2. LITERATURE REVIEW

This chapter reviews previous published work on the control of multivariable systems, and the methods to assess interaction in these systems. It also contains a general review on dynamic simulation and its application to refrigeration systems dynamics and control.

2.1. INTRODUCTION TO THE CONTROL OF MULTIVARIABLE SYSTEMS

Control of multivariable systems has developed dramatically in recent years due to the rapid development in computer technology. Although the principles of control do not differ from the classical single loop control principles, the application and the implementation of the principles is more complex. In this section a review of the subject is presented as a basis for application of relevant techniques in refrigeration system control.

The concepts and applications of multivariable process control are wide, difficult to explain, and cannot be covered totally in such a small space. Consequently some sort of classification is required to facilitate the presentation of these concepts. In this thesis, the concepts are presented in such a way that parallels the development of controller design:

1. The methods to select the variables to be included in the control problem from all the available variables are presented. These methods require the analysis of the process to assess some of its criteria by using several methods and measures. The

main methods used are:

- Input and Output Effectiveness
- Controllability
- Resiliency

2. The measures to analyse the interaction within the selected set of measured and manipulated variables, and guide the selection of the appropriate input / output pairing. Several measures were proposed in the literature, and some found more acceptance than the others. A few measures are discussed in this thesis, with more interest given to the ones used actually in the analysis of the refrigeration system. These measures are:
   - Relative Gain Array (RGA)
   - Relative Interaction Array (RIA)

3. The measures that evaluate the interaction in fixed pairings. These measures rank pairings in the form of single numbers evaluated using specific procedures. The measures discussed in detail in this thesis are:
   - Jacobi Eigenvalue Criterion (JEC)
   - Dynamic Interaction Measure (DIM)
   - Performance Interaction Measure (PIM)
   - μ Interaction Measure (μIM)

4. The methods to assess the closed loop control system design. Two major approaches can be used:
   - Analytical methods that assess the stability and robustness of the control systems. These methods require the information regarding the controller design. However, since this information is not usually available at an early stage of the process design, the usage of these methods is limited, and therefore are not covered in detail here.
   - Time simulations that use the developed linear and /or non-linear models to examine the performance of the control systems. This method is preferred here as it gives better insight into the dynamics of the system.

In order to understand some of the methods and measures presented later, essential mathematical principles and control concepts need to be presented. The aim of the presentation is to aid understanding the terms used in the subsequent sections.

2.2. BASIC CONCEPTS AND MATHEMATICAL PRINCIPLES

In this section, basic concepts and mathematical principles are defined for use in discussion of the methods of multivariable control systems analysis presented later.

2.2.1. State Space Representation

In the time domain, a system can be described in general by a set of linear differential and algebraic equations (i.e. state space model):

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{Ax} + \mathbf{Bu} \tag{2.1}$$

$$\mathbf{y} = \mathbf{Cx} + \mathbf{Du} \tag{2.2}$$

where **x** is the vector of state variables, **u** is the vector of inputs (manipulated variables), **y** is the vector of outputs (controlled / measured variables), and **A**, **B**, **C**, and **D** are constant matrices of appropriate dimensions.

Taking the Laplace transform of Equations 2.1 and 2.2 with zero initial conditions, and re-arranging, the system transfer function matrix **G**(s), the relationship between the inputs **u** and the outputs **y** relates the state model equations 2.1 and 2.2 as:

$$\mathbf{G}(s) = \frac{\mathbf{y}(s)}{\mathbf{u}(s)} = \mathbf{C}\,(s\mathbf{I} - \mathbf{A})^{-1}\,\mathbf{B} + \mathbf{D} \tag{2.3}$$

Representation of the state space model was developed based on graph theory principles. Two main types are used:

1. The signal-flow-graph (SFG) type. See Henley and Williams (1973) for detailed discussion.

2. The matrix representation (Russell and Perkins, 1987).

The first type is easier to use and thus has been adopted by other researches (Johnston, 1990; Wang and Cameron, 1993), and it has also been used here for the same reason.

Each SFG consists of directed branches interconnected at nodes. In the state space model, the nodes represent the variables (signals), and the branches connecting the nodes indicate that these nodes are related. Each branch is assigned a numerical value or a function, which quantifies the relationship between the two variables in terms of a gain factor or a transfer function.

2.2.2. Eigenvalues

In a square matrix **A** of dimensions $n \times n$, the eigenvalues λ_i , i=1,..*n*, are the *n* roots of the polynomial equation 2.4 (Luyben, 1990)

$$\text{Det}\,[\lambda\mathbf{I} - \mathbf{A}] = 0 \tag{2.4}$$

In a state space model, the eigenvalues of the matrix **A** are the roots of the characteristic equation of the system. They are essential for the analysis of the dynamic behaviour of the systems as they can show the stability, the speed and the nature of the system's dynamic response.

The product of the eigenvalues equals the determinant of the matrix **A**, and their sum equals the trace of **A**, denoted as tr(**A)**. Also, the spectral radius of the matrix **A**, denoted as $\rho(\mathbf{A})$, is defined as the magnitude of the largest absolute eigenvalue of **A** (Doyle, 1982).

$$\rho(\mathbf{A}) = \max\,(\,|\,\lambda_i\,|\,) \tag{2.5}$$

2.2.3. Matrix Norms

A norm $\|\cdot\|$ is a single number measuring the size of a vector, a matrix, a signal or a system. The commonly used matrix norms in control theory are:

1. $\|\mathbf{A}\|_1$ induced norm, defined as the maximum column sum of the matrix **A**

$$\|\mathbf{A}\|_1 = \max_j \left(\sum_i |a_{i,j}| \right) \tag{2.6}$$

2. $\|\mathbf{A}\|_\infty$ induced norm, defined as the maximum row sum of matrix **A**

$$\|\mathbf{A}\|_\infty = \max_i \left(\sum_j |a_{i,j}| \right) \tag{2.7}$$

3. $\|\mathbf{A}\|_2$ induced norm, also known as the Frobenius or Euclidean norm, defined as the square root of the spectral radius of the matrix ($\mathbf{A}^T\mathbf{A}$).

$$\|\mathbf{A}\|_2 = \sqrt{\rho\ (\mathbf{A}^T\ \mathbf{A})} \tag{2.8}$$

It should be noted that the spectral radius of the matrix **A** provides a lower bound on any matrix norm such that:

$$\rho(\mathbf{A}) \leq \|\mathbf{A}\| \tag{2.9}$$

2.2.4. Singular Value Decomposition (SVD)

Singular value decomposition (SVD) is a unique decomposition for a given general matrix **G** (not necessarily the system transfer function matrix) which is reduced to a diagonal form by premultiplying and postmultiplying it by unitary matrices (Golub and Van Loan, 1983; Stewart, 1973) as follows:

$$\mathbf{G = U\Sigma V}^T \tag{2.10}$$

Σ is the diagonal matrix of singular values diag(σ_1 , σ_2 , ..., σ_q). σ_i is the i^th^ singular value, and $\sigma_1 \geq ... \geq \sigma_q > 0$. The columns of **U** are known as left singular vectors, and the columns of **V** are known as right singular vectors. **U** and **V** are both unitary, i.e., for **U** matrix: $\mathbf{U}^T\mathbf{U} = \mathbf{I}$.

By definition, the singular values are also given as the positive square roots of the non-zero eigenvalues of the Hermitian Matrix $\mathbf{G}^T\mathbf{G}$ or $\mathbf{G}\mathbf{G}^T$ (Karlsmose *et al.*, 1994)

$$\sigma_i = \sqrt{\lambda_i \left(\mathbf{G}^T \ \mathbf{G}\right)} \tag{2.11}$$

and it can be shown also that

$$\sigma_{max}(\mathbf{G}) = \|\mathbf{G}\|_2 \tag{2.12}$$

The singular values of a matrix are a measure of how close the matrix is to being "singular", i.e., having a determinant that is zero (Koung and MacGregor, 1992).

2.2.5. Condition Number

The condition number γ of the matrix **G** is defined as the ratio between the largest and the smallest singular value

$$\gamma = \frac{\sigma_{max}}{\sigma_{min}} \tag{2.13}$$

Another way to calculate it has been also defined as (Joseph and Brosilow, 1978; Grosdidier *et al.*, 1985; Nett and Manousiouthakis and, 1987):

$$\gamma = \|\mathbf{G}\|_2 \, \|\mathbf{G}^{-1}\|_2 \tag{2.14}$$

In general, large values of γ indicate that the matrix **G** is closer to being singular.

2.2.6. Structured Singular Value (SSV)

The structured singular value (SSV), denoted μ, was introduced by Doyle (1982). It involves the matrix problem of determining necessary and sufficient conditions such that

$$\text{Det}(\mathbf{I} + \mathbf{G}\Delta) \neq 0 \tag{2.15}$$

where **G** is the transfer function matrix from the collective outputs of the perturbations to their inputs, and Δ is a diagonal perturbation matrix containing the uncertainty of the system.

Mathematically, it is defined as:

$$\mu(\mathbf{G}) = \begin{cases} 0 \text{ if no } \Delta \text{ solves Det}(\mathbf{I} + \mathbf{G}\Delta) = 0 \\ \left(\min\limits_{\Delta} \{\sigma_{\max}(\Delta) \mid \text{Det}(\mathbf{I} + \mathbf{G}\Delta) = 0\}\right)^{-1} \end{cases} \tag{2.16}$$

which means that $\mu^{-1}(\mathbf{G})$ is equal to the smallest maximum singular value of Δ which is needed to make (**I** + **G**Δ) singular.

The computation of μ is complicated and no single method exists for its exact calculation (Hovd *et al.*, 1994). Therefore several bounds were found to estimate its value. Hovd *et al.* (1993) stated that the bounds are almost always within 1-2% of the real values, and for engineering purposes μ never has to be calculated exactly.

Skogestad *et al.* (1988) and Morari and Zafiriou (1989) concluded that μ is bound by the spectral radius and the maximum singular value of **G** such that:

$$\rho(\mathbf{G}) \leq \mu(\mathbf{G}) \leq \sigma_{\max}(\mathbf{G}) \tag{2.17}$$

Doyle (1982) stated and proved several properties of μ. His work was followed by many researches, who expanded the μ-approach and explored methods to compute it for specific cases, e.g. Fan and Tits (1986), Nett and Uthgenannat (1988), Morari and Zafiriou (1989), and Yamamoto and Kimura (1995).

2.2.7. Relative Gain Array (RGA)

The relative gain array (RGA) was introduced by Bristol (1966). A theoretical basis for it was later developed by Tung and Edgar (1981), Grosdidier *et al.* (1985), and Hovd and Skogestad (1992). For a non-singular *m* × *m* matrix **G** of steady state gain factors (i.e. **G**(0)), the RGA, denoted by Λ, is in general an *m* × *m* matrix defined as:

$$\Lambda = \mathbf{G} \otimes (\mathbf{G}^{-1})^{T} \qquad (2.18)$$

where ⊗ denotes Hadamard (element by element or Schur) multiplication.

The sum of each row and each column in Λ equals 1, and any permutation of the rows and columns of **G** results in the same permutation in Λ.

A full discussion regarding the applications of RGA in interaction analysis and pairing selection is presented in Section 2.4.1.

2.2.8. Scaling of Process Variables

Scaling can be viewed as the choice of units for process variables when modelling the plant. It requires the engineer to make a judgement at the start of the design process about the required performance of the system regarding the allowed magnitude of each input, the allowed deviation of each output, and the expected magnitudes of disturbances. Unlike system properties which are independent of variable scaling such as the RGA and the eigenvalues of **A** and **G**, SVD analysis is scaling dependent, i.e. for the same system model using different variable units leads to different SVD matrices which consequently result in different measures of system behaviour. Because of this dependence, the variable scaling choice is crucial when using analysis methods based on SVD.

Several authors referred to the scaling as a problem when using SVD, but with few exceptions, most pass over the issue in a few sentences and give no practical solutions of how to handle it (e.g. Smith, 1981; McAvoy, 1983a; Grosdidier *et al.*, 1985; Grosdidier and Morari, 1986; Grosdidier and Morari, 1987; Chang and Yu, 1990; Hovd and Skogestad, 1992; Brambila and D’Elia, 1992; Alatiqi *et al.*, 1994; Karlsmose *et al.*, 1994; Sågfos and Waller, 1995).

Moore (1986) suggested that a good scaling can be expressed in terms of percent changes such that the units of **G**(s) are defined as:

$$\text{Scaling unit} = \frac{\%\ \text{of}\ \ \text{sensor}\ \ \text{span}}{\%\ \ \text{of}\ \ \text{range}\ \ \text{of}\ \ \text{manipulator}} \tag{2.19}$$

Luyben (1990) also suggested using dimensionless gains in the transfer functions matrix such that the gains with engineering units should be divided by the appropriate transmitter spans and multiplied by the appropriate valve gains. The principle of this method is also similar to Moore's suggestion. This method has been applied by Alatiqi *et al.* (1994) on two anaerobic digestive schemes, and by Skogestad and Postlethwaite (1996).

Cao *et al.* (1996) scaled the state and output variables by dividing their values by their associated steady state values. The inputs were scaled to be within the range of ±1 by dividing their values by scaling factors that equal the smaller of the difference between either the higher or lower limits and the steady state values of the associated variables. Based on a similar concept, Skogestad (1997), and Laush *et al.* (1998) scaled all variables to be in an interval of ±1, where inputs were scaled in respect to their constraints, outputs in respect to their expected variations, and disturbances in respect to their expected range.

Keller and Bonvin (1987) made an alternative suggestion that the variables all be scaled so as to exhibit a unity steady state gain. The same method was used by Alsop and Edgar (1990).

Waller and Waller (1995) gave an excellent review of the scaling issue, and they recommended using the scaling that leads to the minimised condition number defined in the next section. The same concept was earlier applied by Nguyen *et al.* (1988), who developed a scaling policy known as G-balancing. This method is iterative and gives results close to the minimised condition number in a matter of 2-3 iterations.

2.2.9. Minimised Condition Number

The minimised condition number $\gamma_{\min}$ of the system transfer function matrix **G** is obtained by minimising the condition number with respect to all input and output scaling. It is defined in Equation 2.20 where $\mathbf{S}_1$, $\mathbf{S}_2$ are real diagonal matrices with non-zero diagonal elements (Grosdidier *et al.*, 1985; Nett and Manousiouthakis, 1987; Maciejowski, 1989).

$$\gamma_{\min}(\mathbf{G}) = \min_{S1,S2} \gamma\,(\mathbf{S}_1\,\mathbf{G}(s)\,\mathbf{S}_2) \qquad (2.20)$$

Methods to calculate the minimised condition number were reported frequently in the literature in terms of upper or lower bounds (Grosdidier et al, 1985; Skogestad and Morari, 1987; Nett and Manousiouthakis, 1987). Grosdidier *et al.* (1985) obtained an upper bound on the minimised condition number by scaling the process transfer function matrix **G** until the minimised condition number is obtained. For the matrix process transfer function **G** of dimensions $n \times n$, this bound is expressed in Equation 2.21, where Λ is the relative gain array RGA, $\|\Lambda\|_{\mathbf{1}}$ is the 1-norm of the RGA, and and $\|\Lambda\|_{\infty}$ is the infinity-norm of the RGA.

$$\gamma_{\min} \leq 2 \max\,(\|\Lambda\|_{\mathbf{1}}, \|\Lambda\|_{\infty}) \qquad (2.21)$$

Nett and Manousiouthakis (1987) obtained a lower bound on the minimised condition number defined as:

$$\gamma_{\min} \geq \max\,(\|\Lambda\|_{\mathbf{1}}, \|\Lambda\|_{\infty}) \qquad (2.22)$$

2.3. INPUT / OUTPUT VARIABLE SETS

A key issue in designing multivariable control systems is the appropriate choice of input (manipulated) and output (measured) variables, as a wrong set of variables may impose fundamental limitations on the system's performance that cannot be overcome even by complex control strategies. The need to find simple methods to assess the choice is more important for large-scale control systems where the number of pairing possibilities becomes huge. A summary of some of the recently suggested methods can be found in van de Wal and de Jager (1995).

Cao and Rossiter (1997) identified that for a process with *M* candidate outputs and *N* candidate inputs, the total number of control schemes CS with *m* measurements and *n* manipulated variables can be computed as

$$\mathrm{CS} = \sum_{m=1}^{m} \sum_{n=1}^{n} \frac{M!}{m!(M-m)!} \frac{N!}{n!(N-n)!} \tag{2.23}$$

The main criteria to select the set of the inputs and outputs is to achieve robust stability and accepted performance. A basic method is to test each selection for the number and location of right half plane poles and zeros (van de Wal and de Jager, 1995). Other methods were developed to try to automate the selection, see, e.g Cao *et al.* (1997a, 1997b).

Most of the research in this area was directed towards the selection of the appropriate measurements. Downs in 1984 used an SVD based method to find controlled variables from a large number of choices (Yu and Luyben, 1987). The method depends on analysing the elements of the matrix **U** resulting from a SVD of the plant steady state gain matrix **G**(0). The largest elements in each column of the **U** matrix indicate which outputs of the process are the most sensitive, and hence which to select as controlled variables.

This method has been applied by several researchers. Moore (1986; 1992) used this method to select the temperature sensors in a 50 tray distillation column. Yu and Luyben (1987) used this method to select the location of temperature measurements in controlling multicomponent distillation columns. Bequette and Edgar (1989) used the SVD to select the appropriate measurements in the control system in a distillation column, This method was also used by Luyben (1990) to select the best variables to be controlled in a distillation column, Chylla and Çinar (1990) to select the most sensitive temperature in a reactor case study, and Alatiqi *et al.* (1994) to determine the best controlled variables in two anaerobic digestive schemes. Keller and Bonvin (1987) used also an SVD based method to select the input variables for the purpose of model reduction and control design.

2.3.1. Input and Output Effectiveness

Cao and Rossiter (1997) developed a method based on SVD to select inputs, and extended it later to select outputs as well (Cao *et al.*, 1997b). The technique they developed is called the Input Effectiveness IE, denoted as η_I, and the Output Effectiveness OE, denoted as η_O. A simple method to calculate these measures is by utilising the non-square RGA (NRGA) of the matrix **G** (see Section 2.4.1). The sum of the ith column of NRGA equals the square of the Input Effectiveness, and the sum of its ith row equals the square of the Output Effectiveness.

Hence for a **G** matrix of size *m* × *n*, the following relations were defined:

$$\sum_{i}^{m} \lambda_{ij} = \eta_I^{2} \tag{2.24}$$

$$\sum_{i}^{n} \lambda_{ij} = \eta_O^{2} \tag{2.25}$$

The candidate input variables are ranked by selecting the best variables in terms of having the most effect on the output variables. The opposite is done to the output variables. This simply means to select the variables that have η greater than 0.5, and eliminate the variables with η less than 0.5. The method should be repeated with the remaining variables until the desired number of candidates is reached. It should be noted that after each elimination η values always increase, and that these measures are scaling dependent since η_I is input scaling dependent and is η_O output scaling dependent.

2.3.2. Controllability:

The term “controllability” is a widely mentioned term in process control literature. However, the exact definition of the term is rather ambiguous. In this thesis, it refers to input-output rather than the usual controllability term in old control literature. Some of the definitions that go online with the meaning adopted in this thesis are mentioned below. According to Moore (1986) it indicates the practicality to satisfy the entire set of control objectives, or in other words, the ability of a process to be operated economically and safely without violating the constraints imposed to achieve various design objectives in the presence of various uncertainties (Kwon and Yoon, 1996).

In process control, the condition number γ of **G** (defined in Equation 2.13) has been widely used recently as a measure for controllability, where it is taken to show on a relative basis how much more difficult it will be to control a process using different alternatives. According to Moore (1986) a large $\gamma(\mathbf{G})$ indicates that it will be impractical to satisfy the entire set of control objectives. However the quantitative meaning of "large" varies between from 10 (Lau *et al.*, 1985), to 50 (McAvoy, 1983a), to 100 (Joseph and Brosilow, 1978).

Alsop and Edgar (1990) studied the control of a high purity distillation column and used the condition number of **G** to screen the selection of process inputs from an input-output model description. Papastathopoulou and Luyben (1991) used the condition number $\gamma(\mathbf{G})$ alongside other steady state indexes to decide alternative control structures for a binary distillation column. They said that the smaller $\gamma(\mathbf{G})$ is, the better is the control, but their analysis results showed some inconsistency between γ and other applied indexes. Alkaya *et al.* (1992) used the same concept to select the best controlled variables in a multicomponent high-purity distillation column, and their evaluation was based on selecting the pair which has the lowest condition number obtained amongst different pairs. This approach was used earlier by Joseph and Brosilow (1978), but proved to be impractical to find the set which gives the minimum condition number by going through all possible combinations. Recently, Dimian *et al.* (1997) used the same condition number to assess alternative control structures in a plant-wide process that handles the removal of impurities in a plant with nested loops.

McAvoy (1983a) used the condition number $\gamma(\mathbf{G})$ as a measure of decoupling. Lau *et al.* (1985) used it as an indication of sensitivity of the system. They considered that if $\gamma(\mathbf{G})$ is less than 10 in the specified frequency range, it implies good condition, which means that modelling uncertainties can be tolerated. Nguyen *et al.* (1988) used $\gamma(\mathbf{G})$ as an indicator of the inherent controllability of a process, and pointed to its significance as the most useful index in assessing the controllability of alternative plant designs.

Sågfos and Waller (1995) defined a system to be “ill-conditioned” if the condition number of the process matrix is large. Waller and Waller (1995) stated that the condition number is used as a measure when comparing or designing different control structures. Karlsmose *et al.* (1994) also suggested that systems with high condition number are termed “ill conditioned” and are inherently difficult to control.

Grosdidier *et al.* (1985), Skogestad and Morari (1987), and Skogestad and Havre (1996) investigated using the RGA as an indicator to assess controllability. Grosdidier *et al.* (1985) stated that the condition number is related rigorously to sensitivity and robustness, and that ill conditioned systems show both high condition numbers and large elements in RGA. The same was concluded by Skogestad and Morari (1987) and Skogestad and Havre (1996) who said that large minimised condition number and large RGA elements are causes of concern. All these authors concluded that RGA can be used as an additional indication of controllability and concluded that systems with large RGA elements are inherently difficult to control and sensitive to modelling errors.

Few authors used the singular values instead of the condition number as controllability indexes, where larger values of singular values indicate better controllability. Nguyen *et al.* (1988) and Cao *et al.* (1996) used the minimum singular value for this purpose.

2.3.3. Resiliency

Morari (1983) defined resiliency as the ability of the plant to move fast and smoothly from one operating condition to another (including start-up and shut-down) and to deal effectively with disturbances. Luyben (1990) considered it as the ease of controllability, i.e. some choices of manipulated and controlled variables produce systems that are easier to control than others.

Morari (1983) developed the Morari Resiliency Index (MRI) which is simply the minimum singular value of the process transfer function matrix **G**(s). His criterion is that the larger the value of MRI the more resilient is the process which can be explained that the larger the minimum singular value of a matrix it is further from being singular and hence easier to find its inverse. This index is a useful tool for comparing alternative processes and alternative manipulating variables as it depends only on the controlled and manipulated variables and not on the pairing of the variables or the controllers tuning.

MRI method was applied by Johnston and Barton (1984) and Nguyen *et al.* (1988). Luyben (1990) used it as a way to determine the manipulated variables choice in the process where he recommended using the set of manipulated variables which have higher resiliency, and it also was used by Papastathopoulou and Luyben (1991) to study alternative control structures for a binary side-stream distillation column. Recently, Laush *et al.* (1998) used it alongside other methods to assess resiliency of control structures in a plant-wide system that handles reaction, recovery and purification processes.

2.4. GUIDANCE ON INPUT / OUTPUT PAIRINGS

The selection of variable pairing is seen as a main step in designing decentralised multivariable controllers. The selection is done usually so as to minimise the interaction in the system so that the multivariable system resembles the SISO independent loops as closely as possible. Here, interaction means that in a multivariable control system each manipulated variable affects more than one controlled variable.

The linkage between the pairings selection and interactions lead to the development and utilisation of several interaction measures to assist in the choice of pairing, see, e.g. Jensen *et al.* (1986) and Khelassi (1991) who both reviewed different interaction measures. In general these measures can be classified into two main categories: measures that assess the interaction by examining the process itself, and measures that include the controller design in its assessment.

Examples of the measures that involve a fixed controller design include the Dynamic Block Relative Gain (Reeves and Arkun, 1989), and the General Interaction Factors (Balchen and Mummé, 1988). These measures are not discussed here as using them requires a full knowledge of the control system structure and design. This undermines the main advantage of interaction analysis of assessing the system by analysing the plant model only.

The most famous interaction measure that utilises the plant model only is the Relative Gain Array, defined in Section 2.2.7. Recently other measures were also suggested including several extensions of the RGA, the Structural Interaction Array SIA (Johnston, 1990), and the Relative Interaction Array RIA.

Johnston (1990) looked at this problem and based on the graph theory and using the state space model developed what he called the Structural Interaction Array SIA. This array gives the ratio between the direct effect between each input variable u and its corresponding output variable y and the total closed loop gain for the same two variables subject to a unit change in each output variable. His analysis becomes very complicated with higher order systems as it analyses the Jacobian matrix **A** of state variables, which has the major drawback of prohibitive computational effort. It also requires the determination of the steady state values of the u variables for a unit change in each y.

2.4.1. The Relative Gain Array and Its Extensions

This is the most famous and most widely used tool in interaction analysis due to its simplicity. Bristol (1966) defined relative gain as the ratio of an open loop gain (when the other loops are on manual) to the same loop gain where the other loops are all under perfect control. With perfect control, a controlled variable is held at its set point no matter what disturbances attempt to change it. Calculation of the RGA was introduced in Section 2.2.7, and requires only the steady state gain information. Hence for a steady state gain matrix **G**(0), the RGA Λ is calculated as:

$$\Lambda = \mathbf{G}(0) \otimes (\mathbf{G}(0)^{-1})^{\mathrm{T}} \qquad (2.26)$$

Table 2.1 summarises the properties associated with the RGA values (Shinskey, 1984; 1988; 1992; Ogunnaike and Ray, 1994).

It should be noted however that the RGA measures the interaction in the individual loops in the system, and the closeness of each element to 1 is a qualitative measure and does not quantify the amount of interaction.

Since the RGA is a dimensionless value, it is not affected by scaling of variables. But in needing only the steady state gain matrix, is a disadvantage as it does not take into account the effect of the process dynamics. Another limitation is that it is only applicable to square matrices and only works if perfect controllers are applied.

In his original RGA proposal, Bristol (1966) gave an RGA pairing rule that pairing should be done always on positive RGA elements closest to unity. McAvoy (1981; 1983a) extended the pairing rules of RGA to check its stability, and suggested an improved pairing rule which states the pairing should also be checked by the Niederlinski Index*. If found unstable the next positive pairing closest to 1 is chosen, if possible avoiding negative pairings. Zhu (1996) added that large RGA elements (i.e. much larger than 1) should also be avoided.

* ***Niederlinski Index (NI):***
This method was introduced by Niederlinski (1971) as a necessary but not sufficient condition for stability for closed-loop systems with integral action. It applies only when integral action is used in all loops. The Niederlinski Index (NI) is defined as:

$$\mathrm{NI} = \frac{|\,\mathbf{G}(0)\,|}{\prod_{i=1}^{n} \mathbf{G}(0)_{ii}}$$

where Π is the product of the diagonal elements of **G**(0), and |**G**(0)| is the determinant of **G**(0).

The criterion states that if NI is negative the closed-loop system is unstable, and if not the system may or may not be stable.

Zhu and Jutan (1993) extended the stability Niederlinski Index NI to measure the overall interaction of a system, and stated that variables should be paired in such a way that the resulting pairing corresponds to an NI closest to 1. This rule lacks any theoretical basis, and was shown that it may result in incorrect pairing choices (Zhu, 1996).

Table 2.1: Significance of Relative Gain Array values features

RGA value	Properties
$\lambda < 0$	Interaction acts in the opposite direction to the interaction-free process gain. Negative numbers indicate that the sign of controller gain may have to be switched when auto/manual loop transfers are made
$\lambda = 0$	Interaction acts in the reverse direction to interaction-free process gain
$0 < \lambda < 1$	Interaction acts in the same direction as interaction-free process gain. The closer the diagonal element to 1, the less interaction exists
$\lambda = 1$	No interaction
$\lambda > 1$	Interaction acts in the same direction and more strongly than the interaction-free process gain. Large numbers indicate very high interaction
$\lambda \rightarrow \infty$	Loops are completely dependent

Smith (1981) and McAvoy (1983a) combined the analysis of both RGA and SVD. Smith (1981) used SVD of the steady state gain matrix as a complementary technique with RGA to choose pairings especially in the case of processes near singularity. McAvoy (1983a) used the condition number of the steady state gain matrix as a measure of decoupling feasibility and recommended to use SVD as a check to linear decoupler design based on RGA.

Zhu and Jutan (1993) proposed using an empirical RGA based measure to assist in the pairing choice in cases where ambiguities occur when more than one set of pairings satisfy the RGA pairing rule. They stated that the final pairing amongst the chosen alternatives is the one in which the relative gains exhibit the minimum overall distance from 1. i.e,

$$\min \sum \left| \lambda_{ij}^{k} - 1 \right| \tag{2.27}$$

where λ_{ij}^{k} is the paired RGA elements corresponding to the kth altrernative.

Zhu (1996) showed that this rule may result in incorrect pairing choices, and he developed the RIA pairing rule (see next section) instead.

Despite its success and its adoption as a first pairing rule, there were reported cases in which the RGA resulted in incorrect results. For example, Hovd and Skogestad (1992) showed a case in which the RGA pairing resulted in an unstable system despite that each individual loop was stable.

Chang and Yu (1989) extended the RGA to include non-square matrix systems. Their array, denoted NRG or Λ^{N}, is similar to Bristol's RGA (including using only steady state gain matrix) with only one difference, that is using the pseudo-inverse $\mathbf{G}^{+}$ instead of the normal matrix inverse $\mathbf{G}^{\mathbf{-1}}$ so

$$\Lambda^{N} = \mathbf{G}(0) \otimes (\mathbf{G}(0)^{+})^{T} \qquad (2.28)$$

NRG measures quantitatively the ultimate steady state performance of a non-square control system. The closeness of each row to unity has a strong implication in non-square control performance. Small values indicate that the specific output will be controlled poorly if a non-square controller is used. Therefore it can be used to select the controlled and manipulated variables. In their examples NRG gave results similar to SVD based methods when the systems examined were properly scaled. It should be noted, however, that unlike the RGA the NRG is dependent on output scaling.

Several authors extended the RGA to account for dynamic features such as Witcher and McAvoy (1977), Bristol (1978), McAvoy (1981) and Tung and Edgar (1981). In terms of matrix operations, the relative dynamic gain array RDGA for the process transfer function matrix **G**(s) is expressed as (Bristol, 1978; Jensen *et al.*, 1986; Hovd and Skogestad, 1992)

$$\Lambda(s) = \mathbf{G}(s) \otimes (\mathbf{G}(s)^{-1})^{T} \qquad (2.29)$$

The information contained in the RDGA is best expressed by plotting it as an array of polar plots (Jensen *et al.*, 1986) or as a magnitude and phase plots (Khelassi, 1991). Witcher and McAvoy (1977) stated that interaction is small if the magnitude of the diagonal elements of Λ(s) is close to 1, and the magnitude of all other elements is small.

The RDGA was used recently by several researchers to assess interaction, see, e.g. (Schmidt *et al.*, 1992, Karlström *et al.*, 1992; Karlsrome *et al.*, 1994; Dimian *et al.*, 1997; Kookos and Lygeros, 1998). Despite this acceptance, some have their reservations on its usage, e.g. Jensen *et al.* (1986) and Grosdidier and Morari (1986; 1987). Luyben (1990) said that the RGA itself is not an effective tool for control applications.

McAvoy (1983a; 1983b) extended the RGA to account for pure integrals. For a 3 ×3 matrix he showed that the 1/s terms are cancelled out, and therefore the RGA for the integrating system **G**(s) turned out to be the same as the RGA of the non-integrating system. Arkun and Downs (1990) generalised this method, and developed a general method to calculate RGA for integrating processes based on the state space model and SVD techniques.

Hovd and Skogestad (1992) proposed the performance relative gain array PRGA, denoted Γ, as an extension to the RGA to indicate the existence of one-way coupling

$$\Gamma(s) = \mathbf{G_d}(s)\, \mathbf{G}(s)^{-1} \tag{2.30}$$

where $\mathbf{G_d}$ is the diagonal elements of **G**. PRGA needs to be calculated any time **G** is rearranged. Also it is dependent on output scaling (i.e. $\Gamma(\mathbf{G}) \neq \Gamma(\mathbf{DG})$). Plants with small values are preferred, and plots of γ_{ii} are used to evaluate the achievable closed loop performance. Grosdidier (1990) originally introduced the PRGA at steady state, and stated that RGA can be used as an indicator of the performance loss caused by interactions in the decentralised control systems, and linked it to the diagonal elements of the matrix Γ(0) such as

$$\gamma_{ii} = \lambda_{ii} \tag{2.31}$$

2.4.2. Relative Interaction Array (RIA)

This method was proposed by Zhu (Zhu, 1996; Zhu and Jatan, 1996) as an improved interaction measure based on the RGA, but unlike the RGA, this measure represents the relative amount of interaction in the loop thus it is quantitative. RIA, denoted ϕ, is defined as

$$\phi_{ij} = \frac{1}{\lambda_{ij}} - 1 \tag{2.32}$$

Table 2.2 summarises the properties associated with the RIA values (Zhu, 1996).

Table 2.2: Significance of Relative Interaction Array values features

RIA value	Properties
$\phi < -1$	Reverse interaction dominates over interaction-free process gain
$-1 < \phi < 0$	Interaction acts in reverse direction as interaction-free process gain
$\phi = 0$	No interaction
$0 < \phi < 1$	Interaction acts in the same direction as interaction-free process gain
$\phi > 1$	Interaction dominates over interaction-free process gain

The difference between RIA and RGA is that the distance of an RIA element from 0 actually quantifies the amount of interaction in a loop.

Zhu (1996) proposed the RIA paring rule which states that variables should be paired such that all RIA elements area as close as possible to 0, NI is positive, the RIA elements are greater than –1, and the elements close to –1 should be avoided.

The best pairing is chosen as the pairing that exhibits the least amount of interaction. And since the distance between any RIA element and 0 actually quantifies the amount of interaction, the pairing whose elements exhibit the minimum overall distance from 0. i.e,

$$\min \sum \left|\phi_{ij}^{k}\right| \tag{2.33}$$

where ϕ_{ij}^{k} is the paired RIA elements corresponding to the kth altrernative, is chosen.

This method was compared to other pairing measures by Zhu (1996) who said it can identify the appropriate pairings. It was later used by Kookos and Lygeros (1998).

2.5. PERFORMANCE MEASURES FOR FIXED INPUT / OUTPUT PAIRINGS

Several interaction measures were suggested that quantify the amount of interaction in a system in terms of a specific number. These measures assume a fixed pairing in the system to perform their calculations. To compare between two different pairings, the calculation procedure must be performed twice. Examples of these measure are the Dynamic Interaction Measure (Lau *et al.* 1985), and the μ Interaction Measure (Grosdidier and Morari, 1986; 1987). Some of these measures are discussed in detail below.

2.5.1. Jacobi Eigenvalue Criterion (JEC)

This criterion (Mijares *et al.*, 1986) is a mathematical criterion based on the difficulty caused by the interaction terms (the off-diagonal elements in **G**(0)) in finding the inverse of the steady state gain matrix **G**(0). It considers the best pairing as the one most closely resembles a set of independent single-loop systems. For the process matrix **G**(0) arranged in such a way that the pairing is on its diagonal, $\mathbf{G_d}$ is the matrix of the diagonals. The Jacobi iteration matrix **J** is defined as

$$\mathbf{J} = \mathbf{I} - \mathbf{G_d}^{-1}\,\mathbf{G} \tag{2.34}$$

The best pairing is the one with the smallest spectral radius of **J**, (min ρ(**J**)).

Papastathopoulou and Luyben (1991) applied this JEC criterion in a study on controlling a binary side-stream distillation column. Zhu (1996) used it as a comparison to his RIA pairing rule, where he showed some inconsistencies.

2.5.2. Dynamic Interaction Measure (DIM)

The DIM measure θ was developed by Lau *et al.* (1985). It indicates how far the process transfer function matrix **G**(s) is from being a completely decoupled system. If $\sigma_1(s)$, $\sigma_2(s)$, ..., $\sigma_q(s)$ are distinct singular values of **G**(s), then **U**(s) and **V**(s) can be partitioned:

$$\mathbf{U}(s) = [\mathbf{u}_1(s) \mid \mathbf{u}_2(s) \mid \ldots \mid \mathbf{u}_q(s)] \tag{2.35}$$

$$\mathbf{V}(s) = [\mathbf{v}_1(s) \mid \mathbf{v}_2(s) \mid \ldots \mid \mathbf{v}_q(s)] \tag{2.36}$$

where $\mathbf{u}_i(s)$, $\mathbf{v}_i(s)$ (i = 1, ..., q) are the singular decomposition vectors which correspond to the i^{th} singular value.

The total interaction measure of the system θ is obtained by

$$\theta = \cos^{-1}\sqrt{\frac{\sum_{i=1}^{n}(\sigma_i^2 \quad \cos^2\theta_i)}{\sum_{i=1}^{n}\sigma_i^2}} \tag{2.37}$$

where

$$\theta_i = \cos^{-1}(w_{ij}\, w_{ij}^{+})^{½} \tag{2.38}$$

and

$$\mathbf{W_i(s) = u_i\ (s)\ v_i^T(s)} \tag{2.39}$$

The maximum entry of **W** defines a u_i-y_j pairing, i.e. the largest vector component of $\mathbf{u}_i$ is paired with the largest vector component of $\mathbf{v}_i$).

The θ values are plotted against log frequency. Values range between 0-90°, with a value of zero indicating no interaction exists, and values above 15° suggesting the need to introduce some compensation to reduce the level of interaction within the system.

It should be noted that the DIM does not assume a fixed measured-manipulated variable pairing, but automatically determines an optimum pairing at each frequency. The pairing procedure was suggested by Lau *et al.* (1985) and also by Moore (1986).

It states that for the process transfer function matrix **G**, the pairing which will yield the least open loop multivariable interaction is one in which the sensor associated with the largest component of the column vector $\mathbf{u}_i$ is paired with the manipulated variable associated with the largest vector of the column $\mathbf{v}_i$.

However, this method may fail on some occasions, e.g. if the largest element in the different column vectors of **U** happens to be the same output. Also Moore (1992) stated that this analysis cannot predict closed loop stability problems that other indexes such as NI and RGA pairing rule can predict.

2.5.3. Performance Interaction Measure (PIM)

Feng and Grimble (1989) introduced the performance interaction measure PIM, denoted τ. For a given feedback control system with fixed pairing, PIM is defined as:

$$\tau = \sigma_{max}(\mathbf{U}\mathbf{V}^T\text{-}\mathbf{I}) \qquad (2.40)$$

This interaction measure is between 0 and 2. Smaller τ indicates lower performance interaction. They stated that the difference between their method and DIM is that the latter depends on the column vectors of **U** and **V** whereas this measure depends on the row vectors for the same matrices.

It should be noted that the PIM operates on a basis of a selected pairing, therefore, to select the least interactive structure, all alternatives need to be tested.

2.5.4. μ Interaction Measure (μIM)

The purpose of this interaction measure introduced by Grosdidier and Morari (1986; 1987) is to quantify the performance degradation caused by using a decentralised control structure (multi SISO), it measures the interaction in the system caused by ignoring the off-diagonal elements in the matrix **G**. For a plant’s transfer function matrix **G**(s), the plant can be approximated by the matrix $\mathbf{G_d}$(s), where

$$\mathbf{G_d}(s) = \text{diag}\ (g_{11}(s), g_{22}(s), ..., g_{mm}(s)) \qquad (2.41)$$

This interaction measure quantifies the difference between a plant **G**(s), and its approximation $\mathbf{G_d}(s)$, and guarantees the stability of the full closed loop transfer function matrix **H**(s) defined as

$$\mathbf{H}(s) = \mathbf{G}(s)\ \mathbf{K_d}(s)\ (\mathbf{I} + \mathbf{G}(s)\ \mathbf{K_d}(s))^{-1} \tag{2.42}$$

where $\mathbf{K_d}$ is the diagonal control matrix, by applying a bound on the magnitude of the decentralised closed matrix $\mathbf{H_d}(s)$ defined as

$$\mathbf{H_d}(s) = \mathbf{G_d}(s)\ \mathbf{K_d}(s)\ (\mathbf{I} + \mathbf{G_d}(s)\ \mathbf{K_d}(s))^{-1} \tag{2.43}$$

such as

$$\sigma_{\max}(\mathbf{H_d}(s)) < \mu^{-1}(\mathbf{E1}(s)) \tag{2.44}$$

The matrices **E1**(s) and **E2**(s) are known as the relative error matrices and are defined as

$$\mathbf{E1}(s) = (\mathbf{G}(s) - \mathbf{G_d}(s))\ \mathbf{G_d}^{-1}(s) \tag{2.45}$$

$$\mathbf{E2}(s) = (\mathbf{G}(s) - \mathbf{G_d}(s))\ \mathbf{G}^{-1}(s) \tag{2.46}$$

The μIM is implemented by examining the value of μ(**E1**) such that if the system **G** is stable, then the plant will be decentralised integral controllable (Morari and Zafiriou, 1989), i.e. there exists a stabilising decentralised controller with integral action such that each individual loop may be tuned independently by a factor of 0-1 without introducing instability (Hovd and Skogetad, 1992), if

$$\mu(\mathbf{E1}) < 1 \tag{2.47}$$

As shown earlier in Section 2.2.6, the exact calculation of μ is not required for engineering purposes, instead upper and lower bounds are used. Grosdidier and Morari (1986; 1987) defined an upper bound for the optimal interaction μIM. This upper bound is the spectral radius ρ of **E1**(s), which equals the magnitude of the largest eigenvalue of |**E1**(s)|, which is the matrix **E1** with all its values replaced by their magnitudes. This bound is found to be within 1-2% from the exact value of μ (Hovd *et al.* 1993; 1994). The simple implementation of μIM ($\mu^{-1}(\mathbf{E1})$)is by plotting ρ(|**E1**(s)|) against log frequency. Then only if $\mu(\mathbf{E1}) < 1$ we are assured that the decentralised controller can incorporate integral actions.

Morari and Zafiriou (1989) defined two lower bounds for μIM depending on the frequency as follows:

$$\mu^{-1}(\mathbf{E1}) > \sigma_{\max}(\mathbf{H_d}) \qquad \text{at high frequencies} \qquad (2.48a)$$

$$\mu^{-1}(\mathbf{E2}) > \sigma_{\max}(\mathbf{E2}) \qquad \text{at low frequencies} \qquad (2.48b)$$

These bounds, however, require the full information regarding the controller structure and design. As a result, the upper bound is usually used in applying this measure.

It should be noted that the μIM operates on a basis of a selected pairing, therefore, to select the least interactive structure, all alternatives need to be tested.

2.6. METHODS FOR EVALUATING CLOSED LOOP PERFORMANCE

The assessment of the closed loop control systems is the last major step in control design. Besides the verification of the design through running the true simulations, several analytical methods were developed to assess the stability and the robustness of the control design. A brief overview of these methods is given.

2.6.1. Stability

In a multivariable control system, the stability of the entire system is a major concern, as the stability of each individual loop does not guarantee overall stability.

It is a well-known fact that the location of the roots of any system's characteristic equation determines the stability, damping and speed of the response of the system. Roots located in the right half s-plane indicate instability. Figure 2.1 illustrates the relationship between pole location and the transient response shape.

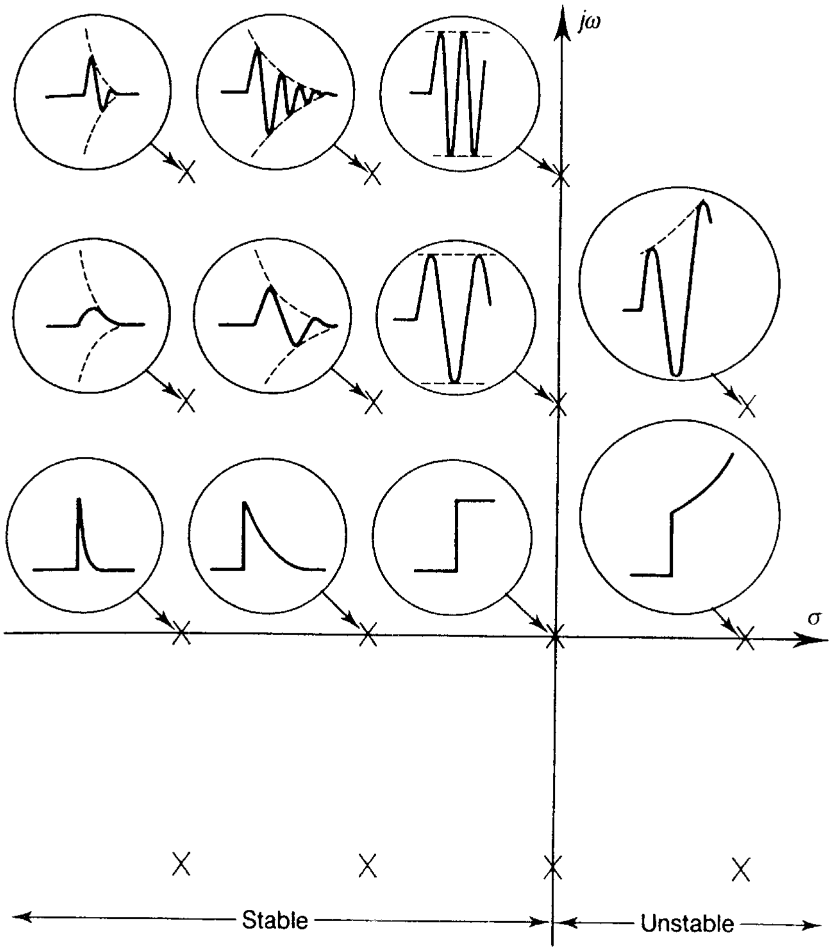

Figure 2.1: Relationship between pole position and transient response (Golten and Verwer, 1991)

The characteristic equation for an open-loop multivariable system **G**(s) can be constructed by setting the denominator of its transfer functions to zero. For a closed-loop system, with a controller transfer function matrix **K,** the characteristic equation is given by

$$\text{Det}\,[\mathbf{I} + \mathbf{G}(\text{s})\,\mathbf{K}(\text{s})] = 0 \tag{2.49}$$

This equation applies to both multi SISO and full multivariable systems. If any of the roots of this equation is located in the right half s-plane, the control system is unstable.

Sinha (1984) discussed the application of several stability criteria and their extension into multivariable systems. A simple method usually used is the extension of Nyquist's stability criterion. In this method, the frequency response of Det [**I** + **G**(s) **K**(s)] is plotted for frequency ranging from 0 to ∞, and if it encircles the origin the system is closed loop unstable. In practice Det [**G**(s) **K**(s)] is plotted, and the encirculation of the point (-1,0) is observed. However, the curves can be quite complex especially with high order systems, which makes it difficult to visualise if the point (-1,0) is encircled (Luyben, 1990).

Another method was introduced by McFarlane and Belletrutti (1973) known as the Characteristic Loci Plots. A brief justification of it is found also in Luyben (1990). For a multivariable system **G** with a controller **K** with specific structure and tuning, the method is simply to plot the eigenvalues of the matrix **G**(s)**K**(s) at changing frequency from 0 to ∞, and if any of the curves encircle the (-1,0) point, the closed-loop system is unstable.

2.6.2. Robustness

Robustness is the ability of a closed-loop system to remain stable in the presence of model/plant mismatch (Grosdidier *et al.*, 1985). Luyben (1990) stated that a control system is robust if it is tolerant to changes in process parameters. Several criteria were developed to assess robustness using mainly SVD or μ-based methods (Doyle and Stein, 1981; Skogestad and Morari, 1987; Khambanonda and Palazoglu, 1989; 1990; Koung and MacGregor, 1992; Lundström *et al.*, 1993). This subject is not covered in this thesis.

2.6.3. Dynamic Simulation

Simulation is a powerful method for qualitative and quantitative analysis of process operating performance and is a useful tool for its design. In control engineering, simulation has been used for many years to analyse and design control loops. However, the usage was limited, for example, a typical model of a single distillation column could take six months to develop (Longwell, 1993). In recent years, the usage of simulation studies has grown rapidly as a result of the accelerating development in computer technology, and the growing number of its users which means that not only “experts” can implement it. It can also play an important role in plant operations (e.g. start-up) and safety analysis as assessment can be made by experimenting with the dynamic simulation model at the design stage rather than with the real plant.

A few years ago, developing a simulation model would require writing the detailed computer code using standard programming languages such as FORTRAN or C++. This has been changed as different companies developed tools that are either complete simulation packages, or high level computer languages that facilitate the simulation development (Fell, 1997; 1998; Glassocock and Hale, 1994; Basta, 1995).

Comprehensive packages for steady state simulation were developed for many years. They expanded in the nineties to include dynamic simulation. These packages now use graphical interfaces, which makes them user-friendly, and expanded rigorous thermodynamic data, which improve their accuracy. The most important packages are ASPEN PLUS / SPEEDUP from Aspen Technology (1998), PRO-II from Simulation Sciences (1998), and HYSIM / HYSIS from Hyprotech (1998).

Despite their success, the above packages have several limitations in investigating non-standard problems (e.g. some catalytic reactors). In such cases, high level languages have been widely accepted as a tool in developing simulation models. Their usage requires the user to develop his own code. However, several powerful features supplied by the packages, such as integration algorithms, make the task easier. The most important packages available are MATLAB / SIMULINK from Mathworks (1998), MATHEMATICA from Wolfram Research (1998), and ACSL from MGA (1998).

In general, simulation work consists of several steps: firstly gathering the data, then building the model, translation into computer code, debugging and running the code, and finally analysing the results (Zeghal *et al.*, 1991).

There are two basic methods used to solve dynamic problems (Griffiths, 1992; Juslin *et al.*, 1991): Sequential modular methods in which each unit operation or a part of an operation is modelled as one block, then the whole plant model is solved sequentially one block after another, and the iterations continue until the needed accuracy is reached. Equation methods in which the whole plant model is described by a set of differential and algebraic equations which are solved simultaneously. For complex models this method looks better because it can use the vector processing capabilities of modern computer hardware to speed calculations.

As a result of the increased capability and lower cost of digital computers, more accurate mathematical models can be used to provide economical solutions. Thus extremely complicated mathematical models which were out of reach can now be solved easily. However, this does not mean that a rigorous solution of a model is the best approach to all problems. Judgement should be applied so that a solution can be reached without using much unnecessary effort which can add a little to the accuracy of the results. Though, wherever possible rigorous calculation methods should be used to calculate thermodynamic properties in order to achieve a realistic representation of the process (Griffiths, 1992).

The efficiency of dynamic models depends on the integration algorithms. Large scale dynamic problems need a powerful integration method which can handle stiff ordinary differential equations (Tyréus, 1997). Verification of simulation models can be done either by comparing the results with existing equipment, or for design purposes using pilot plant data (if the circumstances require it). Uncertain physical parameters and correlations can be measured and corrected. Then the simulation model can be scaled up to represent a best estimate model of a full scale plant. Finally when the full scale plant has been taken into use, the design model can be validated, updated if needed, and used for optimisation studies of the operation of the plant and operator training (Juslin *et al.*, 1991; Kronberger *et al.*, 1991, Vasek *et al.*, 1991).

Among the problems which have to be overcome when building dynamic models is the precise definition of the initial conditions for integration. This is one of the most serious problems which must be taken always into account since it can cause the failure of the whole model if it was defined inaccurately (Gerstle *et al.*, 1984; Asmar, 1995).

2.7. SIMULATION AND CONTROL OF REFRIGERATION SYSTEMS

In this section, a brief review concerning the dynamic simulation of refrigeration systems the application of control systems on such simulation models is included.

2.7.1. Dynamic Simulation of Refrigeration Systems

Unlike steady state simulation, only a few models (14 references) describing the transient behaviour of refrigeration systems were found in the literature. The papers found deal with relatively small-scale single stage refrigeration systems operating with a single refrigerant. Most approached the problem in a sequential method where submodels were used to simulate each component of the system, and then all were combined to form the dynamic problem.

With the exception of Goldfarb and Oldham (1996) who used a commercially available dynamic simulator, all the papers developed their own mathematical models and corresponding computer programs to solve them. FORTRAN was the language used to write the programs. The integration method used in solving the differential equations was discussed by some authors (Dhar and Soedel, 1979b; James, 1984-85). They both found that using a fourth order Runge-Kutta method does not yield an appreciable reduction in computational time as compared to the Euler method for the same desired accuracy. However, Vargas and Praise (1995) used Runge-Kutta fifth order method to achieve better accuracy in their model.

The first dynamic model cited was developed in 1965 by Hasegawa (James 1984-

85). This described an air conditioning system in a railway passenger car and was run on an analogue computer. Marshall and James (1975) then presented a mathematical model that describes an industrial quick freezing plant using ammonia as the refrigerant. Later, James (1984-85) reviewed several models describing refrigeration and air conditioning systems, which he classified as either detailed models of specific refrigeration plants, and simpler generic models for larger scale processes.

Dhar and Soedel (1979a; 1979b) formulated a general mathematical model for a refrigeration system consisting of a compressor, a thermostatic expansion valve, an evaporator, a condenser and an accumulator. They modelled each unit separately and then combined them to study the dynamic behaviour of the overall system. The program they developed was then applied to several cases to validate it. However, as indicated in the discussion of their paper, their model cannot be applied to centrifugal compressors.

Several other models for the simulation of heat pumps and refrigeration took roughly the same approach, i.e. developing submodels for each component and then combining them (Chi and Didion, 1982; MacArthur, 1984; Murphy and Goldschmidt, 1985; Chen and Lin, 1991; Vargas and Praise, 1995). The models differ by using different expansion devices (thermostatic expansion valve or capillary tube) and by including or excluding the accumulator. The model developed by Murphy and Goldschmidt (1985) did not include a submodel for the evaporator. Instead experimental data were used to replace it, thus restricting its usability.

Sami *et al.* (1987) developed a simulation model for a heat pump based on the use of a lumped parameter approach. It consisted of five submodels describing the evaporator, condenser, compressor, accumulator and expansion device (thermostatic expansion valve and capillary tube). They initially used this model to examine the dynamic performance of traditional pure refrigerants. In a subsequent study they used the model to examine the use of alternative pure refrigerants (Sami and Duong, 1991). Later this model was extended to investigate the usage of a non-azeotropic refrigerant mixture (Sami and Comeau, 1992).

Wilson and Jones (1994) developed a model describing a two-stage side-load refrigeration unit with a single component working fluid. They examined the effect of equipment sizing on the performance of the system. Their work was the background for the investigation reported in this thesis.

In a paper designed to illustrate the usefulness of a commercial dynamic simulation package PROTISS, Goldfarb and Oldham (1996) used the package to simulate a multi-stage refrigeration loop, and to study several control configurations. The process they simulated was derived from an industrial case study in natural gas processing. Good agreement between model and plant data was achieved.

2.7.2. Control of Refrigeration Systems

The main advantage of developing dynamic simulation models is that it enables the testing of several control strategies before the plant is commissioned. This approach should produce a better designed and more controllable plant. Thus, control investigation is an essential and complementary part in all dynamic simulation studies. Some of the researchers mentioned above included a control study as a complementary part in their research.

The published work on the control of refrigeration systems is very limited (9 references were cited) and, with the exception of Wilson and Jones (1994) and Goldfarb and Oldham (1996), discussed the control of single stage refrigeration. Earlier references explored the concept of capacity control, which means the compressor's capacity is matched to the changing flowrate, pressure and temperature requirements, i.e. the capacity of the compressor is changed by varying the number of cylinders operating in response to a signal obtained from the temperature of the liquid refrigerant supplied to the evaporator.

In 1975, Marshall and James (1975) showed that using capacity control methods instead of traditional on/off control improves the refrigeration system performance. Later., Wong *et al.* (1987) and Wong and James (1988; 1989) conducted a review on capacity control where they found that significant energy savings are reported when using variable-speed control to accommodate part load. They also compared their model with experimental results. With this model they studied six types of control including on/off control and variable speed compressor. The results of their investigations favoured using a variable speed compressor.

Balchen *et al.* (1989) investigated adaptive control of a single control loop system. They developed an algorithm to estimate the frequency response parameters, and tested on a laboratory scale refrigeration cycle.

Heyen *et al.* (1994) developed an object-oriented, lumped parameter, dynamic simulation model for process compression, expansion and heat transfer. Control design procedures are also included in the program which uses MATLAB. The control model enables designing PI block diagonal control systems.

Wilson and Jones (1994) presented initial results for a study on controlling a multi-stage refrigeration system. The investigation was performed using ACSL, and the effect of the equipment size on control system performance was studied.

Vargas and Praise (1995) presented a control system for a heat pump system. They proposed a novel control approach based on power consumption in the process to keep the compressor at maximum speed until its temperature reaches a value very close to the desired value, then the speed is gradually reduced. They showed that substantial energy savings can be achieved.

Goldfarb and Oldham (1996) used PROTISS to simulate a multi-stage refrigeration process. They used the controller units available in the package to design PI controllers. Their objective was to control the process gas temperature exiting the chiller, and also to protect the compressor against surge.

2.8. DISCUSSION

In recent years, research on the control of multivariable processes has increased dramatically. The advances in computer technology enhanced the development process, and allowed for more complex solutions to be considered. This initiated a new awareness of the interaction between design and control and attempts to incorporate the control design into the process design at an early stage.

Research on the control of multivariable processes can be divided into four major categories paralleling the development of a control design:

1. Selection of sets of variables for the control problem
2. Methods to guide on pairings selection
3. Interaction analysis of the selected fixed pairings
4. Analytical or simulation evaluation of the closed loop control systems.

Several methods and measures were proposed to cover each of the categories above. Most researchers were interested in the mathematical rather than the physical interpretation of the measures recommended. Little research has been conducted applying the measures to higher multivariable systems, and comparing the outcomes of several measures. In this thesis, work is performed on the usefulness of several measures to a high order multivariable system, namely the refrigeration system.

It has also been noted that despite the rapid development in dynamic simulation, its implementation in refrigeration systems is still limited, with the majority concentrating on the single stage systems. Research on the application of control systems on the refrigeration systems is even rarer with only a handful of cited references. This lack of application was amongst the reasons that encouraged conducting the current research.