriate attributes and returned to the Availability Part.

The basic structure of the three separate parts of each of the smaller plants is graphically depicted in Figure 2.1: *Smaller Plant Parts*.

The three separate parts of each of the smaller plants therefore identify the number of modules that is available, being serviced, and being repaired after failure, in each of the smaller plants at any given moment in time. The number of modules that is available in each of the smaller plants at any given moment in time is of special importance, because it is used to determine the maximum possible throughput of the smaller plants and hence the maximum possible throughput of the system that is under scrutiny at any given moment in time.

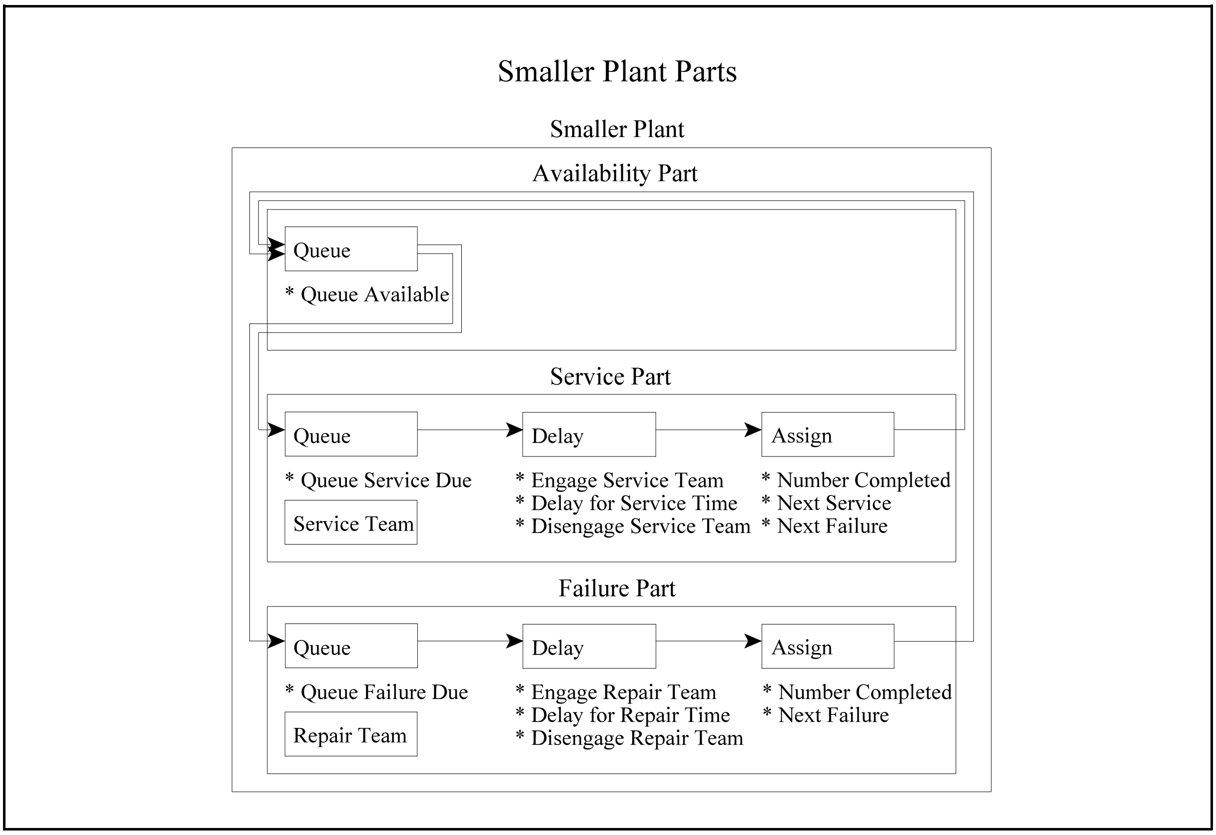

Figure 2.1: Smaller Plant Parts

It is important to realise that a module is in one of the three queues at any given moment in time. This leads to the interesting phenomenon that a module may be in the Failure Part of a smaller plant when its next service is due. It is obvious that the required service cannot start at the scheduled time because the module is still being repaired.

In such an instance one of the following options is applicable:

a) The module is released from the Failure Part before the time that the service would have been completed and consequently the module is immediately sent to the Service Part for the remainder of the service time. Such an event is counted as a completed service and the next-service attribute is assigned a value that corresponds to the cycle time of the appropriate service cycle in exactly the same manner as a regular service. The next-failure attribute is also assigned a new value because the module has been restored to an "approximately as good as new" configuration.

b) The module is released from the Failure Part after the time that the service would have been completed and consequently the module is sent to the Availability Part. Such an event is counted as a missed service and the next-service attribute is assigned a value that ensures that the specific module is serviced next at exactly the right time to be in its original service sequence with respect to the other modules of that specific smaller plant. The next-failure attribute is not assigned a new value because a new value has already been assigned to the next-failure attribute when the module left the Failure Part.

In this instance the service is not really partially completed or missed in the real-world situation, because in the real-world situation the Service Team moves to the module that is stationary. In the real-world situation the Service and Repair Teams of a smaller plant can both be working on one module at the same time. The elaborate approximation that is described above is necessary because it is impossible to emulate the concept of both teams working on one module at the same time in the ERM method. In the ERM method the module is moved and it can only be in one part at any specific moment in time, either in the Availability, Service or Failure Part.

Another interesting phenomenon that occurs is that a module may be due for its next service while another module is still being serviced. This phenomenon can only occur in a smaller plant with a service schedule that consists of a multiple service cycle because the services of the different service cycles may overlap. In a smaller plant with a regular service cycle the services cannot overlap if they are assigned correctly. It is obvious that the required service cannot start at the scheduled time because another module is still being serviced.

In such an instance one of the following options is applicable:

a) The module that is being serviced is released from the Service Part and returned to the Availability Part before the time that the service of the module that is due for its next service would have been completed. Consequently the module that is due for its next service is immediately removed from the Availability Part and sent to the Service Part for the remainder of the service time of the specific service cycle, provided that another module is not due for its next service in a service cycle that has a higher priority than the service cycle of the original module that is due for its next service. Such an event is counted as a completed service of the specific service cycle and the next-service attribute of that specific service cycle is assigned a value that corresponds to the cycle time of that specific service cycle in exactly the same manner as a regular service of that specific service cycle. The next-failure attribute is also assigned a new value because the module has been restored to an "approximately as good as new" configuration.

b) The module that is being serviced is released from the Service Part and returned to the Availability Part after the time that the service of the module that is due for its next service would have been completed. Consequently the module that is due for its next service is not removed from the Availability Part. Such an event is counted as a missed service and the next-service attribute of the specific service cycle is assigned a value that ensures that the specific module is serviced next at exactly the right time to be in its original service sequence with respect to the other modules of that specific smaller plant, as far as that specific service cycle is concerned. The next-failure attribute is not assigned a new value because the service has been missed and the module has not been restored to an "approximately as good as new" configuration.

Even though these phenomena do not occur very frequently, they disturb the service sequences of the modules in the smaller plants when they occur. The disturbances are more pronounced in longer simulation runs because once a service sequence is disturbed the effect is repeated every service cycle that follows from that point onward. A disturbed service sequence looks like a "row of teeth" with one or more of the "specimens" conspicuously missing when viewed graphically on a time graph of the Service Team utilisation. The number of services that is missed is used for simulation modelling verification and validation purposes.

The discussion in the previous paragraphs clearly illustrates the complexities that are involved when the services and the failures of the modules are modelled. Another compounding factor is that the values of the next-service attributes of the modules sometimes start to deviate from the correct values in longer simulation runs because of the accumulation of floating-point errors in the calculations. A floating-point error is a very small error that affects the value of a real variable in a digital computer when a multitude of calculations are done with that real variable because the computer, of necessity, has to approximate each real number with a fixed number of decimal digits. Therefore it is necessary to incorporate mechanisms that continuously test the values of the next-service attributes of the modules and immediately correct them if they start to deviate from the correct values.

To summarise, the following three phenomena can cause a disturbed service sequence:

a) A module is in the Failure Part when its next service is due.

b) A module is due for its next service while another module is still being serviced (overlapping service cycles of a multiple service cycle).

c) The value of the next-service attribute of a module starts to deviate because of the accumulation of floating-point errors.

It is essential to note that the ERM method of the original simulation modelling method does not make provision for the occurrence of these phenomena and consequently the service sequences of the modules in some of the smaller plants are disturbed during longer simulation runs. This leads to a minor inaccuracy as far as the effect of the services on the throughput of the Synthetic Fuel plant is concerned. The previous paragraphs indicate how this shortcoming is addressed in the ERM method of the generic simulation modelling methodology. Therefore the service sequences of the modules in all the smaller plants are always correct when the ERM method of the generic methodology is used, irrespective of the length of the simulation run.

The advantages of the ERM method are the following:

a) The ERM method greatly reduces the size of the simulation model because the three separate parts of each of the smaller plants are constructed from basic simulation software package building blocks and no high-level building blocks are used. In the instance of the Synthetic Fuel plant 147 *Servers* or *Work Centers* (high-level building blocs) are needed to represent the modules if a conventional simulation modelling method is used. If the ERM method is used, 147 entities (which may be regarded as basic building blocks) are needed to represent the modules. Sometimes it is not even necessary to represent a module with an entity, depending on the type of smaller plant that is represented. This concept is clarified in the latter part of this section. Even though queues, resources and other associated basic building blocks are used in the three separate parts of each of the smaller plants, the size of an ERM method simulation model is significantly less than that of a conventional simulation model because no high-level building blocks are used in the ERM method. This aspect of the ERM method therefore supports the compact simulation model size criterion of the generic simulation modelling methodology (see Point e) of the design criteria in Section 1.5).

b) The ERM method allows total control over all the relevant aspects of the services of the modules, namely: the start time, the cycle time and the service time. In most instances it is impossible to achieve this level of control or accuracy if the high-level building blocks of simulation software packages are used. This aspect of the ERM method therefore supports the accurate modelling ability criterion of the generic simulation modelling methodology (see Point g) of the design criteria in Section 1.5).

c) The inclusion of techniques to handle the "disturbed service sequence" phenomena enhances the accuracy of the ERM method. Therefore it also supports the accurate modelling ability of the generic simulation modelling methodology (see Point g) of the design criteria in Section 1.5).

The three separate parts of each of the smaller plants can, of course, be combined to form an ERM method high-level building block for each of the smaller plants. A scrutiny of Table A2 indicates that all the smaller plants are not necessarily subjected to both services (with either a regular or a multiple service cycle) and failures.

The following five different types of smaller plants can be identified:

a) A smaller plant with a multiple service cycle and failures of the modules.
b) A smaller plant with a service cycle and failures of the modules.
c) A smaller plant with a service cycle of the modules.
d) A smaller plant with failures of the modules.
e) A smaller plant with neither a service cycle nor failures of the modules.

It is obvious that all the types of smaller plants can be represented by one high-level building block if it includes a multiple service cycle and failures of the modules. It is also clear that the fifth type of smaller plant does not need to be represented by a high-level building block because all the modules in such a smaller plant are available all the time. That leaves two possible options, namely: use only one high-level building block to represent the first four types of smaller plants, or use four different high-level building blocks to represent the first four types of smaller plants. The first option supports the user-friendliness criterion (see Point c) of the design criteria in Section 1.5) of the generic simulation modelling methodology. It introduces simplicity because only one high-level building block is used, learnt and understood (*i.e.* a standardisation principle). However, the simulation model size suffers because unnecessary and unused options are included. The second option supports the compact simulation model size criterion (see Point e) of the design criteria in Section 1.5) of the generic methodology by not including any options that are unnecessary or unused. However, user-friendliness suffers a bit because four different high-level building blocks are used. This once again presents a Scylla and Charybdis situation where the avoidance of the problems of one option leads to the problems of the other. In this instance the compact simulation model size is deemed more important and therefore four different high-level building blocks (representing the first four types of smaller plants) are used in the ERM method. The fifth type of smaller plant is incorporated into the "virtual" part of the simulation model where the actual processes are represented by variables and logical equations only. The four different high-level building blocks are used to represent the first four types of smaller plants in the "real" part of the simulation model.

Summary

This section details an innovative method that determines the state of the modules in the system that is under scrutiny at any given moment in time. The ERM method uses entities to represent the modules rather than the cumbersome *Servers* or *Work Centers* that are usually used in simulation software packages. The relevant information about a module is stored in the attributes of the entity that represents the module. Each of the smaller plants is represented by three separate parts, namely: the Availability, Service and Failure Parts. Before the start of a simulation run the Availability Part of each of the smaller plants is populated with the correct number of modules and the attributes of the modules are assigned appropriate values. During a simulation run, at any given moment in time, each of the smaller plants is evaluated to determine the state of the modules in the Availability Part. Modules that are due for a service are removed and sent to the Service Part while modules that have failed are removed and sent to the Failure Part. The services and failures are governed by complex rules. The main advantages of the ERM method are a compact simulation model size, total control over all the relevant aspects of the services and accuracy. The number of modules that is available in each of the smaller plants at any given moment in time is used to determine the maximum possible throughput of the smaller plants and hence the maximum possible throughput of the system at any given moment in time.

The three separate parts of each of the smaller plants are combined to form a high-level building block. Four types of smaller plants are represented in the ERM method by the following four different high-level building blocks: a smaller plant with a multiple service cycle and failures of the modules, a smaller plant with a service cycle and failures of the modules, a smaller plant with a service cycle of the modules and a smaller plant with failures of the modules. The four different high-level building blocks are used to represent all the smaller plants in the "real" part of the simulation model and the fifth type of smaller plant (with neither a service cycle nor failures of the modules) is incorporated into the "virtual" part of the simulation model, where the actual processes are represented by variables and logical equations only.

The ERM method of the generic simulation modelling methodology is more compact and accurate than the ERM method of the original simulation modelling method because it reduces the number of queues that is used in each of the smaller plants from four to three and it introduces techniques that address the "disturbed service sequence" phenomena. The ERM method of the generic methodology is referred to as the advanced version and the ERM method of the original method is referred to as the original version.

* * * * *

2.4 THE FC METHOD

The abbreviation FC stands for Fraction-comparison. The FC method is the most important innovation of the generic simulation modelling methodology. It is an elegant method that identifies the momentary "bottleneck" in a complex system at any given moment in time. Section 2.2 indicates that the determination of the maximum possible throughput of the Synthetic Fuel plant, as a function of time, primarily depends on the identification of the momentary "bottleneck" in the main-gas-cycle at any given moment in time. The identification of the momentary "bottleneck" in the Synthetic Fuel plant at any given moment in time is no arbitrary exercise due to the presence of feedback-loops, the division of the output of the Steam and Oxygen plants and the fact that the number of available modules in each of the 13 possible "bottleneck" points is a function of time. This significant challenge represents one of the major problem areas that has to be addressed by the generic methodology.

The entire FC method is based on the simple fact that the actual output throughput values at any given moment in time of the 13 possible "bottleneck" points in the main-gas-cycle are in fixed relations in terms of one another for all possible throughput options of the Synthetic Fuel plant. Section 2.2 indicates that it is a common convention to express the actual throughput of each of the smaller plants as the actual output throughput (*i.e.* the actual downstream throughput). The statement concerning the 13 possible "bottleneck" points at the beginning of this paragraph is based on the assumption that the input to output ratios of all the smaller plants are constant for all possible throughput options of the Synthetic Fuel plant.

This assumption is not necessarily true for all chemical processes but it is a valid assumption in this instance, because of the following:

a) It can be justified by the fact that the requirement is for a decision support tool on a strategic level, not a detail level (see the explanation of strategic versus detail level in Section 1.1).

b) The managers of plants usually strive towards the maximisation of the throughput and therefore the bandwidth of variation in the maximum possible throughput of the Synthetic Fuel plant is generally restricted to a small range (typically less than 10% of the total range of the maximum possible throughput of the plant). The small range of variation in the maximum possible throughput of the Synthetic Fuel plant justifies the assumption that the input to output ratios of the smaller plants are constant over that range.

The validity of this assumption is proved in Sections 3.6, 3.7 and 4.3 by the verification and validation of the Arena and Simul8 simulation models that are developed in Chapter 3.

The fixed relations of the actual output throughput values of the 13 possible "bottleneck" points in the main-gas-cycle depend on the specific system description of the Synthetic Fuel plant (see Section 2.2). The fixed relations are expressed as the actual output throughput values of the 13 possible "bottleneck" points when the Synthetic Fuel plant is operating at the steady state maximum possible throughput. The term "steady state" implies that the influence of time has been removed from the system. In this instance the steady state implies that all the modules in all the smaller plants are available. The influence of the time-dependent services and failures are disregarded. At the steady state the Synthetic Fuel plant operates at the maximum possible throughput of the steady state momentary "bottleneck" (or the steady state multiple momentary "bottleneck" if the steady state momentary "bottleneck" consists of more than one of the 13 possible "bottleneck" points). The steady state actual output throughput of the momentary "bottleneck" is, of course, the steady state maximum possible output throughput of the momentary "bottleneck". The steady state actual output throughput of each of the possible "bottleneck" points that do not qualify as the momentary "bottleneck", is less than the steady state maximum possible output throughput of the possible "bottleneck" point. Therefore the fixed relations of the actual output throughput values of the 13 possible "bottleneck" points are defined by the steady state actual output throughput values of the 13 possible "bottleneck" points. The 13 steady state actual output throughput values of the 13 possible "bottleneck" points are referred to as the FC method parameter set of the Synthetic Fuel plant. The FC method parameter set depends on the specific system description of the system that is under scrutiny. The determination of the FC method parameter set of the Synthetic Fuel plant for the system description that is provided in Section 1.2 is detailed in the next section.

If the steady state actual output throughput values of the 13 possible "bottleneck" points in the main-gas-cycle are known, the momentary "bottleneck" at any given moment in time is easily identified. The maximum possible output throughput values of the 13 possible "bottleneck" points at any given moment in time can be determined if the number of available modules in the 13 possible "bottleneck" points at any given moment in time is known. The previous section indicates how the number of available modules in each of the smaller plants is determined.

The maximum possible output throughput of each of the 13 possible "bottleneck" points, as a function of time, is the number of available modules in the possible "bottleneck" point, as a function of time, multiplied by the output capacity of a module in the possible "bottleneck" point, as a constant (see the maximum possible output throughput option of Equation 2.1).

The maximum possible output throughput of each of the 13 possible "bottleneck" points, as a function of time, divided by the steady state actual output throughput of the possible "bottleneck" point, as a constant, gives a fraction value for the possible "bottleneck" point, as a function of time.

$$Fraction_{Plt}(t) = (Throughput_{PltMaxPosOut}(t)) / (Throughput_{PltSSActOut}) \quad (number) \quad (Eq.:2.7)$$

Where:

$Fraction_{Plt}(t)$	:	The fraction value of the smaller plant, as a function of time.
$Throughput_{PltMaxPosOut}(t)$	:	The maximum possible output throughput of the smaller plant, as a function of time, in ton/h, m^3/h or nm^3/h.
$Throughput_{PltSSActOut}$	:	The steady state actual output throughput of the smaller plant, as a constant, in ton/h, m^3/h or nm^3/h.

Even though the discussions in this section use the term "possible "bottleneck" point" to make provision for instances where some of the smaller plants consist of groupings of different types of modules, Equations 2.7 to 2.9 use the term "smaller plant" to maintain commonality with the nomenclature of Equations 2.1 to 2.6.

The fraction value of each of the possible "bottleneck" points at any given moment in time indicates the level of compliance of the possible "bottleneck" point, in terms of the steady state actual output throughput of the possible "bottleneck" point. A fraction value of more than one indicates that the maximum possible output throughput of the possible "bottleneck" point is more than the steady state actual output throughput of the possible "bottleneck" point, a fraction value of one that it is equal to and a fraction value of less than one that it is less.

The fraction values of the 13 possible "bottleneck" points in the main-gas-cycle at any given moment in time can be compared because they are normalised by the division process. The effect of the relative sizes of the maximum possible output throughput values of the 13 possible "bottleneck" points is negated by the normalisation process that turns the relative sizes into dimensionless fraction values. The possible "bottleneck" point with the smallest fraction value is the momentary "bottleneck" (or the multiple momentary "bottleneck" if the smallest fraction value consists of the fraction values of more than one of the possible "bottleneck" points). The smallest fraction value is referred to as the Benben value in reference to the "magical" squat obelisk found in the Egyptian temples of antiquity, because this value is the "magical" value that determines the maximum possible throughput of the Synthetic Fuel plant at any given moment in time. The Benben value is a function of time and it can only assume values that are equal to, or smaller than one.

$$Benben(t) = Smallest(Fraction_{Plt}(t)) \quad (number) \qquad (Eq.:2.8)$$

Where:

$Benben(t)$: The Benben value is the smallest fraction value, as a function of time.

The actual output throughput of the momentary "bottleneck" at any given moment in time is, of course, the maximum possible output throughput of the momentary "bottleneck". The actual output throughput of each of the possible "bottleneck" points, as a function of time, is the Benben value (*i.e.* the smallest fraction value that is also the fraction value of the momentary "bottleneck"), as a function of time, multiplied by the steady state actual output throughput of the possible "bottleneck" point, as a constant.

$$Throughput_{PltActOut}(t) = (Benben(t))(Throughput_{PltSSActOut}) \quad (ton,m^3,nm^3/h) \qquad (Eq.:2.9)$$

Where:

$Throughput_{PltActOut}(t)$: The actual output throughput of the smaller plant, as a function of time, in ton/h, m^3/h or nm^3/h.

The actual output throughput of each of the smaller plants, as a function of time, is used in Equation 2.5 to determine the number of modules that is switched on in each of the smaller plants, as a function of time.

For example, consider the imaginary two-plant system that is used in Section 2.2 to illustrate the impact of the complex interrelationship characteristic on the operation of the Synthetic Fuel plant and revisit it using the FC method to determine the momentary "bottleneck" and the maximum possible throughput.

The imaginary two-plant system consists of the Gas Production and Temperature Regulation plants. The steady state maximum possible output throughput of the Gas Production plant is 1596000 nm^3/h (all 40 modules with an output capacity of 39900 nm^3/h each are available) and the steady state maximum possible output throughput of the Temperature Regulation plant is

1680000 nm^3/h (all eight modules with an output capacity of 210000 nm^3/h each are available). It is obvious that the steady state momentary "bottleneck" is the Gas Production plant. The steady state actual output throughput of the Gas Production plant is therefore equal to the steady state maximum possible output throughput of the Gas Production plant and that is 1596000 nm^3/h. The steady state actual output throughput of the Temperature Regulation plant is also 1596000 nm^3/h because the input to output ratio of the Temperature Regulation plant is one. In this instance the fixed relations of the actual output throughput values of the two possible "bottleneck" plants are easy to determine.

If, at a specific moment in time, two of the modules in the Gas Production plant are being repaired after failure and one module in the Temperature Regulation plant is being serviced, then the maximum possible output throughput of the Gas Production plant is 1516200 nm^3/h (38 of the 40 modules with an output capacity of 39900 nm^3/h each are available) and the maximum possible output throughput of the Temperature Regulation plant is 1470000 nm^3/h (seven of the eight modules with an output capacity of 210000 nm^3/h each are available).

At that specific moment in time the fraction value of the Gas Production plant is 0,950 (1516200 nm^3/h divided by 1596000 nm^3/h) and the fraction value of the Temperature Regulation plant is 0,921 (1470000 nm^3/h divided by 1596000 nm^3/h). The momentary "bottleneck" is identified by the smallest fraction value and that indicates that the Temperature Regulation plant is the momentary "bottleneck". The Benben value is the smallest fraction value and that is 0,921.

At that specific moment in time the actual output throughput of the momentary "bottleneck" (*i.e.* the Temperature Regulation plant) is, of course, the maximum possible output throughput of the Temperature Regulation plant and that is 1470000 nm^3/h or alternatively, the actual output throughput of the Temperature Regulation plant is the Benben value (*i.e.* the smallest fraction value) multiplied by the steady state actual output throughput of the Temperature Regulation plant and that is also 1470000 nm^3/h (0,921 multiplied by 1596000 nm^3/h). The actual output throughput of the Gas Production plant is the Benben value multiplied by the steady state actual output throughput of the Gas Production plant and that is also 1470000 nm^3/h (0,921 multiplied by 1596000 nm^3/h). The actual output throughput values of the Gas Production and Temperature Regulation plants are only equal because the input to output ratio of the Temperature Regulation plant is one.

Section 2.2 indicates that the maximum possible throughput of the imaginary two-plants system at that specific moment in time is defined by a "throughput vector" that comprises the actual output throughput of the Gas Production and Temperature Regulation plants, as well as the actual input throughput of the imaginary two-plant system. This example uses the FC method to determine two components of the "throughput vector". They are the actual raw gas output throughput of the Gas Production and Temperature Regulation plants. If these components are known, the other four components (*i.e.* the actual steam, oxygen and coarse coal input throughput of the Gas Production plant and the actual gas-water output throughput of the Temperature Regulation plant) of the "throughput vector" can be determined with the input to output ratios of the two smaller plants, as shown in the example in Section 2.2.

To summarise, the example shows that the FC method calculations to identify the momentary "bottleneck" and the maximum possible throughput of the imaginary two-plant system at that specific moment in time, are described by the following:

a) The steady state number of available modules in each of the smaller plants is (see Column 3 of Table A1):
   i) Gas Production : 40 modules
   ii) Temperature Regulation : 8 modules
b) The steady state maximum possible output throughput of each of the smaller plants is (use Equation 2.1):
   i) Gas Production : 1596000 nm^3/h
   ii) Temperature Regulation : 1680000 nm^3/h
c) The steady state momentary "bottleneck" of the two-plant system is:
   i) Gas Production
d) The steady state actual output throughput of each of the smaller plants is (*i.e.* the steady state output "throughput vector" of the two plants):
   i) Gas Production : 1596000 nm^3/h
   ii) Temperature Regulation : 1596000 nm^3/h
e) The number of available modules in each of the smaller plants at that specific moment in time is (use Equation 2.3):
   i) Gas Production : 38 modules out of a possible 40 modules
   ii) Temperature Regulation : 7 modules out of a possible 8 modules
f) The maximum possible output throughput of each of the smaller plants at that specific moment in time is (use Equation 2.1):
   i) Gas Production : 1516200 nm^3/h
   ii) Temperature Regulation : 1470000 nm^3/h
g) The fraction values of the smaller plants at that specific moment in time are (use Equation 2.7):

i) Gas Production : 0,950
ii) Temperature Regulation : 0,921

h) The momentary "bottleneck" of the two-plant system at that specific moment in time is:
i) Temperature Regulation

i) The Benben value (*i.e.* the smallest fraction value) is (use Equation 2.8):
i) Benben : 0,921

j) The actual output throughput of each of the smaller plants at that specific moment in time is (use Equation 2.9):
i) Gas Production : 1470000 nm^3/h
ii) Temperature Regulation : 1470000 nm^3/h

It is essential to note that the original simulation modelling method does not use the FC method to identify the momentary "bottleneck" at any given moment in time. The original method uses a FORTRAN subroutine to identify the momentary "bottleneck". The *Magister* dissertation (Albertyn, 1995:48-53) provides a description of the technique that the FORTRAN subroutine uses to identify the momentary "bottleneck". A detail description is unnecessary, but the technique that the FORTRAN subroutine uses to identify the momentary "bottleneck" can be described as a "push-product-forward-until-it-reaches-the-bottleneck" technique that operates in a sequential step by step manner. Section 1.4 indicates that the FORTRAN subroutine has a complex structure and that to a large extent it is not generic. Some changes in the system description of the Synthetic Fuel plant can easily be accommodated by the FORTRAN subroutine through the manipulation of the input files of the original simulation model. However, changes in the system description that concern the configuration, process flow or process logic cannot be accommodated easily. The FORTRAN subroutine consists of approximately two thousand lines of FORTRAN programming code and it has an extremely complex structure because of the presence of feedback-loops, the division of the output of the Oxygen and Steam plants and the fact that the number of available modules in each of the 13 possible "bottleneck" points is a function of time. For example, if feedback-loops are changed (*i.e.* moved, removed or added) or if the rules of operation of the plant are changed, it cannot be accommodated without substantial changes in the FORTRAN subroutine. The FORTRAN subroutine uses the gas-feedback-loop-fraction, the steam-division-ratio and the oxygen-division ratio to determine the momentary "bottleneck". The gas-feedback-loop-fraction, the steam-division-ratio and the oxygen-division ratio are referred to as the governing parameters of the original method and they depend on the specific system description of the system that is under scrutiny.

The FC method, by comparison, uses a parameter set that contains the fixed relations of the steady state actual output throughput values of the possible "bottleneck" points (or smaller plants) to identify the momentary "bottleneck" at any given moment in time. The FC method always operates in exactly the same way, irrespective of any changes in the system description. Changes in the system description are incorporated into the parameter set which is unique for every specific system description. The FC method uses a matrix-based technique for the determination of the momentary "bottleneck" and the actual output throughput values of the possible "bottleneck" points and it is contained in less than one hundred lines of programming code, or the equivalent thereof if basic simulation software package building blocks are used. The FC method parameter set is referred to as the governing parameters of the generic simulation modelling methodology. It depends on the specific system description of the system that is under scrutiny. The FC method does not use the gas-feedback-loop-fraction, the steam-division-ratio and the oxygen-division ratio directly, but their influence on the operation of the Synthetic Fuel plant is incorporated into the parameter set. The determination of the parameter set is detailed in the next section.

The simplicity of the FC method contradicts the complexity of the problem if it is compared to the technique that is used in the original simulation modelling method by the FORTRAN subroutine to determine the momentary "bottleneck". The size of the FC method solution is approximately 5% of the size of the FORTRAN subroutine in the original method. The FC method successfully addresses one of the major problem areas of the generic simulation modelling methodology. It impacts positively on all the design criteria (or simulation model characteristics) of the generic methodology, namely: short development and maintenance times, user-friendliness, short simulation runtimes, compact size, robustness, accuracy and a single software application (see Section 1.5).

Summary

This section describes the FC method. It is an elegant method that identifies the momentary "bottleneck" in a complex system at any given moment in time. The cornerstone of the FC method is that the actual output throughput values of the possible "bottleneck" points (or smaller plants) at any given moment in time are in fixed relations in terms of one another for all possible throughput options of the system that is under scrutiny. The fixed relations are expressed as the steady state actual output throughput values of the possible "bottleneck" points. This is referred to as the governing parameters of the generic simulation modelling methodology or the FC method parameter set and it depends on the specific system description of the system that is under scrutiny. The momentary "bottleneck" is identified by dividing the maximum possible output throughput of each of the possible "bottleneck" points with the steady state actual output throughput of the possible "bottleneck" point and comparing the resulting fraction values. The possible "bottleneck" point with the smallest fraction value (*i.e.* the Benben value) is the momentary "bottleneck" and that fraction value is used to determine the actual output throughput of each of the possible "bottleneck" points. The original simulation modelling method uses a FORTRAN subroutine with its own governing parameters to identify the momentary "bottleneck" at any given moment in time. The FC method successfully resolves one of the major problem areas of the generic methodology with a solution that is much simpler and smaller than the FORTRAN subroutine of the original method. The solution also has a positive impact on all the design criteria of the generic methodology.

* * * * *

2.5 DETERMINATION OF THE GOVERNING PARAMETERS

Sections 2.2 and 2.4 indicate that it is necessary to determine the governing parameters of the system that is under scrutiny. The governing parameters are the gas-feedback-loop-fraction, steam-division-ratio, oxygen-division-ratio and the FC method parameter set. The first three are used by the FORTRAN subroutine of the original simulation modelling method and follows from the presence of feedback-loops and the fact that the output of the Steam and Oxygen plants is divided. The governing parameters of the FC method parameter set are necessary for the FC method to function. In the instance of the Synthetic Fuel plant the FC method parameter set consists of the steady state actual output throughput values of the 13 possible "bottleneck" points in the main-gas-cycle. The governing parameters depend on the specific system description of the system that is under scrutiny and are unique for every specific system description.

In essence the problem is to determine the value of the steady state actual output throughput of each of the 13 possible "bottleneck" points in the main-gas-cycle of the Synthetic Fuel plant. As previously explained the term "steady state" implies that the influence of time is removed from the system and therefore the influence of services and failures are ignored. At the steady state the Synthetic Fuel plant operates at the maximum possible throughput of the steady state momentary "bottleneck" or multiple momentary "bottleneck". It is not easy to identify the steady state momentary "bottleneck" because of the presence of feedback-loops and the fact that the output of the Steam and Oxygen plants is divided.

Crowe *et al.* (1971:14) discuss the complexities that follow from the problem of the recycling (*i.e.* the feedback-loops) of either heat or matter in chemical plants and suggest the following technique to handle feedback-loops:

> "*The output from a unit can only be calculated if its input is known, but for a process with recycle its input is only known once its output has been calculated. The classic chemical engineering approach has been to assume values for as many streams as are required to compute a unit and then proceed until the calculated values of stream variables agree with the assumed values. Although steps can be taken to accelerate the solution of a recycle problem, unaided convergence to solution is still widely used.*"

This haphazard technique can be much improved by using an iterative-loop technique that automatically converges to the correct solution. The iterative-loop technique is best explained by an exposition of the steps that are necessary to determine the governing parameters of the Synthetic Fuel plant.

The following steps are necessary to determine the governing parameters of the Synthetic Fuel plant:

a) Identify all the points of evaluation (*i.e.* the possible "bottleneck" points in the system that influence the maximum possible throughput of the system that is under scrutiny), the relevant process flow (including feedback-loops and the divided output of smaller plants) and the relevant process logic or rules of operation. Characterise all the points of evaluation with their respective number of modules and their input and output capacities. The process flow that influences the maximum possible throughput of the system that is under scrutiny is referred to as the primary process flow and it is divided into the main process flow and the auxiliary process flow. It is extremely important to incorporate the process logic or rules of operation. For example, in this instance the Water Treatment plant can be excluded from the points of evaluation because it never acts as a "bottleneck" in the main-gas-cycle (see Point f) of the rules of operation in Appendix B). The process flow of fine coal from the Coal Processing plant to the Steam plant can also be excluded from the relevant process flow because the "bottleneck" capacity of the Coal Processing plant is determined by its capacity to supply coarse coal (see Point e) of the rules of operation in Appendix B). In this instance the points of evaluation are the 13 possible "bottleneck" points in the main-gas-cycle of the Synthetic Fuel plant. They are the following: 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.

b) Identify the points of evaluation of the main process flow. The main process flow is the process flow of the coal and its derivatives in the main-gas-cycle. In this instance the points of evaluation of the main process flow are the following: Coal Processing (coarse coal), Gas Production (raw gas), Temperature Regulation (raw gas), Plant(I) (pure gas), Plant(II)-A (residue gas), Plant(II)-B (residue gas), Plant(III) (down gas), Division Process (H_2 and CH_4) and Recycling (recycled gas).

c) Identify the points of evaluation of the auxiliary process flow. The auxiliary process flow is the process flow that supports the main process flow, namely: the steam and oxygen process flow. In this instance the points of evaluation of the auxiliary process flow are the Steam plant (steam) and the Oxygen plant (oxygen).

d) Use an iterative loop to determine the actual output throughput values of the points of evaluation of the main process flow with a "push" principle that evaluates the points of evaluation in the sequence of the main process flow. Start with an actual output throughput that is less than the maximum possible output throughput of the first point of evaluation. For example, start off with an actual output throughput of 661,5 ton/h (50% of 14 multiplied by 94,5 ton/h) coarse coal for the actual output throughput of the Coal Processing plant. Move forward through the main process flow and determine all the actual output throughput values of the points of evaluation using their input to output ratios. The actual input throughput of Plant(II)-A is the sum of the actual output throughput of Plant(I), the Division Process plant and the Recycling plant. During the first iteration the actual output throughput of the Division Process and Recycling plants are obviously zero because they follow on Plant(II)-A in the sequence of the main process flow. During the second iteration the actual output throughput of the Division Process and Recycling plants are not zero anymore and they start to influence the actual input throughput of Plant(II)-A. When a number of iterations are completed, the actual output throughput of the Division Process and Recycling plants and hence also the actual input throughput of Plant(II)-A all stabilise on their correct respective actual throughput values. Stop the iterative loop when the actual output throughput values have stabilised. Verify that the actual output throughput values of the points of evaluation of the main process flow do not exceed their respective maximum possible output throughput values. If this happens, reduce the actual output throughput of the first point of evaluation and start the iterative loop again.

e) Use a straightforward calculation to determine the actual output throughput values of the points of evaluation of the auxiliary process flow with a "pull" principle that evaluates the points of evaluation in the reverse sequence of the auxiliary process flow. If the auxiliary process flow is linked, once again use the reverse sequence of the linking auxiliary process flow. In this instance the auxiliary process flow of the Steam and Oxygen plants is linked because the Steam plant supplies steam to the Oxygen plant. Using the reverse order of the linked auxiliary process flow, the oxygen will first be "pulled" from the Oxygen plant by the oxygen user plants to determine the actual output throughput of the Oxygen plant (using the input to output ratios of the relevant plants) and then the steam will be "pulled" from the Steam plant by the steam user plants to determine the actual output throughput of the Steam plant (using the input to output ratios of the relevant points of evaluation). Once again, verify that the actual output throughput values of the points of evaluation of the auxiliary process flow do not exceed their respective maximum possible output throughput values. If that happens, reduce the actual output throughput of the first point of evaluation and start the iterative loop again.

f) Determine the gas-feedback-loop-fraction by determining the ratio of pure gas (the actual output throughput of Plant(I)) to the pure gas plus the H_2 (the actual output throughput of the Division Process plant) and the recycled gas (the actual output throughput of the Recycling plant).

g) Determine the steam-division-ratio by determining the ratio of the portion of the actual output throughput of the Steam plant that is supplied to the Gas Production plant to the total actual output throughput of the Steam plant. Repeat the calculation for the portion of the actual output throughput of the Steam plant that is supplied to the Oxygen plant.

h) Determine the oxygen-division-ratio by determining the ratio of the portion of the actual output throughput of the Oxygen plant that is supplied to the Gas Production plant to the total actual output throughput of the Oxygen plant. Repeat the calculation for the portion of the actual output throughput of the Oxygen plant that is supplied to the Recycling plant.

i) Determine the FC method parameter set. The actual output throughput values of the 13 possible "bottleneck" points and therefore their fixed relations in terms of one another, are already available at this point, because the actual output throughput values of the 13 possible "bottleneck" points are in fixed relations in terms of one another for all possible throughput options of the Synthetic Fuel plant. These actual output throughput values only represent one possible throughput option of the Synthetic Fuel plant and not the steady state maximum possible throughput of the Synthetic Fuel plant. The FC method parameter set is defined by the steady state actual output throughput values of the 13 possible "bottleneck" points. The steady state actual output throughput values are determined by using an inverse variation of the FC method.

The steady state maximum possible output throughput of each of the 13 possible "bottleneck" points, as a constant, is the steady state number of available modules in the possible "bottleneck" point, as a constant, multiplied by the output capacity of a module in the possible "bottleneck" point, as a constant (see the steady state maximum possible output throughput option of Equation 2.1).

The actual output throughput of each of the 13 possible "bottleneck" points, as a constant, divided by the steady state maximum possible output throughput of the possible "bottleneck" point, as a constant, gives a fraction value of the possible "bottleneck" point, as a constant. The fraction value represents the utilisation fraction value of the possible "bottleneck" point in terms of the steady state maximum possible output throughput of the possible "bottleneck" point.

$$Fraction_{PltUtl} = (Throughput_{PltActOut}) \ / \ (Throughput_{PltSSMaxPosOut}) \quad (number) \quad (Eq.{:}2.10)$$

Where:

$Fraction_{PltUtl}$	:	The utilisation fraction value of the smaller plant, as a constant.
$Throughput_{PltActOut}$	:	The actual output throughput of the smaller plant, as a constant, in ton/h, m^3/h or nm^3/h.
$Throughput_{PltSSMaxPosOut}$	:	The steady state maximum possible output throughput of the smaller plant, as a constant, in ton/h, m^3/h or nm^3/h.

Even though the discussions in this section use the term "possible "bottleneck" point" to make provision for instances where some of the smaller plants consist of groupings of different types of modules, Equations 2.10 to 2.12 use the term "smaller plant" to maintain commonality with the nomenclature of Equations 2.1 to 2.9.

The possible "bottleneck" point with the largest utilisation fraction value is obviously the momentary "bottleneck" of that particular throughput option of the Synthetic Fuel plant. The reciprocal (*i.e.* the inverse) of the largest utilisation fraction value gives a fraction value that can be used to determine the steady state actual output throughput values of the 13 possible "bottleneck" points. This reciprocal is referred to as the parameter set determination Benben value. The parameter set determination Benben value is a constant and can only assume values that are equal to, or larger than one. (In contrast to the regular Benben value that is a function of time and can only assume values that are equal to, or smaller than one.)

$$Benben_{PSDet} = (1) / (Largest(Fraction_{PltUtl})) \quad (number) \qquad (Eq.:2.11)$$

Where:

$Benben_{PSDet}$	:	The parameter set determination Benben value is the reciprocal of the largest utilisation fraction value, as a constant.

The steady state actual output throughput of each of the 13 possible "bottleneck" points, as a constant, is the parameter set determination Benben value, as a constant, multiplied by the actual output throughput of the possible "bottleneck" point, as a constant.

$$Throughput_{PltSSActOut} = (Benben_{PSDet})(Throughput_{PltActOut}) \quad (ton,m^3,nm^3/h) \qquad (Eq.:2.12)$$

Where:

$Throughput_{PltSSActOut}$	:	The steady state actual output throughput of the smaller plant, as a constant, in ton/h, m^3/h or nm^3/h.

None of the variables in Equations 2.10, 2.11 and 2.12 is a function of time, because the governing parameters depend on the specific system description of the system that is under scrutiny and they are constants for that specific system description.

The steps that are necessary to determine the governing parameters of the Synthetic Fuel plant are graphically depicted in Figure 2.2: *Governing Parameters Determination*.

The steps of the iterative-loop technique that are described in the previous paragraph (to determine the governing parameters of the Synthetic Fuel plant) can easily be adopted to determine the governing parameters of any system of the class or type of system that is considered in this document.

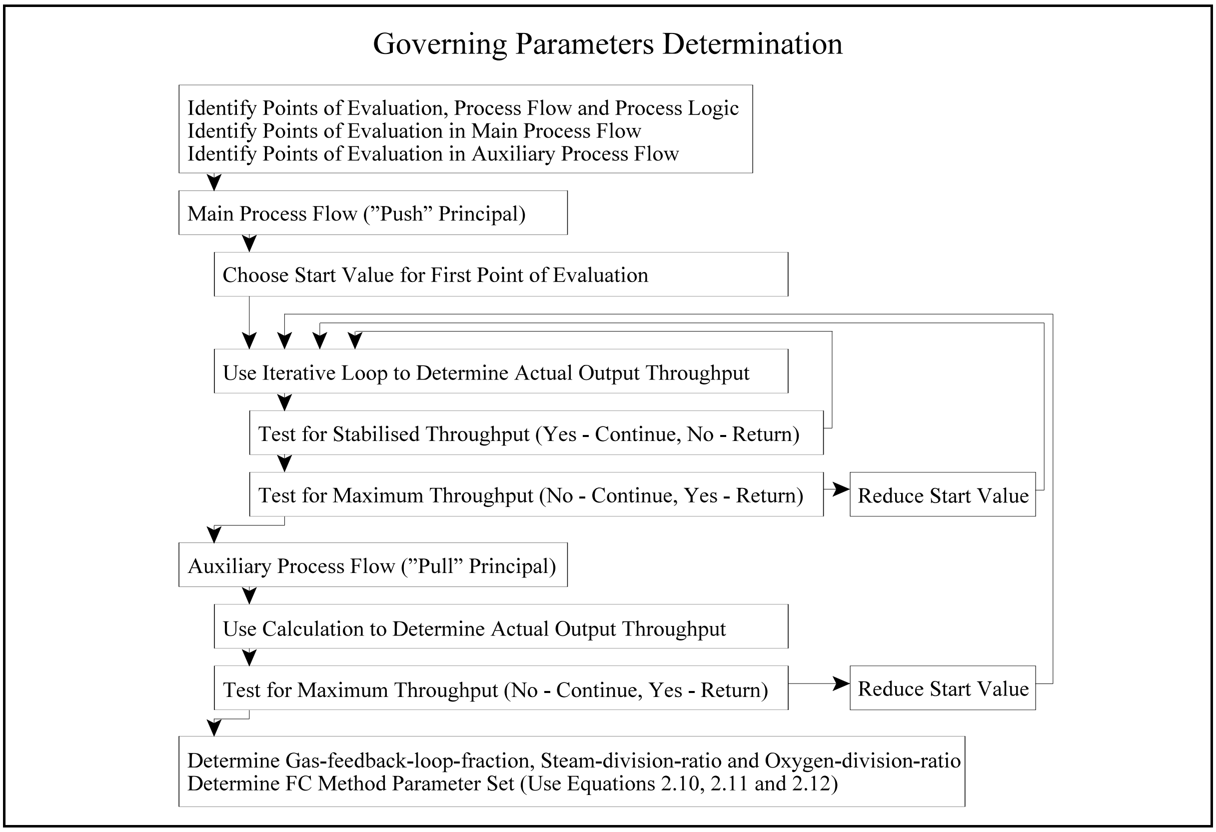

Figure 2.2: Governing Parameters Determination

A software programme that determines the governing parameters of any system of the class or type of system that is considered in this document can also be developed in a general scientific and engineering software package like FORTRAN or Visual Basic for Applications (VBA) quite easily. In this instance the governing parameters are determined with a FORTRAN software programme called PSCALC.FOR. The relevant input values are handled by an input file called PSCALC.IN. The use of an input file enhances the user-friendliness of the determination of the governing parameters and therefore it supports the user-friendliness criterion (see Point c) of the design criteria in Section 1.5) of the generic simulation modelling methodology. An example of PSCALC.IN is provided in Appendix C: *PSCALC.IN (Governing Parameters Determination Input File)*. This example contains the input values of the system description of the Synthetic Fuel plant that is detailed in Section 1.2. A scrutiny of PSCALC.IN reveals that it contains the number of modules in each of the 13 possible "bottleneck" points and the respective relevant input and output capacities of each of their modules. If the number of modules in each of the 13 possible "bottleneck" points changes, or if the input and output capacities of each of their modules change, it can easily be accommodated by the manipulation of the input file alone. However, if the process flow or process logic (*i.e.* the rules of operation) changes, then

PSCALC.FOR has to be revised and changed accordingly.

Visual Basic is a registered trademark and is usually denoted by Visual Basic®. However, for the sake of simplicity it will be written simply as Visual Basic in this document. Visual Basic is a software package from the Microsoft Corporation.

The relevant values for the governing parameters of the Synthetic Fuel plant are determined by PSCALC.FOR and written to an output file named PSCALC.OUT. The use of an output file enhances the user-friendliness of the determination of the governing parameters and therefore it supports the user-friendliness criterion (see Point c) of the design criteria in Section 1.5) of the generic simulation modelling methodology. An example of PSCALC.OUT is provided in Appendix D: *PSCALC.OUT (Governing Parameters Determination Output File)*. This example contains the output values of the system description of the Synthetic Fuel plant that is detailed in Section 1.2. A scrutiny of PSCALC.OUT reveals that the format of lines three to eighteen is identical. Each line represents one iteration of the iterative loop and gives, from left to right, the actual output throughput of Plant(I), the actual input throughput of Plant(II)-A, Plant(III), Division Process and Recycling and the actual output throughput of the Recycling plant. A scrutiny of the second values in lines three to eighteen therefore indicates that the actual input throughput of Plant(II)-A stabilises on a value of 1144532 nm^3/h (for a start value of 661,5 ton/h - 50% of 14 multiplied by 94,5 ton/h - coarse coal for the actual output throughput of the Coal Processing plant). In this instance 16 iterations are necessary for the actual throughput values to stabilise. The governing parameters of the Synthetic Fuel plant, for the system description that is detailed in Section 1.2, follow in the rest of PSCALC.OUT.

The governing parameters are summarised in Table 2.1: *Governing Parameters of the Synthetic Fuel Plant*. The values of the gas-feedback-loop-fraction, steam-division-ratio and oxygen-division-ratio are given to six decimal digits which might seem excessive, but it should be remembered that the actual output throughput values of some of the smaller plants are in the order of millions and when a value of that size is multiplied by a parameter set value, it is prudent to provide the parameter set value to a few decimal digits in order to ensure high accuracy. The FC method parameter set values of Coal Processing and Steam are given to three decimal digits because these values are expressed in tons per hour, while the parameter set values of the rest of the 13 possible “bottleneck” points are given to one decimal digit because these values are expressed as normalised cubic metres per hour.

Table 2.1: Governing Parameters of the Synthetic Fuel Plant

Governing Parameter	Value
Gas-feedback-loop-fraction	
Forward (Plant(I) to Plant(II)-A)	1,576576
Backward (Plant(II)-A to Plant(I))	0,634286
Steam-division-ratio	
Gas Production	0,537612
Oxygen	0,462388
Oxygen-division-ratio	
Gas Production	0,741043
Recycling	0,258957
FC Method Parameter Set	
Coal Processing	931,253 ton/h
Steam	1762,830 ton/h
Gas Production	1460000,0 nm^3/h
Temperature Regulation	1460000,0 nm^3/h
Oxygen-A	1569088,9 nm^3/h
Oxygen-B	273062,2 nm^3/h
Oxygen-C	273062,2 nm^3/h
Plant(I)	1022000,0 nm^3/h
Plant(II)-A	515603,6 nm^3/h
Plant(II)-B	515603,6 nm^3/h
Plant(III)	444708,1 nm^3/h
Division Process	180461,2 nm^3/h
Recycling	408800,0 nm^3/h

Section 2.4 indicates that the FC method does not use the gas-feedback-loop-fraction, steam-division-ratio and oxygen-division-ratio directly, but that their influence on the operation of the Synthetic Fuel plant is incorporated into the parameter set. This is illustrated by observing the parameter set values (steady state actual output throughput) of Plant(I) and Plant(II)-A. The parameter set value (steady state actual output throughput) of Plant(I) is 1022000,0 nm^3/h and therefore the parameter set value (steady state actual output throughput) of Plant(II)-A should be 515603,4 nm^3/h (1022000,0 nm^3/h multiplied by the forward gas-feedback-loop-fraction - 1,576576 - multiplied by the output to input ratio of Plant(II)-A - 69440 nm^3/h divided by 217000

nm^3/h). The calculated steady state actual output throughput of Plant(II)-A of 515603,4 nm^3/h is sufficiently close to the parameter set value (steady state actual output throughput) of 515603,6 nm^3/h and the small difference can be attributed to the fact that the forward gas-feedback-loop-fraction is only given to six decimal digits, but the parameter set values are determined by FORTRAN with Double Precision accuracy which is 15 decimal digits.

In this example of PSCALC.OUT the actual output throughput of Plant(II)-A is 366250,2 nm^3/h (1144532 nm^3/h multiplied by the output to input ratio of Plant(II)-A - 69440 nm^3/h divided by 217000 nm^3/h) for the stabilised actual input throughput of 1144532 nm^3/h (see the second value of the 16th and last iteration in PSCALC.OUT - Appendix D). This actual output throughput only represents one possible throughput option (for a chosen start value of the actual output throughput of the Coal Processing plant) of the Synthetic Fuel plant and not the steady state maximum possible throughput of the Synthetic Fuel plant. The actual output throughput values of the FC method parameter set represent the steady state actual output throughput values of the 13 possible "bottleneck" points, which is the steady state maximum possible throughput of the Synthetic Fuel plant. The steady state actual output throughput of Plant(II)-A is 515603,6 nm^3/h.

Summary

A computerised iterative-loop technique that determines the governing parameters of the Synthetic Fuel plant is presented in this section. The governing parameters are the gas-feedback-loop-fraction, steam-division-ratio, oxygen-division-ratio and the FC method parameter set. They are not easy to determine because of the presence of feedback-loops and the fact that the output of the Steam and Oxygen plants is divided. A FORTRAN software programme called PSCALC.FOR determines the governing parameters of the Synthetic Fuel plant. The input file to the programme can easily accommodate changes to the number of modules in the 13 possible "bottleneck" points and their input and output capacities, but changes to the process flow or process logic (*i.e.* the rules of operation) will necessitate changes to the programme itself. The FC method parameter set values that are presented in Table 2.1 represent the parameter set of the Synthetic Fuel plant for the system description that is provided in Section 1.2.

* * * * *

2.6 IDENTIFICATION OF THE "BOTTLENECKS"

Section 1.1 indicates that a simulation model can be used to identify problem areas or "bottlenecks" in a system and Section 1.4 indicates that the identification of the "bottlenecks" in the Synthetic Fuel plant is one of the objectives of the original simulation model that is detailed in the *Magister* dissertation (Albertyn, 1995:3,15). The identification of the "bottleneck" smaller plants should not be confused with the identification of the momentary "bottleneck" (see Section 2.2). The identification of the "bottleneck" smaller plants are necessary to determine which of the smaller plants are "bottlenecks" over a period of time, typically a year or more, while the identification of the momentary "bottleneck" is necessary to determine the maximum possible throughput of the Synthetic Fuel plant at a specific moment in time.

The importance of the throughput as the definitive measurement of plant performance is discussed in Section 2.2. In order to devise an effective strategy to increase the throughput of a plant, it is of vital importance to accurately identify the "bottleneck" smaller plants in the plant. Goldratt and Cox (1992:294) indicate that the principal purpose of the "bottleneck" identification and elimination process is to increase the throughput of the plant.

> "*The entire bottleneck concept is not geared to decrease operating expense, it's focussed* [*sic*] *on increasing throughput.*"

Therefore it seems prudent to incorporate techniques into the generic simulation modelling methodology that accurately identify the "bottleneck" smaller plants. The original simulation modelling method uses the throughput utilisation values of the smaller plants to identify the "bottlenecks" (Albertyn, 1995:29-30). The throughput utilisation value of each of the smaller plants over a chosen period of time, as a percentage, is the mean actual output throughput of the smaller plant over the chosen period of time, as a constant, divided by the mean maximum possible output throughput of the smaller plant over the chosen period of time, as a constant, multiplied by 100.

$$Utilisation_{PltThr} = ((Throughput_{PltMnActOut}) / (Throughput_{PltMnMaxPosOut}))(100) \quad (\%) \quad (Eq.:2.13)$$

Where:

$Utilisation_{PltThr}$: The throughput utilisation value of the smaller plant over the chosen period of time, as a percentage.

$Throughput_{PltMnActOut}$: The mean actual output throughput of the smaller plant over the chosen period of time, as a constant, in ton/h, m^3/h or nm^3/h.

$Throughput_{PltMnMaxPosOut}$: The mean maximum possible output throughput of the smaller plant over the chosen period of time, as a constant, in ton/h, m^3/h or nm^3/h.

The mean maximum possible output throughput of each of the smaller plants over the chosen period of time, as a constant, is the mean number of available modules in the smaller plant over the chosen period of time, as a constant, multiplied by the output capacity of a module in the smaller plant, as a constant. (It is a logical derivative of Equation 2.1.)

$$Throughput_{PltMnMaxPosOut} = (n_{PltModMnAvl})(Capacity_{PltModOut}) \quad (ton,m3,nm3/h) \qquad (Eq.:2.14)$$

Where:

$n_{PltModMnAvl}$: The mean number of available modules in the smaller plant over the chosen period of time, as a constant.

$Capacity_{PltModOut}$: The output capacity of a module in the smaller plant, as a constant, in ton/h, m^3/h or nm^3/h.

Equation 2.13 determines the throughput utilisation value of each of the smaller plants over the chosen period of time in terms of the mean maximum possible output throughput of the smaller plant and not in terms of the steady state maximum possible output throughput of the smaller plant. The mean maximum possible output throughput of each of the smaller plants incorporates the influence of the services and failures and therefore it is a more useful measurement to use than the steady state maximum possible output throughput of each of the smaller plants that does not take the influence of the services and failures into account.

The throughput utilisation value of each of the smaller plants over a period of time gives an indication of how hard the smaller plant worked over the period of time. A high throughput utilisation value indicates that the smaller plant had very little reserve capacity over the period of time and therefore it was highly utilised over the period of time, while a low throughput utilisation value indicates that the smaller plant had substantial reserve capacity over the period of time and therefore it was not highly utilised over the period of time. Therefore a high throughput utilisation value translates into a high importance as a “bottleneck” and a low throughput utilisation value translates into a low importance as a “bottleneck”.

The generic simulation modelling introduces the following two additional "bottleneck" identification techniques:

a) The time that each of the smaller plants is the "bottleneck", as a percentage.

b) The possible production that is lost due to each of the smaller plants, as a percentage.

The time that each of the smaller plants is the "bottleneck" over a chosen period of time, as a percentage, is the period of time that the smaller plant is the "bottleneck" over the chosen period of time, as a constant, divided by the chosen period of time, as a constant, multiplied by 100.

$$\text{“}Bottleneck\text{”}_{PltTim} = ((Time_{PltBtt}) / (Time_{Tot}))(100) \quad (\%) \qquad (Eq.:2.15)$$

Where:

$\text{“}Bottleneck\text{”}_{PltTim}$	:	The time that the smaller plant is the "bottleneck" over the chosen period of time, as a percentage.
$Time_{PltBtt}$	:	The period of time that the smaller plant is the "bottleneck" over the chosen period of time, as a constant, in hours.
$Time_{Tot}$	:	The chosen period of time, as a constant, in hours.

The production that is lost due to each of the smaller plants over a chosen period of time, as a percentage, is the production that is lost due to the smaller plant over the chosen period of time, as a percentage of the steady state maximum possible production over the chosen period of time, divided by the total production that is lost over the chosen period of time, as a percentage of the steady state maximum possible production over the chosen period of time, multiplied by 100.

$$\text{“}Bottleneck\text{”}_{PltPrdLst} = ((Production_{PltLst}) / (Production_{SFPltLst}))(100) \quad (\%) \qquad (\text{Eq.:2.16})$$

Where:

$\text{“}Bottleneck\text{”}_{PltPrdLst}$	:	The production that is lost due to the smaller plant over the chosen period of time, as a percentage.
$Production_{PltLst}$	:	The production that is lost due to the smaller plant over the chosen period of time, as a percentage of the steady state maximum possible production over the chosen period of time.
$Production_{SFPltLst}$	:	The total production that is lost over the chosen period of time, as a percentage of the steady state maximum possible production over the chosen period of time.

Even though Equations 2.13 to 2.16 use the term "smaller plant", they are equally applicable when the term "smaller plant" is replaced with the term "possible "bottleneck" point" to make provision for instances where some of the smaller plants consist of groupings of different types of modules.

The effect of the multiple momentary "bottleneck" occurrences is taken into account when the "bottleneck" smaller plants in the Synthetic Fuel plant are identified. When a multiple momentary "bottleneck" occurs, the time that the multiple momentary "bottleneck" is the "bottleneck", is divided equally among the possible "bottleneck" points that make up the multiple momentary "bottleneck" and the same applies for the production that is lost due to the multiple momentary "bottleneck". This ensures that the two "bottleneck" identification techniques that are included in the generic simulation modelling methodology, give a true reflection of the "bottleneck" status of each of the smaller plants.

The two "bottleneck" identification techniques do not form part of the FC method, but the concepts of the FC method lend themselves to the easy implementation of the two techniques. For example, when the production that is lost due to each of the smaller plants is determined, the difference between the actual output throughput and the steady state actual output throughput (FC method parameter set value) of the momentary "bottleneck" point, is used as the point of departure for the calculation.

The second rule of operation in Appendix B indicates that only the smaller plants that form part of the main-gas-cycle can act as "bottlenecks" that influence the rate of production or throughput of the Synthetic Fuel plant. It is obvious that the two "bottleneck" identification techniques that are detailed in the previous paragraphs are aimed at identifying the "bottlenecks" in the main-gas-cycle of the Synthetic Fuel plant. The two techniques can be used to prioritise the 13 possible "bottleneck" points. The 13 possible "bottleneck" points are referred to as the primary "bottlenecks" and they are the following: 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. They are referred to as primary "bottlenecks" because the throughput of the Synthetic Fuel plant at any given moment in time is adjusted to coincide with the maximum possible throughput of the momentary "bottleneck" at that specific moment in time.

The fourth rule of operation in Appendix B indicates 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 in the Synthetic Fuel plant, then the portions of the throughput that cannot be processed are flared. It is obvious that these smaller plants act as “bottlenecks” if it is necessary to flare portions of their throughput at any of their respective positions in the Synthetic Fuel plant. These smaller plants are referred to as secondary “bottlenecks” because they do not influence the main-gas-cycle but flare the portions of the throughput that cannot be accommodated at their respective positions. The portions of the throughput that are flared at their respective positions are determined by the generic simulation modelling methodology to ensure that the secondary “bottlenecks“can be identified, prioritised and managed accordingly.

Both the primary and secondary “bottlenecks” are undesirable from the perspectives of increased efficiency and the realisation of profit (see Section 1.3). Therefore they need to be managed with circumspection. Furthermore, the secondary “bottlenecks” are also undesirable as seen from the environmental perspective.

Summary

This section indicates that a simulation model can be used to identify the problem areas or “bottlenecks” in a system. The original simulation modelling method uses the throughput utilisation values of the 13 possible “bottleneck” points to identify the “bottlenecks” in the main-gas-cycle of the Synthetic Fuel plant. They are referred to as the primary “bottlenecks”. The generic simulation modelling methodology introduces two additional techniques to identify the primary “bottlenecks”. The first technique determines the time that each of the 13 possible “bottleneck” points is the ”bottleneck” and the second technique determines the production that is lost due to each of the possible “bottleneck” points. If portions of the throughput are flared at Plant(IV), Plant(V) and Sub(I) to Sub(VI) it is indicative of the existence of a secondary “bottleneck” and they also have to be identified and managed.

* * * * *

2.7 STRUCTURE OF THE GENERIC METHODOLOGY

From the discussions in the previous sections of this chapter, the structure of the generic simulation modelling methodology can now be conceptualised. This section indicates how the different methods and techniques that are developed in the previous sections of this chapter are integrated to render the structure of the generic methodology. A simulation model mimics the behaviour of a system and in this instance the behaviour of a stochastic continuous system is mimicked. It is of cardinal importance for any simulation modelling methodology to be based on the characteristics of the class or type of system that is under scrutiny.

In Section 2.1 the characteristics of the Synthetic Fuel plant are identified and in the following sections methods and techniques are identified and developed to effectively accommodate the characteristics in a simulation model. Table 2.2: *System Characteristics and Appropriate Methods and Techniques* gives an overview of the characteristics of the Synthetic Fuel plant and the corresponding “toolbox” of appropriate methods and techniques that are detailed in this chapter to solve the problems that are posed by the characteristics.

The two “bottleneck” identification techniques are shown in Table 2.2 as in relation to the complex interrelationship characteristic because even though the two techniques do not form part of the FC method, they both use the FC method concepts as the point of departure for their respective calculations.

The key objective of the generic simulation modelling methodology is to provide a simulation modelling methodology that can be used to construct simulation models of stochastic continuous systems (*i.e.* systems that are similar to the Synthetic Fuel plant) effectively. The first rule of operation in Appendix B states that the Synthetic Fuel plant always strives to maintain the maximum possible rate of production or throughput and Section 2.2 indicates that the throughput of a plant is considered to be the definitive measurement of plant performance. The two statements in the previous sentence clearly highlight the pivotal role that the determination of the maximum possible throughput, as a function of time, plays in a simulation model of the Synthetic Fuel plant. Equation 2.4 (repeated here for the sake of the continuity of the argument) indicates that the maximum possible throughput of the Synthetic Fuel plant is a function of the maximum possible throughput of each of the smaller plants.

Table 2.2: System Characteristics and Appropriate Methods and Techniques

System Characteristic	Method or Technique	Purpose
Continuous Process	Variables Technique (Section 2.2)	Uses variables to represent process flow, like the output throughput values of the smaller plants, as real numbers.
	Fixed Time Interval Technique (Section 2.2)	Uses a fixed time interval to advance the simulation model in time.
Discrete Events (Services and Failures)	ERM Method (Section 2.3)	Determines the state of the modules in the system that is under scrutiny at any given moment in time.
Complex Interrelationships	FC Method (Section 2.4)	Identifies the momentary "bottleneck" in a complex system at any given moment in time.
	Iterative-loop Technique (Section 2.5)	Determines the governing parameters (gas-feedback-loop-fraction, steam-division-ratio, oxygen-division-ratio and FC method parameter set).
	Time "Bottleneck" Identification Technique (Section 2.6)	Identifies primary "bottleneck" smaller plants based on the time that the smaller plant is the "bottleneck".
	Production Lost "Bottleneck" Identification Technique (Section 2.6)	Identifies primary "bottleneck" smaller plants based on the production that is lost due to the smaller plant.

$$Throughput_{SFPltMaxPos}(t) = f(Throughput_{PltMaxPos}(t) \text{ for No.1} \ldots n_{Plt}) \quad (ton, m^3, nm^3/h) \ (Eq.:2.4rep)$$

Where:

$Throughput_{SFPltMaxPos}(t)$: The maximum possible throughput of the Synthetic Fuel plant, as a function of time, in ton/h, m^3/h or nm^3/h.

$Throughput_{PltMaxPos}(t)$: The maximum possible throughput of the smaller plant, as a function of time, in ton/h, m^3/h or nm^3/h.

n_{Plt} : The number of smaller plants, as a constant.

It is difficult to determine the maximum possible throughput of the Synthetic Fuel plant, as a function of time, because of the presence of feedback-loops, the division of the output of the Steam and Oxygen plants and the fact that the number of available modules in each of the smaller plants is a function of time.

A scrutiny of Table 2.2 indicates that the "toolbox" of methods and techniques provides solutions to all the problems that are posed in the previous paragraph. The ERM method determines the number of available modules in each of the smaller plants at any given moment in time and then the FC method identifies the momentary “bottleneck” and determines the maximum possible throughput of the Synthetic Fuel plant at that specific moment in time. The FC method uses a parameter set that is determined with the iterative-loop technique. The FC method parameter set is unique for every specific system description and incorporates the influence of the gas-feedback-loop-fraction, steam-division-ratio and oxygen-division-ratio on the operation of the Synthetic Fuel plant.

The maximum possible throughput of the Synthetic Fuel plant at any given moment in time is only influenced by the 13 possible “bottleneck” points in the main-gas-cycle and therefore only the 13 possible “bottleneck” points are included in the FC method. This implies that the actual output throughput values of only the 13 possible “bottleneck” points at that specific moment in time are provided by the FC method. The 13 possible “bottleneck” points belong to the 10 smaller plants in the main-gas-cycle and these plants are referred to as the “heart” of the Synthetic Fuel plant. The smaller plants that do not form part of the main-gas-cycle are referred to as the peripheral plants.

The maximum possible throughput of the Synthetic Fuel plant at any given moment in time is defined by a “throughput vector” that consists of the actual input throughput of the Synthetic Fuel plant and the actual output throughput of each of the smaller plants (see the convention that is detailed in Section 2.2). The FC method only renders the actual output throughput values of the 13 possible “bottleneck” points at that specific moment in time and therefore the other outstanding throughput values need to be determined. The outstanding throughput values (*i.e.* the actual input throughput of the Synthetic Fuel plant and the actual output throughput of all the peripheral plants) of the “throughput vector” at that specific moment in time are easy to determine because there are no feedback-loops or the division of output to complicate the calculations. There is one complication though, the modules of some of the peripheral plants are subject to services and failures. Fortunately the ERM method also provides the number of available modules at any given moment in time in each of the peripheral plants that are subject to services and failures.

The operation of the Synthetic Fuel plant can be likened to a huge transfer function that turns coal and water into chemical products. If the main-gas-cycle is viewed as the primary transfer function, then the ERM method determines the status of the time-dependent elements (*i.e.* the modules) of the transfer function and the FC method identifies the momentary “bottleneck” in the primary transfer function and hence determines the maximum possible throughput of the primary transfer function. The FC method actually optimises the primary transfer function in terms of possible throughput. The FC method parameter set values represent the governing parameters of the elements of the primary transfer function that determine the maximum possible throughput of the primary transfer function. If the configuration of the transfer function changes, it means that the governing parameters must change to reflect these changes. The peripheral plants can be viewed as constituting the secondary transfer functions of the Synthetic Fuel plant that turn the throughput from the main-gas-loop into the final products of the Synthetic Fuel plant. The secondary transfer functions are straightforward, because there are no feedback-loops or the division of output in the secondary transfer functions and the ERM method determines the status of the time-dependent elements.

To summarise, Equation 2.4 represents the 13 possible "bottleneck"points in the "heart"of the Synthetic Fuel plant and it is solved over time for the maximum possible throughput of the Synthetic Fuel plant with the help of the ERM method (which determines the state of the time-dependent elements) and the FC method (which identifies the momentary "bottleneck" and determines the maximum possible throughput). The actual input throughput of the Synthetic Fuel plant and the actual output throughput of the peripheral plants are determined over time with straightforward calculations and the help of the ERM method.

If the maximum possible throughput at any given moment in time is known, the corresponding number of modules that is switched on or off in each of the smaller plants at that specific moment in time, can easily be determined with Equations 2.5 and 2.6. The input that is needed to identify the "bottleneck" smaller plants can also be determined at that specific moment in time and after the completion of the simulation run it is used to identify the "bottleneck" smaller plants with Equations 2.15 and 2.16.

The generic simulation modelling methodology, as presented in this instance, assumes that the system that is under scrutiny strives to maintain the maximum possible rate of production or throughput, but the generic methodology can easily be adapted to represent a system that strives to maintain a given constant rate of production or throughput. An example of such a system is a power plant that supplies electricity into a national network or grid. In such an instance the demand for electricity from the power plant is relatively constant (depending on seasonal variation) and the maximum possible rate of production is reserved for emergencies only.

A scrutiny of the "toolbox " of seven methods and techniques that is presented in Table 2.2 indicates that they are applicable at different stages during the completion of a simulation run.

The majority of the methods and techniques are used continuously by the simulation model during the simulation run. The notable exception to this rule is the iterative-loop technique that determines the governing parameters of the system that is under scrutiny before the start of the simulation run. This implies that the iterative-loop technique does not need to be an integral part of the simulation model. Therefore the generic simulation modelling methodology consists of two separate parts, namely: an iterative-loop technique part and a simulation model part. The iterative-loop technique part accommodates the specific system description of the system that is under scrutiny and the simulation model part contains the six methods and techniques that accommodate the time dependent behaviour of the system that is under scrutiny. This concept is graphically depicted in Figure 2.3: *Generic Simulation Modelling Methodology Parts, Methods and Techniques*.

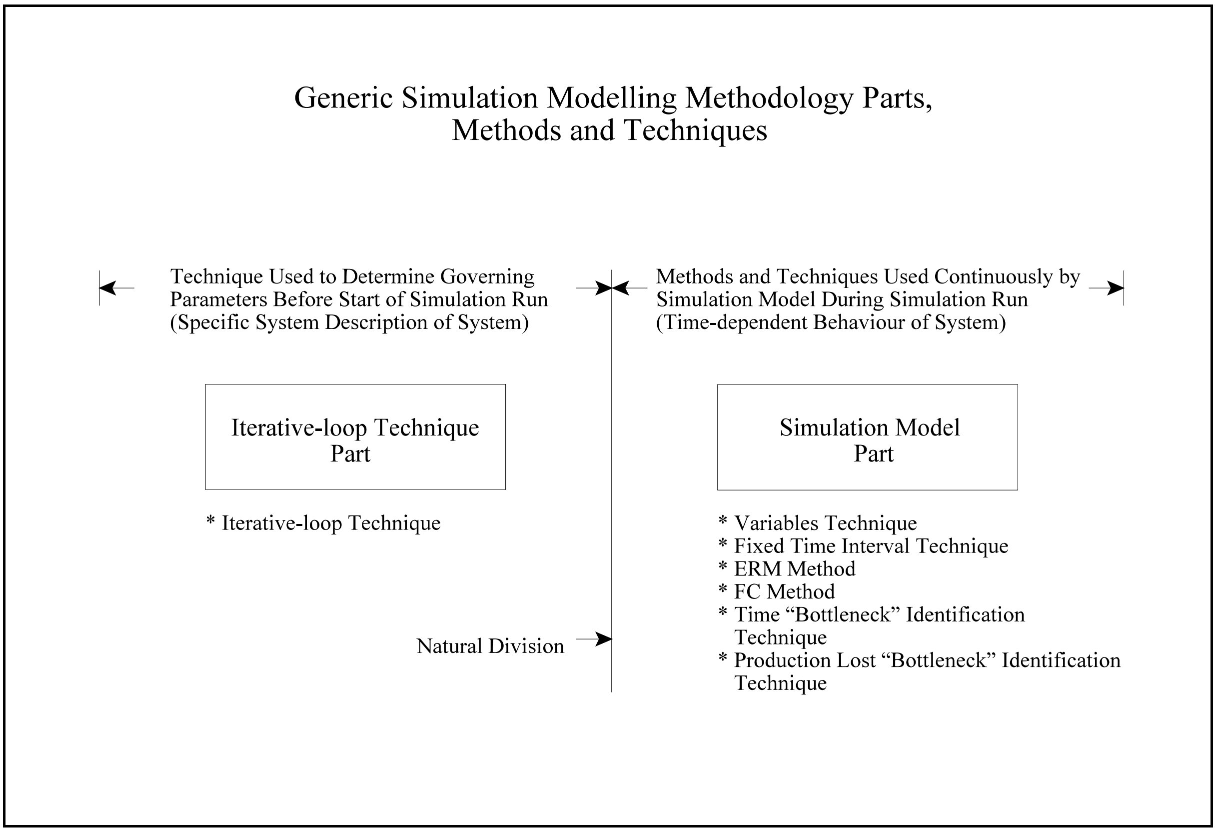

Figure 2.3: Generic Simulation Modelling Methodology Parts, Methods and Techniques

The advantages of this natural division of the generic simulation modelling methodology are the following:

a) It supports the compact simulation model size design criterion of the generic simulation modelling methodology (see Point e) of the design criteria in Section 1.5), because a general scientific and engineering software package like FORTRAN can be used for the cumbersome but straightforward calculations that are necessary for the iterative-loop technique to determine the governing parameters (see Section 2.5). If a general scientific and engineering software package like FORTRAN is used, the resulting programme that consists of lines of programming code is appreciably smaller than if basic simulation software package building blocks are used to achieve the same outcome in a simulation software package.

b) It supports the short simulation runtime criterion of the generic simulation modelling methodology (see Point d) of the design criteria in Section 1.5), because a general scientific and engineering software package like FORTRAN is ideally suited to the "number crunching" that is required when the iterative-loop technique determines the governing parameters. Simulation software packages are not partial to "number crunching" and a time penalty is incurred when "number crunching" is performed by a simulation software package.

Section 2.2 indicates that the continuous processes of the Synthetic Fuel plant can be presented by variables in a simulation model and Section 2.3 indicates that the behaviour of the modules can be represented by the ERM method in a simulation model. The substantial differences between the representation of the continuous processes and the representation of the behaviour of the modules lead to a natural division of the simulation model into two parts. One part deals with the continuous processes while the other deals with the behaviour of the modules. The part of the simulation model that deals with the continuous processes is referred to as the "virtual" part of the simulation model because the actual processes are represented by variables and logical equations (*i.e.* the process flow and process logic or rules of operation are represented by variables and logical equations). The "virtual" part of the simulation model also accommodates all the other concepts that are necessary for the simulation model to function. The part that deals with the behaviour of the modules is referred to as the "real" part of the simulation model because the actual modules are represented by standard simulation software package building blocks. This concept is already introduced in Section 2.3 but it is repeated here for the sake of the continuity of the argument.

The concepts that are accommodated by the "virtual" part of the simulation model are the following:

a) The variables technique that uses variables to represent process flow.

b) The fixed time interval technique that uses a fixed time interval to advance the simulation model in time.

c) The control of the ERM method that determines the number of available modules in each of the smaller plants at any given moment in time.

d) The FC method that identifies the momentary "bottleneck" in a complex system at any given moment in time.

e) The determination of the maximum possible throughput (*i.e.* the "throughput vector") at any given moment in time.

f) The determination of the number of modules that is switched on or off at any given moment in time.

g) The determination of the input that is needed to identify the primary and secondary "bottleneck" smaller plants at any given moment in time.

h) The determination of the variables that keep record of the functioning of the simulation model at any given moment in time (*i.e.* the number of evaluations completed, the number of services completed, the number of failures repaired, *etc.*).

i) The determination of all the mean values of the relevant variables at the end of the simulation run (*i.e.* the mean values of the "throughput vector", the mean values of the number of available modules, the mean values of the number of modules that is switched on or off, *etc.*).

j) The identification of the primary and secondary "bottleneck" smaller plants at the end of the simulation run. (The primary "bottleneck" smaller plants are identified with the time and production lost "bottleneck" identification techniques.)

Section 2.3 indicates that four of the different types of smaller plants can be represented in the "real" part of the simulation model by four different high-level building blocks. The four different high-level building blocks are the following: a smaller plant with a multiple service cycle and failures of the modules, a smaller plant with a service cycle and failures of the modules, a smaller plant with a service cycle of the modules and a smaller plant with failures of the modules. The concepts of the "virtual" part of the simulation model that are discussed in the previous paragraph can be grouped together in one high-level building block that represents the "virtual" part of the simulation model. This high-level building block is referred to as the logic engine.

To summarise, the simulation model consists of a "virtual" part that deals with the continuous processes and all the other concepts that are necessary for the simulation model to function and a "real" part that deals with the behaviour of the modules. The "virtual" part of the simulation model is represented by the logic engine high-level building block and the "real" part is represented by the four different high-level building blocks of the ERM method.

From the discussions in the previous paragraphs it is clear that a simulation model of the Synthetic Fuel plant, or any other stochastic continuous system, can easily be constructed with the five high-level building blocks. The basic structure of the simulation model is graphically depicted in Figure 2.4: *Simulation Model Parts and Building Blocks*.

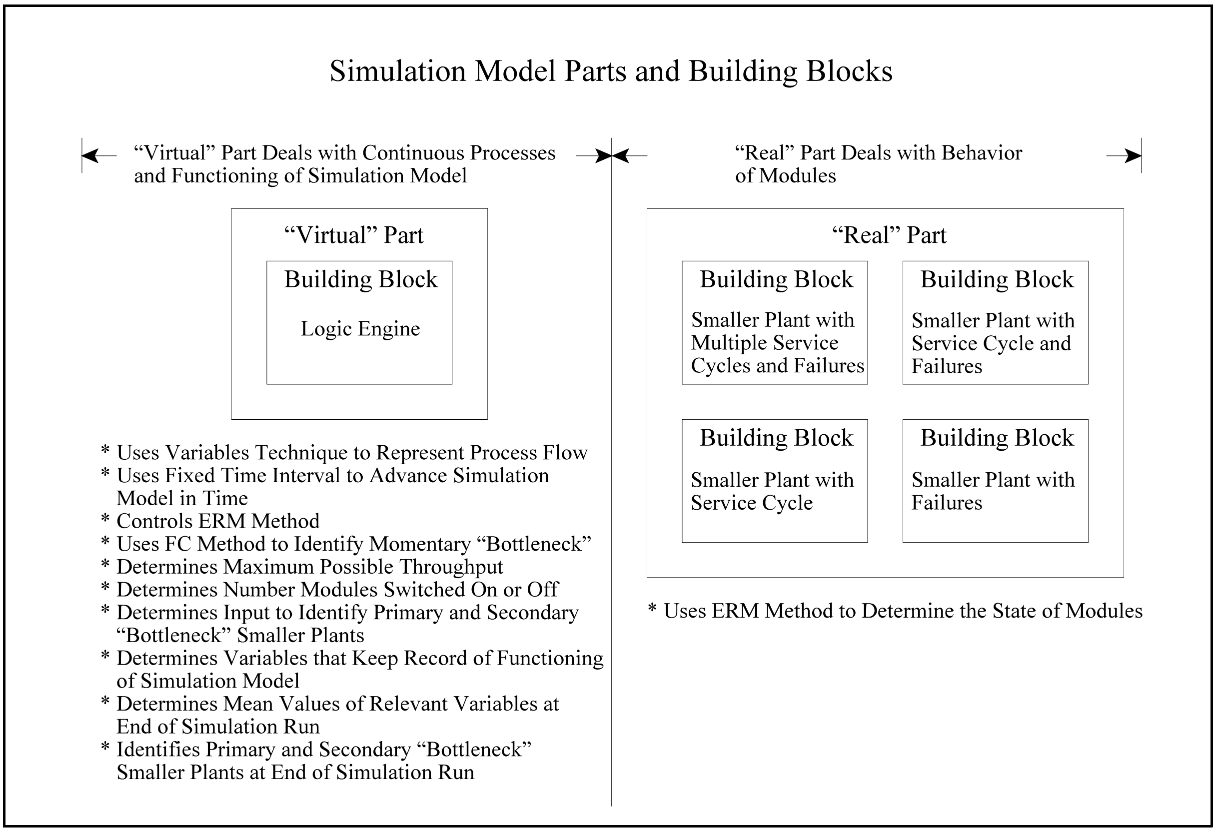

Figure 2.4: Simulation Model Parts and Building Blocks

Section 2.3 indicates that the building blocks that represent the smaller plants in the "real" part of the simulation model are populated with the corresponding correct number of entities that represent the modules and appropriate values are also assigned to the attributes of each of the entities (*i.e.* the modules) before the start of the simulation run. This process can either be handled centrally by the logic engine or every one of the building blocks that represent the smaller plants can populate itself, depending on the simulation software package that is used and the personal preference of the modeller. For example, in the Arena simulation model that is developed in Chapter 3 the smaller plants are populated with entities by the logic engine (*i.e.* centralised populating), but in the Simul8 simulation model that is developed in Chapter 3 each of the building blocks that represents the smaller plants is populated by itself (*i.e.* decentralised populating).

During the simulation run the logic engine controls the functioning of the simulation model and uses the fixed time interval technique to advance the simulation model in time. Every time interval an evaluation takes place and the logic engine completes the necessary tasks of the concepts of the "virtual" part of the simulation model that are discussed above in Points c) to h). After the completion of the simulation run the logic engine completes the necessary tasks of the concepts of the "virtual" part of the simulation model that are discussed above in Points i) and j).

One of the major benefits of using the variables technique to represent the process flow is that the simulation run can start immediately after the building blocks of the smaller plants have been populated with modules, no warm-up period is necessary to wait for the simulation model to "fill up" with entities before the actual simulation run can start. A simulation model of a simulation modelling method that uses entities to represent the "commodities" that move or flow through the system, is usually empty when a simulation run starts and therefore a warm-up period is necessary for the simulation model to "fill up" with entities. The exceptions, of course, are when the actual start-up of a system is modelled (*i.e.* the commissioning of a new plant), or if the system starts every cycle empty (*i.e.* the post office opens at nine o'clock in the morning). Usually only the actual part of the simulation run is of importance and Taha (1987:714) indicates that the observations gathered during the warm-up period of the simulation run have to be discarded in such an instance.

> "*We have seen ... that early output of the simulation experiment is* ***unstable*** *(transient state) and that* ***stability*** *(steady state) is usually reached after the simulation run becomes "sufficiently" long. As a result, care must be taken that observations are not gathered during the early stages of the simulation run, because the information obtained is subject to large variation and hence may not be representative of the true behaviour of the system.*" [Bold typeface added for emphasis]

Taha uses the terms "transient" and "steady state" in a slightly different context than the way that the two terms are used in this document. Taha uses the two terms on the "macro" level (*i.e.* the level of the behaviour of the simulation model) to distinguish between the "fill up" period of the simulation model and the actual simulation run. In this document the term "transient" is used on the "micro" level (*i.e.* the level of the behaviour of the system that is modelled) to indicate the behaviour of the system if it changes form one state to another during the simulation run and the term "steady state" is also used on the "micro" level to indicate that the influence of time has been removed from the system that is modelled. In this document the terms "unstable" and "stable" are preferred to distinguish between the warm-up period and the actual simulation run.

Pegden *et al.* (1995:180) indicate that, while there are some "rules" to determine the length of the warm-up period, they are subject to constraints and therefore restricted in their application.

> "*..., but experience suggests that a rule's performance depends largely on the nature of the simulation response. Consequently, these rules are generally* ***not used*** *in simulation applications.*" [Bold typeface added for emphasis]

Pegden *et al.* (1995:180) also propose a practical method to identify the truncation point (*i.e.* to determine the length of the warm-up period).

> "*The simplest, most practical, and probably best method for selecting the truncation point is visual determination, i.e., selecting the point from a plot of the simulation response over time.*"

The *Simul8®: Manual and Simulation Guide* (1999:35-38) suggest a short simulation run, visual inspection of the results (*i.e.* the data and the graphs) and a judgement call to determine the warm-up period. Harrell and Tumay (1999:129-130) and Kelton *et al.* (1998:219-223) also advocate the use of this technique. Two of the three aforementioned references also suggest adding a 20% to 30% safety factor to the observed warm-up period. It seems time-consuming and also risky from an accuracy perspective to use this technique.

The advantages of the fact that the variables technique needs no warm-up period are the following:

a) It supports the short simulation runtime criterion of the generic simulation modelling methodology (see Point d) of the design criteria in Section 1.5) because no computer time is wasted on a warm-up period.
b) It supports the accurate modelling ability criterion of the generic simulation modelling methodology (see Point g) of the design criteria in Section 1.5) because the risk of including data from the "unstable" warm-up period into the results is negated.

Another small improvement of the generic simulation modelling methodology over the original simulation modelling method is that the generic methodology immediately starts the simulation run, whereas the original method uses the first time interval to set up the simulation model and only then starts the simulation run. This does not have a major impact because the part of the behaviour of the system that is lost over the first time interval by the original method constitutes only a very small fraction of the total behaviour of the system over the period of time that is usually modelled in a simulation run. However, it is still important to work as accurately as possible and therefore the generic methodology eliminates this small aberration that exists in the original method. This small change obviously also supports the accurate modelling ability criterion of the generic methodology (see Point g) of the design criteria in Section 1.5).

Summary

This section conceptualises the structure of the generic simulation modelling methodology. The seven methods and techniques that are developed in the previous sections to solve the problems that are posed by the characteristics of stochastic continuous systems are integrated to form the generic methodology. There is a natural division of the generic methodology into two parts, namely: an iterative-loop technique part that determines the governing parameters before the start of a simulation run and a simulation model part that uses the other six methods and techniques continuously during the simulation run. The simulation model itself consists of a "virtual" part that deals with the continuous processes and the functioning of the simulation model (*i.e.* the logic engine high-level building block) and a "real" part that deals with the behaviour of the modules (*i.e.* the four different high-level building blocks of the ERM method). The five high-level building blocks can facilitate the construction of simulation models of stochastic continuous systems. The use of the variables technique ensures that simulation models that are developed with the generic methodology do not need a warm-up period and therefore it supports the short simulation runtimes and accurate modelling ability criteria.

* * * * *