The Application of Parallel Computer Technology to the Dynamic Analysis of Suspension Bridges

A Thesis Submitted to

The University of Glasgow

By

Jason Gordon Beith

B.Eng. M.Sc. D.I.C.

For

The Degree of Doctor of Philosophy

In

The Faculty of Engineering

UNIVERSITY
of
GLASGOW

To Linda and Family

ACKNOWLEDGEMENTS

I would like to express my sincere thanks to my supervisor Dr Alan Agar for his invaluable advice, encouragement and help throughout the course of this study.

I am grateful for the financial support extended by the Engineering and Physical Science Research Council (EPSRC).

I would like to express my thanks to my friends and colleagues, both at The University of Glasgow and at Sir William Halcrow & Partners, for their support and encouragement.

Finally, I would like to acknowledge the support and encouragement of my wife and family, without which this thesis may never have been completed.

ABSTRACT

This research is concerned with the application of distributed computer technology to the solution of non-linear structural dynamic problems, in particular the onset of aerodynamic instabilities in long span suspension bridge structures, such as flutter which is a catastrophic aeroelastic phenomena.

The thesis is set out in two distinct parts:-

Part I, presents the theoretical background of the main forms of aerodynamic instabilities, presenting in detail the main solution techniques used to solve the flutter problem. The previously written analysis package ANSUSP is presented which has been specifically developed to predict numerically the onset of flutter instability. The various solution techniques which were employed to predict the onset of flutter for the Severn Bridge are discussed. All the results presented in Part I were obtained using a 486DX2 66 MHz serial personal computer.

Part II, examines the main solution techniques in detail and goes on to apply them to a large distributed supercomputer, which allows the solution of the problem to be achieved considerably faster than is possible using the serial computer system. The solutions presented in Part II are presented as Performance Indices (PI) which quote the ratio of time to performing a specific calculation using a serial algorithm compared to a parallel algorithm running on the same computer system.

Table of Contents

Notation

Symbols used in the text are defined wherever they appear. But for convenience, a summary is presented below:-

Symbol

B	Width of deck section (B=2b).
b	Semi-chord or half width of deck section.
[C]	Diagonal matrix of the system.
C_D	Drag force per unit length /($\rho V^2 D/2$).
C_F	Averaged aerodynamic force per unit length / ($\rho V^2 D/2$).
C_L	Lift force per unit length / ($\rho V^2 D/2$).
c_a	Non-dimensional aerodynamic damping coefficient ($2m\delta_a / \rho D^2$).
c_s	Non-dimensional structural damping coefficient ($2m\delta_s / \rho D^2$).
D	Across wind dimension of the body.
F(k), G(k)	Real and Imaginary parts, respectively, of Theodorsen Circulation function C(k);
h	Vertical displacement.
$\dot{h}$	Vertical velocity.
$\tilde{H}_i^*$, $\tilde{A}_i^*$	Non-dimensional aerodynamic stability coefficients associated respectively, with bending and torsion; mass moment of inertia per unit span;
J_i, Y_i	Bessel functions of first and second order;
K	Reduced frequency in Scanlans notation (K= BN/V, K= k/π);
K_h	Vertical stiffness of deck section.
K_α	Torsional stiffness of deck section.
[K]	Stiffness matrix of the system.
k	Reduced frequency (k= bω/V).
k_a	Non dimensional aerodynamic stiffness ($K_a / \rho N^2 D^2$).
k_s	Non dimensional aerodynamic stiffness ($K_s / \rho N^2 D^2$).

L_h	Aerodynamic lift on deck section.
M_α	Aerodynamic moment on deck section;
[M]	Mass matrix of the system.
m	Mass of bridge (incl. cables) per unit length (kg/m).
N	Oscillation frequency in cycles per second ($N=\omega/2\pi$).
N_h, N_α	Natural frequencies associated with vertical and torsional motions respectively.
n	Frequency of vortex shedding.
{P}	Vector of applied loads.
Rv	Reduced velocity ($Rv= 1/k = V/\omega b$).
r	Radius of gyration ($I = mr^2$).
S	Strouhal number ($S= nD/V$).
s	Span of the bridge (m).
T	Period of oscillation.
t	Time.
{U}	vector of displacement of structure at its nodes.
V	Wind speed.
V_{cr}	Critical wind speed.
α	Torsional displacement.
$\dot{\alpha}$	Torsional velocity.
ρ	Density of air.
ω	Flutter frequency in radians/second.
ω_h, ω_α	Natural frequencies in vertical and torsional motion, respectively.
ω_α / ω_h	Natural frequency ratio (N_α / N_h).
δ_a, δ_s	Aerodynamic and structural logarithmic decrement (damping) respectively.

Subscripts

a	Associated with aerodynamic terms.
h	Associated with degree of freedom along vertical direction.
s	Associated with structural terms.
α	Associated with degree of freedom along torsional direction.

PART I

Civil Engineering Aerodynamics

1 INTRODUCTION

1.1 Historic Review of Suspension Bridges

The origins of suspension bridges derive from the need to bridge natural features. Some of the most spectacular primitive suspension bridges were built in the highlands of Peru where turbulent rivers running through deep canyons made travel and communication impossible for Incan tribes without the help of bridges. One such bridge is the 50m long Keswachaca Bridge, spanning 36m above the Apurimac river, at an altitude of 4800m above sea level, Figure 1-1.

Figure 1-1 The Keswachaca Bridge.

Considerable increases in the length of suspension bridges were heralded when natural ropes were replaced by wrought iron chains, for example the 21m long bridge over the river Tees near Middleton in 1741, the 30m long Lahn bridge, Germany in 1785 and the 21m Long Uniontown bridge in Pennsylvania in 1796.

Advances in theoretical mechanics in the 19th century such as the works of Clericetti and Melan on elastic and deflection theory, Timoshenko's work on energy methods and Castigliano's strain energy method for arches (a suspension bridge is basically an inverted arch bridge) led to the explanation in mathematical terms of how a suspension bridge actually worked.

In 1819 Thomas Telford started to build the Menai Straits bridge, spanning a distance of 177m, far greater than had ever been bridged before in a single span.

In 1849 Charles Ellet completed what was the longest spanning suspension bridge in the world, the 308m long span Wheeling bridge, West Virginia. However, this collapsed a few years later during a windstorm.

Around the same period John Roebling was building the 235m long Niagara Falls bridge. The bridge was a two-level structure joined by a timber truss, with trains crossing on the top level and vehicles below. Roebling learned from the disaster in West Virginia and recognised the importance of stiffening the response of the bridge. This was achieved using trusses and wire stays to minimise swaying and twisting. The bridge remained in service for forty-two years, by which time greatly increased loading had rendered it obsolete. In 1867 Roebling started to build The Brooklyn Bridge which was finished by his son, Washington Roebling fourteen years later. Similar to the Niagara bridge, it had steel trusses and stays radiating from the towers to the deck which stabilised the bridge and also improved its strength..

At the beginning of the twentieth century, a period of great progress came and with it all previous records were broken. Many long span suspension bridges were built in the USA such as the George Washington bridge in New York in 1931, with a main span of 1067m which was double the length of the previous longest bridge. This was surpassed in 1937 with the construction of the Golden Gate bridge in San Francisco, with a main span of 1280 m, Figure 1-2.

Figure 1-2 The Golden Gate Bridge

The demand for more economical and lighter as well as more aesthetically pleasing bridges led to the construction of longer and more slender structures. In July 1940 the Tacoma Narrows bridge with a main span of 853m was opened to traffic across Puget Sound in Washington State. It was seventy-two times as long as it was wide and had very little torsional stiffening. Three months after it was opened, it collapsed in a transverse wind speed of 40mph (18 m/s). After the collapse of the Tacoma Narrows bridge, the stability of suspension bridges under such aeroelastic phenomena began to be extensively investigated.

The need for fast and efficient rebuilding of approximately 8500 bridges in post-war Germany called for the development of new design concepts and fabrication techniques. The box girder, originally introduced by Robert Stephenson in the 19th century, was developed into the thin walled all-welded structural member commonly used today. In contrast to the traditional truss girder, the orthotropic steel deck in a box girder serves as an integral part of the structure with a considerably larger torsional rigidity than an equivalent sized truss girder. Substantial savings were obtained in the weight of the bridge, also in construction and maintenance costs, but aerodynamically problems persisted, particularly during the erection stages when the girder lacks the final torsional stiffness, mass and continuity.

In 1966 the Severn Bridge linking England and Wales was constructed. The deck structure of this 1036m main span bridge was a 3 m deep closed box, which had been extensively tested in wind tunnels in order to study its aerodynamic stability. It behaved satisfactorily, providing low drag and reduced flow separation. Many subsequent bridges constructed around the world were designed to have similar aerodynamic characteristics as the Severn bridge, for example the Bosporous bridges in Turkey, the Lillaebelt bridge in Denmark and the Humber bridge.

The last decade has seen bridge span records being broken time and time again as engineers devise new construction methods and pioneer new designs. Since the George Washington Bridge was finished, all successive world record holders have been suspension bridges. When completed in late 1997, the Great Belt East Bridge in Denmark will enjoy a brief period as the world's longest single span at 1624m. This would shortly be surpassed by the Akashi-Kaikyo suspension bridge in Japan with a main span of 1990m. However, if the economic and political problems surrounding the £2500M proposed Messina Strait bridge, planned to link the Italian mainland to Sicily, can be resolved, this would increase the world's longest suspension bridge free span to 3300m. An artist's impression of the bridge is shown in Figure 1-3.

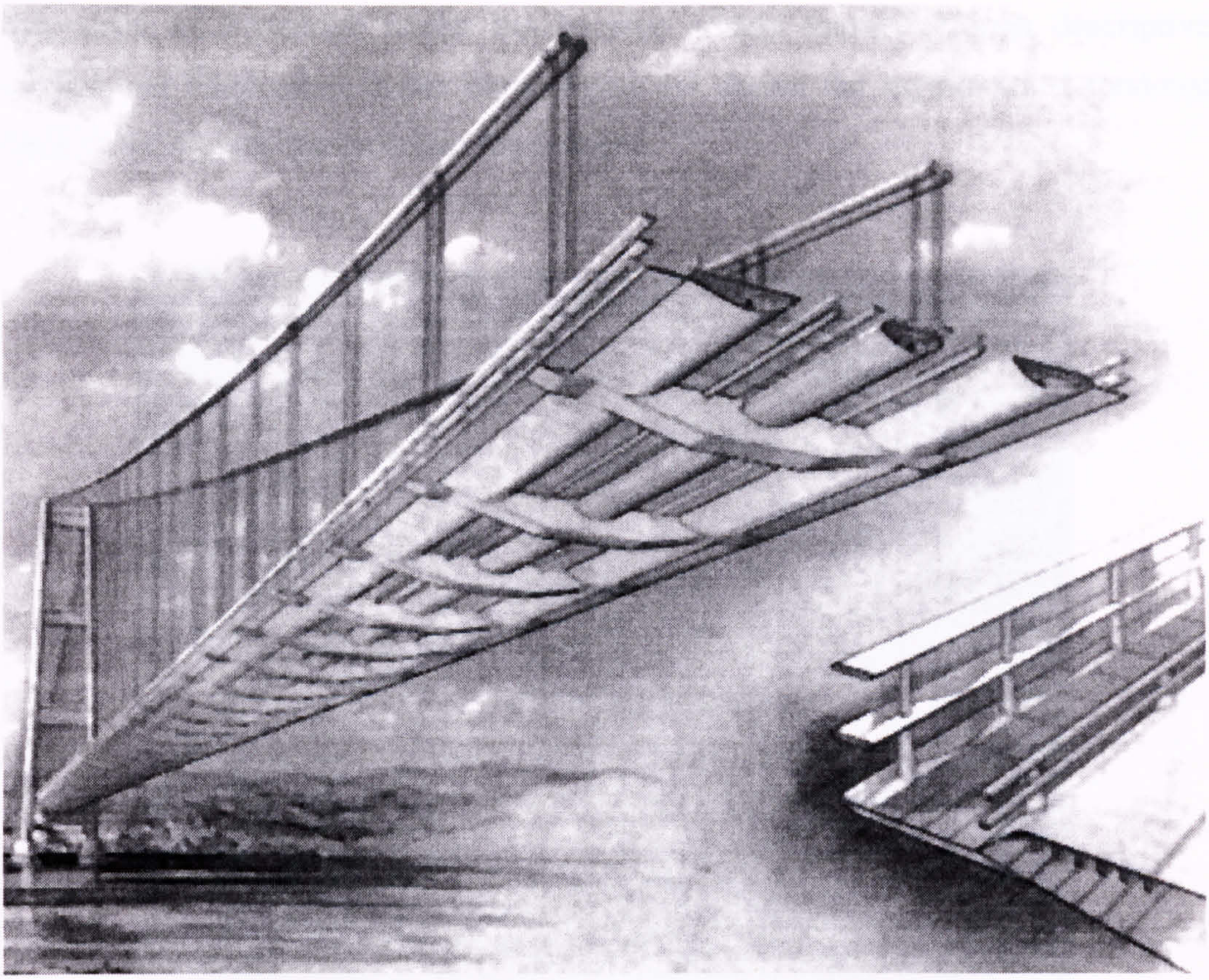


Figure 1-3 The proposed Messina Bridge.

1.2 Historic Review of Bridge Aerodynamics

The action of wind has been a common cause of failure of bridges. Many failures have been due to the inability of the bridges to withstand the steady wind forces, and the have been "blown over". Other failures have been caused by destructive oscillations set up by wind, these oscillations resulting from one of the main forms of aerodynamic instability that will be discussed in Section 2.2. The wind speeds capable of exciting such oscillations are approximately related to the values of the natural frequencies of the bridge: hence bridges with low natural frequencies, such as slender suspension or cable stayed bridges, would be expected to be the most vulnerable.

The attention that engineers now devote to the aeroelastic effects of wind on bridges only dates from 1940, when the original Tacoma Narrows bridge, oscillated to

destruction in a manner which has become widely known through descriptive account and films taken at the time. Figure 1-4 shows the asymmetrical torsional mode of oscillation that preceded the collapse.

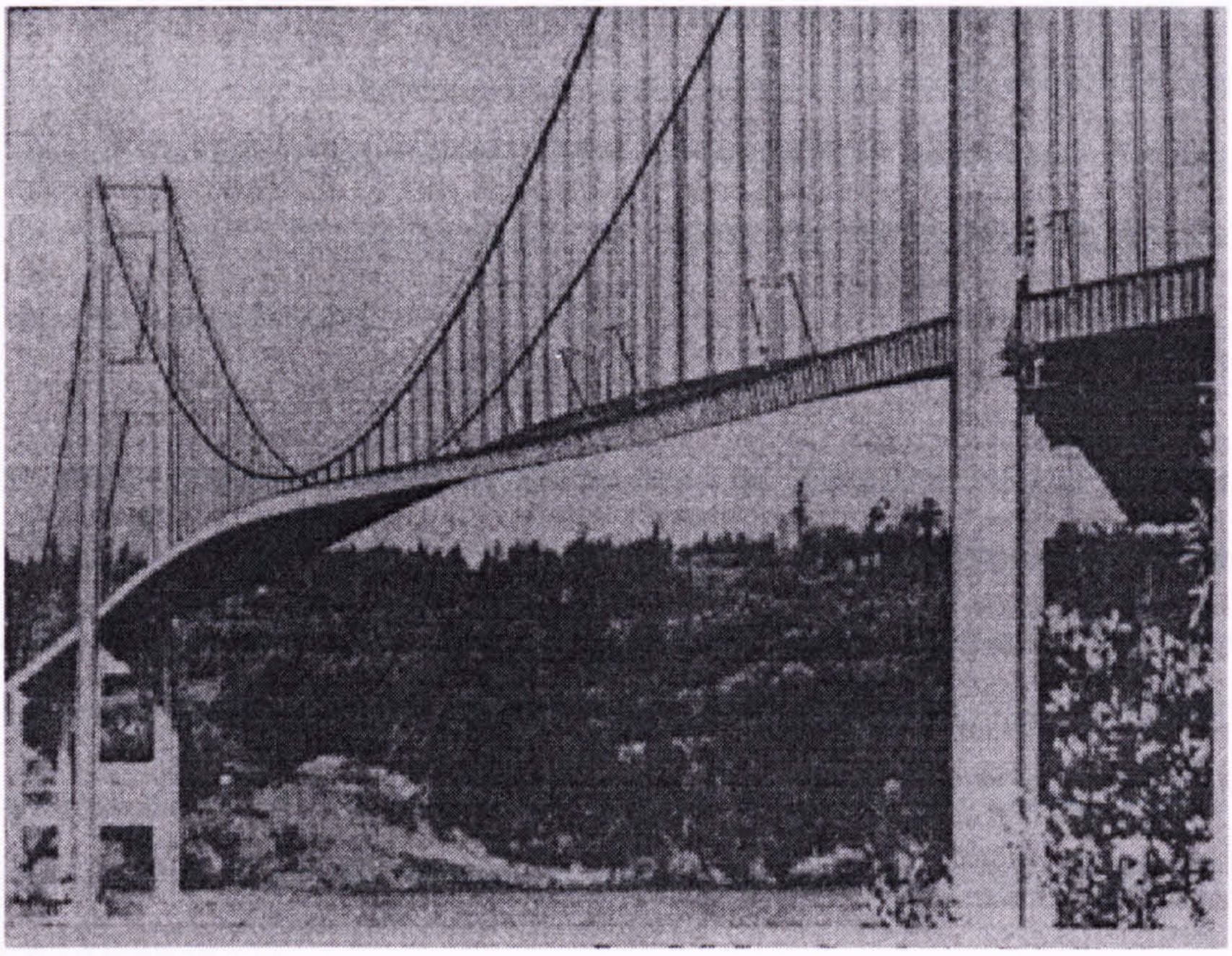


Figure 1-4 Tacoma Narrows Bridge failure of 1940.

This bridge with stiffened plate girders deck with a main span of 853m had been prone to flexural oscillations, both during its construction and during its brief four month life. These oscillations occurred at a variety of different frequencies depending on the prevailing wind speed, with double amplitudes of oscillation at times reaching 1.5m.

On the morning of the collapse, the flexural oscillation, at a wind speed of 18 m/s, turned to the violent torsional oscillation that ultimately led to the catastrophic destruction of the bridge within one hour.

Although the failure of the Tacoma Narrows bridge forced engineers to take aerodynamic stability into account in the future design of long-span bridges and prompted a number of scientific investigations into the problem, it was not the first bridge to be destroyed or damaged due to severe wind effects. Several short-span

suspension bridges built in the British Isles during the nineteenth century had suffered in a similar way.

The Menai Straits bridge was damaged due to torsional oscillations in 1826, 1836 and 1839. In 1836 a span of a chain pier at Brighton was broken during a storm by torsional oscillations with a node at mid-span. The eyewitness accounts indicate that the mode of failure was precisely the same as that which destroyed the first Tacoma Narrows bridge over 100 years later. In 1854 the Wheeling suspension bridge collapsed during a windstorm.

In 1967 the plate girder stiffened roadway deck of the cable stayed bridge at Long's Creek in Canada, oscillated in a non-destructive flexural bending mode in wind speeds within the range of 11-35 m/s, as did the cable stayed bridge of somewhat similar deck configuration at Onomichi, Japan. The box girder bridge over the River Wye completed in 1966 has exhibited vertical bending oscillations for a range of wind speeds around 7-8 m/s. However, they are low amplitude oscillations of no practical significance [1].

Bridges are particularly susceptible to wind-induced oscillations during the erection stages when their full torsional stiffness has not yet been completely developed, also bridge components such as the towers and deck units may present aerodynamic stability problems quite distinct from those of the completed deck. An overview of various forms of aerodynamic instabilities is presented in Section 2.1.

1.3 Modern suspension bridge design

Most of the developments in suspension bridges have been brought about due to the design and construction of new bridges. One of the more difficult decisions comes when choosing the deck section at the initial design phase.

Historically, most long span cable supported bridges have been built with truss girders in order to facilitate fabrication and erection, whereas little attention was paid to maintenance and aerodynamic performance.

Trusses can be designed to exhibit sufficient torsional stiffness to safeguard the bridge against torsional flutter instability by introducing horizontal top and bottom wind bracing and adopting a truss depth of 1:170 - 1:120 of the span length. The flutter resistance can be further enhanced by longitudinal open slots in the road deck, a well known feature from post World War II suspension bridges in North America and Japan [2,3]. The high lateral wind loads for truss girders compared to streamlined box girders are usually only of relatively minor importance for medium span classical suspension bridges. However, truss girders are commonly found to be 15% - 20% heavier than a box girders designed for similar loading.

The aerodynamic profile of box girders reduce the lateral wind loading in comparison with the truss girder, while maintaining the structural stiffness in torsion. Vortex shedding, a common problem in truss girders which may not have immediate catastrophic consequences but is unacceptable for users and may cause structural fatigue and wear in joints and bearings, can be reduced in box girders to an acceptable level. This is done by "streamlining" the box section, by using aerodynamic fairings and guide vanes at the wind-ward and the lee-ward edges.

This method can also be considered as a retrofit measure , as in the case of the Long's Creek cable-stayed Bridge (Canada). However the box girder still has the problem of aerodynamic instability that may make it unsuitable for ultra long suspension bridges.

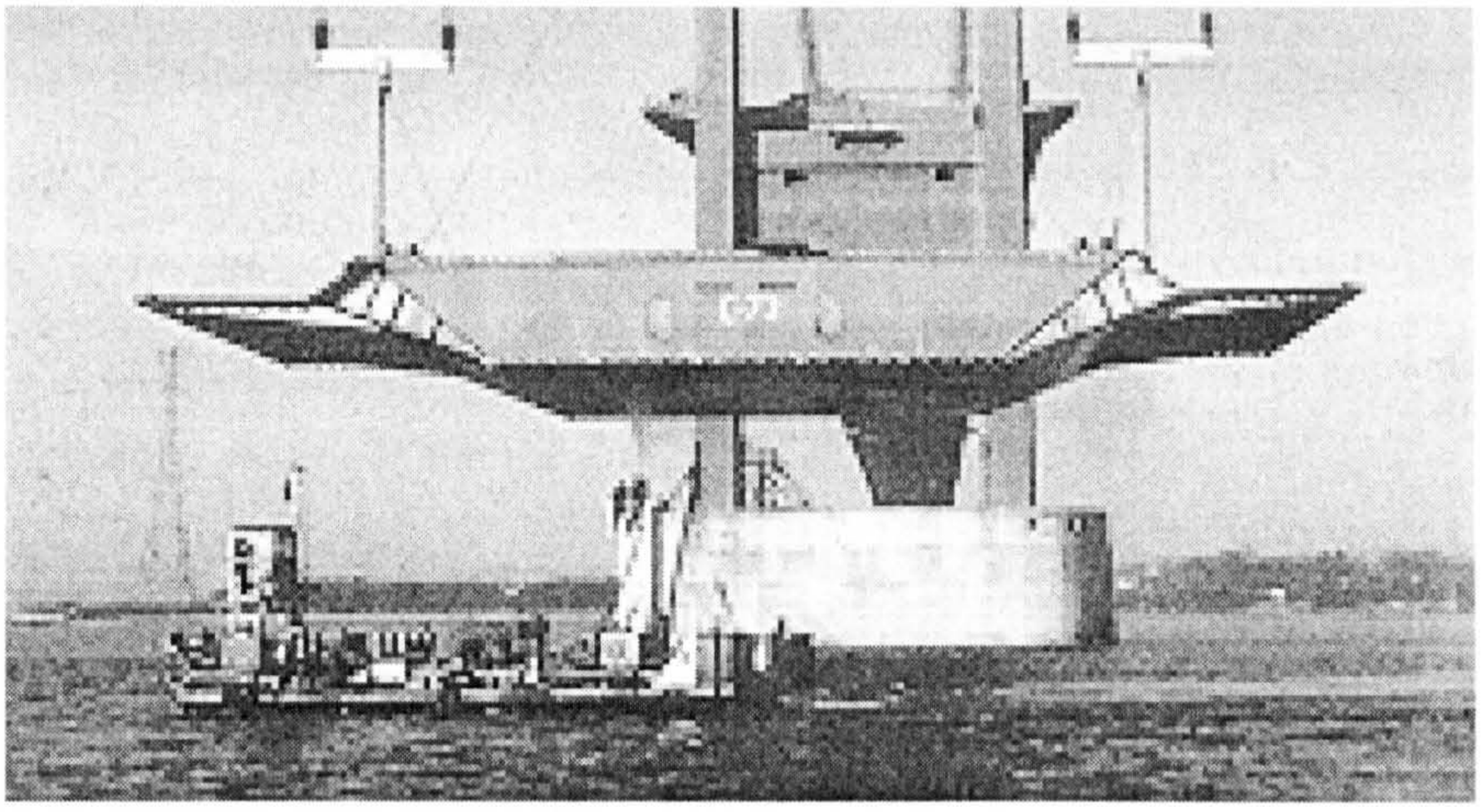


Figure 1-5 A Typical deck Section of the Severn Bridge

For the Great Belt Bridge [4] a closed box section has been adopted, which ensures an acceptably high flutter wind speed for the meteorological conditions of the area. However a study for the design of the Akashi-Kaikyo Bridge [5] showed that for spans over 1700m, only truss deck sections satisfied the Japanese code requirements for stability. The proposed 60m wide section for the Messina Strait Bridge [6] is constructed from a "multibox" section which seems to behave satisfactorily for future ultra long span bridges, Figure 1-3.

The development of analytical tools for the numerical analysis of aerodynamic performance and the advent of Finite Element Methods (some commercial available packages include formulation of Navier-Stokes equations) promise a new era in bridge aerodynamics, where designers will be able to eliminate inefficient configurations before turning to the wind tunnel for verification.

2 Fundamentals of Aerodynamics

2.1 General Aeroelastic problems associated with suspension bridges

Suspension bridges are structures continuously exposed to wind and prone to several forms of aerodynamic excitations that may result in motions in isolated vertical bending or torsional modes or in coupled vertical bending and torsional modes. Depending on the nature of the excitation the motions may be of:

i. Limited Amplitude (Non-Divergent): oscillations produced by vortex shedding and buffeting, which could cause unacceptable stresses or fatigue damage.
ii. Non-Oscillatory Divergent: The structure fails due to a constant pseudo-static wind load.
iii. Divergent Oscillatory Amplitude: oscillations produced by galloping, stall flutter and classical flutter, which must be avoided.

The consequence of each of these behaviours is different. The limited amplitude oscillations, may be considered as a serviceability problem in limit state terminology that could be responsible for serious fatigue damage in the long term. The latter classes of divergent oscillatory and non-oscillatory divergent, in particular flutter, may be considered to be ultimate conditions where the basic safety of the structure may be threatened.

Current UK code guidance [8] on stability is available for structures with individual spans not exceeding 200m. However, for spans greater than this, the regulations advise that the stability should be verified by wind-tunnel tests, which are both time consuming and expensive. Hence methods have been developed to attempt to model the behaviour numerically, refer to Section 3.

2.1.1 Limited Amplitude Response

2.1.1.1 Vortex shedding

Vortex induced oscillations of limited amplitude may be excited by the periodic cross-wind forces arising from the shedding of vortices alternatively from the upper and lower surfaces of the bridge deck. The excitation is created by a periodic

variation in the pressure on the lee-ward face and may result in a dynamic response of the structure either in torsion or more frequently in flexure. The frequency of shedding single vortices is given by 2S, where S is the Strouhal number

$$S = \frac{nD}{V}$$

Equation 2-1

D is the height of the bridge deck (Figure 2-2), V is the wind speed and n is the vortex frequeney. The onset of the oscillations arise when the frequency of vortex shedding approaches the natural frequency of the structure.

If the structure has low structural damping δ_s and the aerodynamic damping δ_a is negative (referred to as an aerodynamic excitation), the net global damping could be zero and oscillations may start and continue to increase in amplitude until they are limited to a finite value by the presence of non-linear effects such as a decrease in the value of δ_a or an increase in the amplitude (Figure 2-1). The oscillations tend to occur at a range of wind speeds that starts at a critical wind speed V_{cr} persisting as the wind speed increases to an upper limit V_{cu}, where the structure becomes stable.

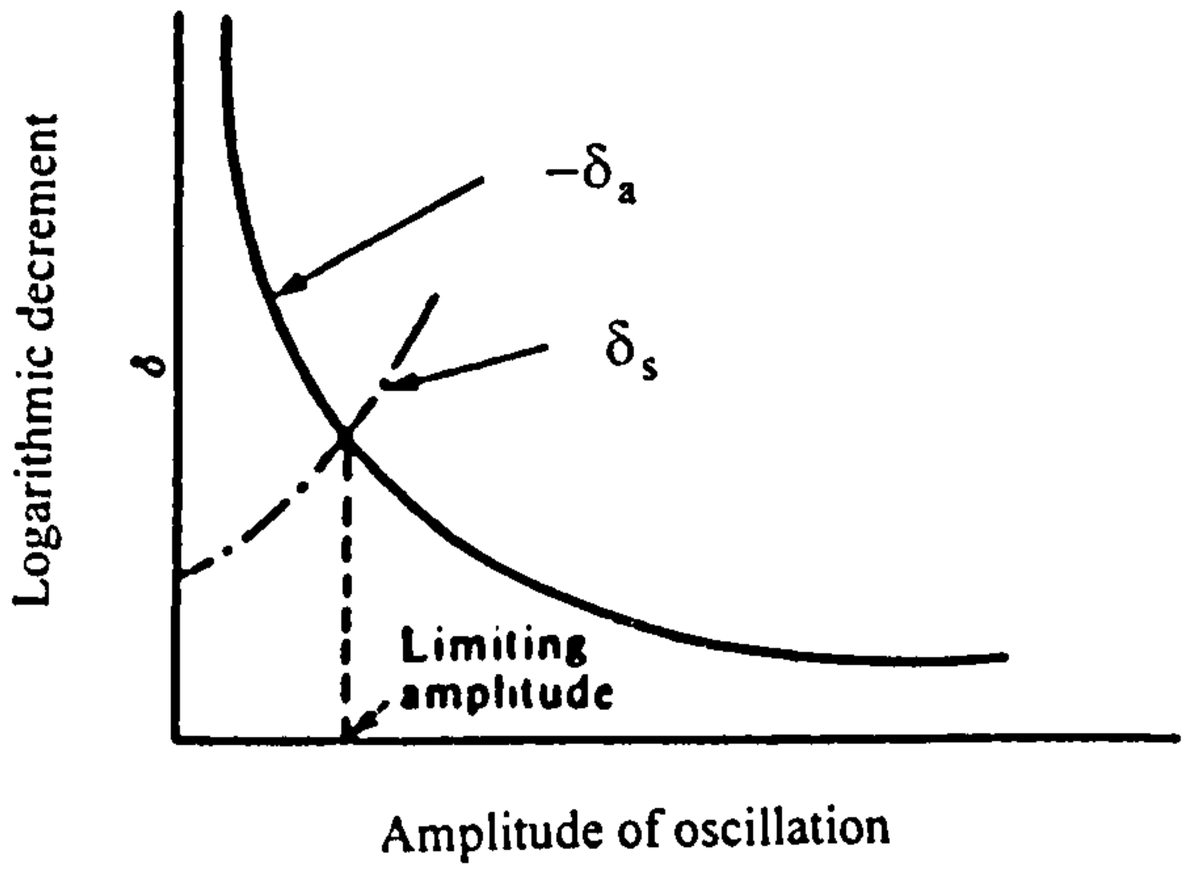


Figure 2-1 Limiting amplitude of oscillations as determined by non-linear aerodynamic excitations and structural damping

2.1.1.2 Buffeting

Oscillations of a structure may be caused by the buffeting action of the turbulent wake of an upstream obstruction, Figure 2-2. Because of its turbulent nature, forces and moments developed by wind on bridge decks fluctuate over a range of

frequencies. If sufficient energy is available at frequencies close to one or more modal frequencies, the structure may be forced to oscillate.

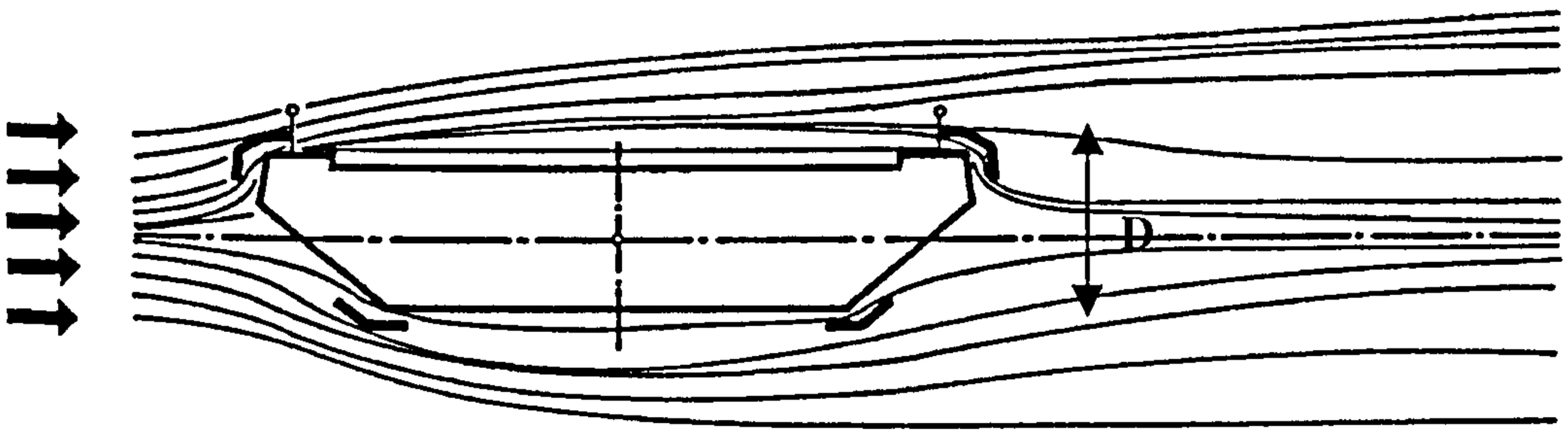


Figure 2-2 The flow pattern around a Typical Bridge Deck

2.1.2 Non-Oscillatory Divergent

2.1.2.1 Torsional Divergence

Divergence is a non-oscillatory instability. It occurs under the influence of the twisting moments induced by the transverse wind stream. This causes the bridge to twist, thus increasing the angle of attack, which in turn increases the lift and moment.

Usually the aerodynamic stiffness k_a, which is the rate of change of aerodynamic force with the angle of attack, is negligible compared to the structural stiffness k_s. However for some structural shapes, k_a may become negative at a critical wind speed. For wind speeds below this critical value the bridge deck is stable. However, beyond this critical wind speed the negative aerodynamic stiffness may become numerically equal to the torsional structural stiffness resulting in zero net torsional stiffness, causing the lift and moment to grow, ultimately resulting in the failure of the bridge by a catastrophic "flipping over".

2.1.3 Divergent Oscillatory Response

2.1.3.1 Galloping

Galloping instabilities arise on certain shapes of deck cross sections because of the characteristics of the variation of the wind drag, lift and pitching moments with the angle of incidence. Instability can arise when $\frac{dC_F}{d\alpha}$, the slope with respect to the

angle of attack α of the cross wind force F is negative. Airfoils and bodies of square cross sections are prone to galloping instability. For small amplitudes this condition can be written

$$\frac{dC_F}{d\alpha} = \frac{dC_L}{d\alpha} + C_D < 0$$

Equation 2-2

where

C_D, C_L and C_F are the non-dimensionalised drag, lift and pitch coefficients.

The velocity of motion across the wind stream induces a relative wind at incidence to the body. With the above condition the wind force is in the direction of motion, and energy is transmitted to the body from the wind.

Galloping oscillations occur when c_a is negative and numerically greater than c_s, where c_a and c_s is the non-dimensional aerodynamic and structural damping respectively. Scruton [7] has shown that the energy input per cycle can be calculated and that c_a may be determined from the equation

$$c_a = \frac{Rv^2}{4\pi^2 \eta_0} \int_0^{2p} C_F(1+\tan^2\alpha) \cdot \cos(\omega t) \cdot d(\omega t)$$

Equation 2-3

Provided that the variation of C_F is known, the integral may be evaluated either graphically or analytically using a polynomial expression for C_F. Oscillations start when $c_a + c_s = 0$. There is no upper wind speed for this type of instability.

2.1.3.2 Flutter

2.1.3.2.1 Stall flutter

This is a Single degree of freedom oscillation of airfoils in torsion, driven by the non-linear characteristics of the lift in the vicinity of the stall, or loss-of-lift condition. This has been particularly studied in relation to aircraft design, but is also a possible cause of oscillations in suspension bridges. This is associated with systems

undergoing strongly separated flows. Bluff, unstreamlined bodies are typical examples. Prominent among these are the decks of suspension bridges, which can in various instances exhibit single-degree torsional instability. As in galloping, there is no upper wind speed limit for a stalling instability.

2.1.3.2.2 Classical Flutter

Classical flutter is a self-excited oscillatory instability of a body suspended in air stream. It involves the interaction of aerodynamic, inertial and elastic structural forces such that, at a certain wind speeds, the aerodynamic forces act to feed energy into the oscillating structures and increase the magnitude of vibration, sometimes to catastrophic levels.

During a flutter oscillation vertical and torsional motions occur together in a simple harmonic oscillation at a common frequency somewhere between the natural frequency for independent vertical (heaving) and torsional (pitching) oscillation, Figure 2-3. The cross sectional shape of the body affects the aerodynamic forces that act during pitching and heaving oscillations. This form of aerodynamic instability is thought to have been responsible for the catastrophic failure of the Tacoma Narrows Bridge. The theory of classical flutter will be presented in detail in Section 2.2.

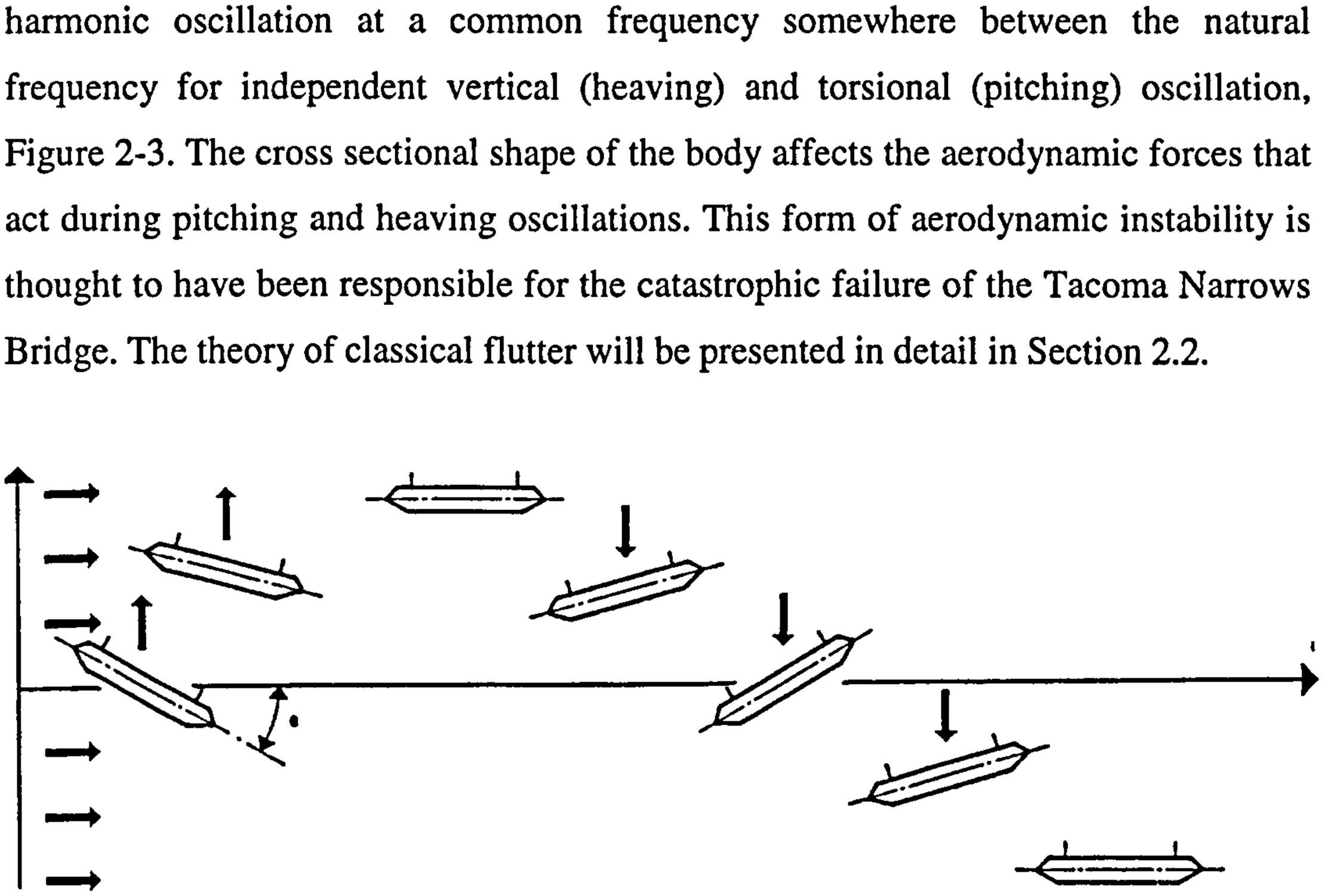


Figure 2-3 The Flutter response of a Bridge Deck.

2.2 Flutter Solution Techniques

In most cases the assumption that wind loading be treated as a constant pseudo static force is probably adequate for the design of most engineered building and structures. Bridges however are long, slender structures with low levels of structural damping, which are continuously exposed to wind loading for the duration of their serviceable life. Thus wind needs to be considered as a dynamic load.

It is assumed that unsteady aerodynamic forces acting in one plane of a structure are independent of its motion and displacements in others. The out-of-plane aerodynamic forces tending to move a structure in the lateral (cross) wind direction are not significantly influenced by movements in the longitudinal (drag) direction. This assumption has been found to be justified by experience for all the mechanisms of aerodynamic instability described, with the notable exception of classical flutter that is formed by the coupling of flexural and torsional motions in a multi-degree of freedom system.

In dynamic aeroelasticity we are mainly concerned with two distinct fundamental physical phenomena, those of galloping and flutter. Galloping, which is typified by large amplitude oscillation of bluff bodies normal to the direction of an airstream does not generally pose a serious problem. However, flutter is the main and most dramatic physical phenomenon in the field of aeroelasticity, a dynamic instability that if allowed to occur would lead to catastrophic structural failure.

2.2.1 Theoretical Methods

To discuss flutter more fully, the dynamic theoretical model from which the aerodynamic forces are obtained will be presented. The forces are developed from the fundamentals of fluid mechanics and are applied to the 'typical section' model widely used in aeronautical practice to illustrate the physical implications of dynamic aeroelasticity, Figure 2-4.

The section when placed in a constant wind stream is assumed to have an aerodynamic pressure distribution given by p(x,t), the resultant force L and M are the

aerodynamic lift and moment, respectively about the elastic axis of the section, Equation 2-4.

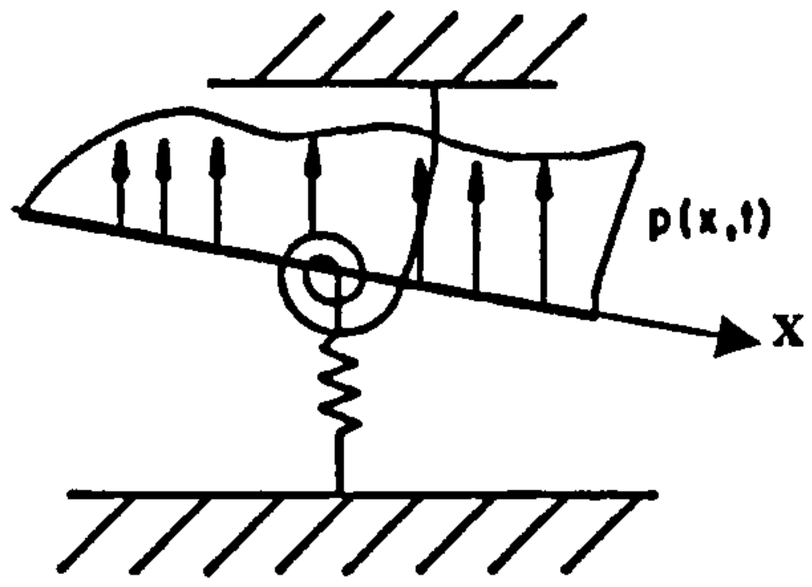


Figure 2-4 Typical Section Model. (From Dowell *et al*[9])

$$L = \int p(x,t)\,dx \qquad M_y = \int p(x,t)\cdot x\,dx$$

Equation 2-4

The equations of motion for such a system are

$$m\ddot{h} + k_h h + S_\alpha \ddot{\alpha} = -L$$

$$S_\alpha \ddot{h} + I_\alpha \ddot{\alpha} + k_\alpha \alpha = M_y$$

Equation 2-5

Where

$m = \int \rho\,dx$ Mass of the model

$S_\alpha = \int \rho \cdot x\,dx$ Additional Inertial Effect of Eccentricity of Centre of mass, (0 for symmetric sections)

$I_\alpha = \int \rho \cdot x^2\,dx$ Mass moment of Inertia of model.

The problem is essentially one of unsteady aerodynamics, which is concerned with the time-dependent fluid motion generated by bodies moving in a fluid. For most applications in aeroelasticity in which appreciable viscous and compressibility effects are absent, linearised small perturbation models of inviscid fluid flow are appropriate since the body motion is generally composed of a small time-dependent

motion combined with a steady state motion. In this case the model is appropriate, since the theory is derived using a flat plate idealisation for the airfoil, which has little or no separation at the leading edge and thus a laminar wake of vorticity is formed.

The circulation around an aerodynamic surface executing unsteady motion constantly changes as vorticity is shed into the wake, thus affecting the lift generated by the surface. The distribution of vorticity in the wake depends on the motion history since each increment in the motion sheds vorticity into the wake. In a two-dimensional inviscid fluid, however, the total circulation around any closed curve is constant, as postulated in Kelvin's Circulation Theorem.

Flutter instability has been observed in experiments on thin airfoil. As a result of the similarity in the shape and geometry of aerodynamic (i.e. faired) bridge cross-sections, there is behavioural similarity between the aerodynamic instability of suspension bridge decks and the flutter of airfoils. Thus flat plate airfoil theory is an excellent guide for the investigation of potential susceptibility of suspension bridge structures to classical flutter phenomena.

2.2.1.1 Theodorsen's Two Dimensional Unsteady Airfoil Theory

In 1935 Theodorsen [10] derived his theoretical formulae for flutter problems under incompressible flow. In classical flutter, two degrees of freedom of the structure, one rotational, the other transitional, are coupled in a self excited oscillation produced by a constant transverse wind flow.

In linear, inviscid, incompressible aerodynamic theory, the airfoil and its wake are represented by thin surfaces of vorticity (i.e. 2-D vortex sheets). Thus, for a two-dimensional airfoil undergoing unsteady motion in a uniform free stream, it is convenient to idealise the airfoil as a flat plate, Figure 2-5.

The most generally quoted solution for the transient aerodynamic response of a thin airfoil subjected to arbitrary heave and pitching motions, $h(t)$ and $\alpha(t)$, are

attributed to Theodorsen. These expressions were developed in the frequency domain and are presented below in aeronautical notation and terminology, Equation 2-7.

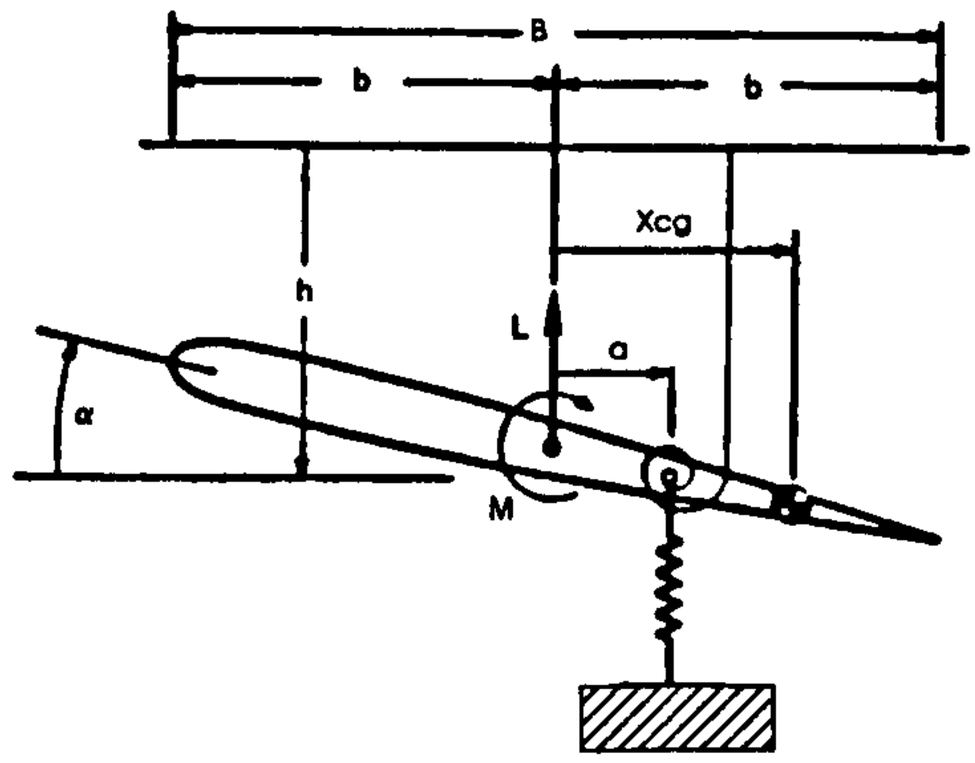


Figure 2-5 Flat Plate Airfoil Model (From Dowell *et al*[9])

$$\overline{L}(\omega) = 2\pi\rho VbC(k)\left(V\overline{\alpha} + i\omega\overline{h} + i\omega\overline{\alpha}b\left(\tfrac{1}{2} - a\right)\right)$$
$$+\rho\pi b^2\left(Vi\omega\overline{\alpha} - \omega^2\overline{h} + \omega^2\overline{\alpha}ab\right)$$

Equation 2-6

$$\overline{M}(\omega) = 2\pi\rho Vb^2\left(\tfrac{1}{2} + a\right)C(k)\left(V\overline{\alpha} + i\omega\overline{h} + i\omega b\overline{\alpha}\left(\tfrac{1}{2} - a\right)\right)$$
$$+\rho\pi b^2\left(-ba\omega^2\overline{h} + Vb\left(\tfrac{1}{2} - a\right)i\omega\overline{\alpha} + b^2\left(\tfrac{1}{8} + a^2\right)\omega^2\overline{\alpha}\right)$$

Equation 2-7

Where

ω	Frequency of excitation.
$\overline{L}(\omega)$ & $\overline{M}(\omega)$	Aerodynamic Lift and Moment in frequency domain.
$\overline{h}$ & $\overline{\alpha}$	Lift and Torsional displacements.
C(k)	Theodorsen lift deficiency function or Circulation function.
b	is the Semi-Chord Length of the airfoil, B/2.
k	is the dimensionless Reduced Frequency, $\omega b/V$.
a	horizontal offset of centre of elasticity from centre of mass.

The Theodorsen Circulation function C(k) is a complex function defined as

$$C(k)=F(k)+iG(k)$$

Where, F(k) & G(k) are expressed in terms of Bessel functions of the first and second kind,

$$F(k)=\frac{J_1(J_1+Y_0)+Y_1(Y_1-J_0)}{(J_1+Y_0)^2+(Y_1-J_0)^2} \qquad G(k)=-\frac{Y_1Y_0+J_1J_0}{(J_1+Y_0)^2+(Y_1-J_0)^2}$$

The real and imaginary parts of Theodorsen Circulation functions are shown graphically, in the following Figure 2-6.

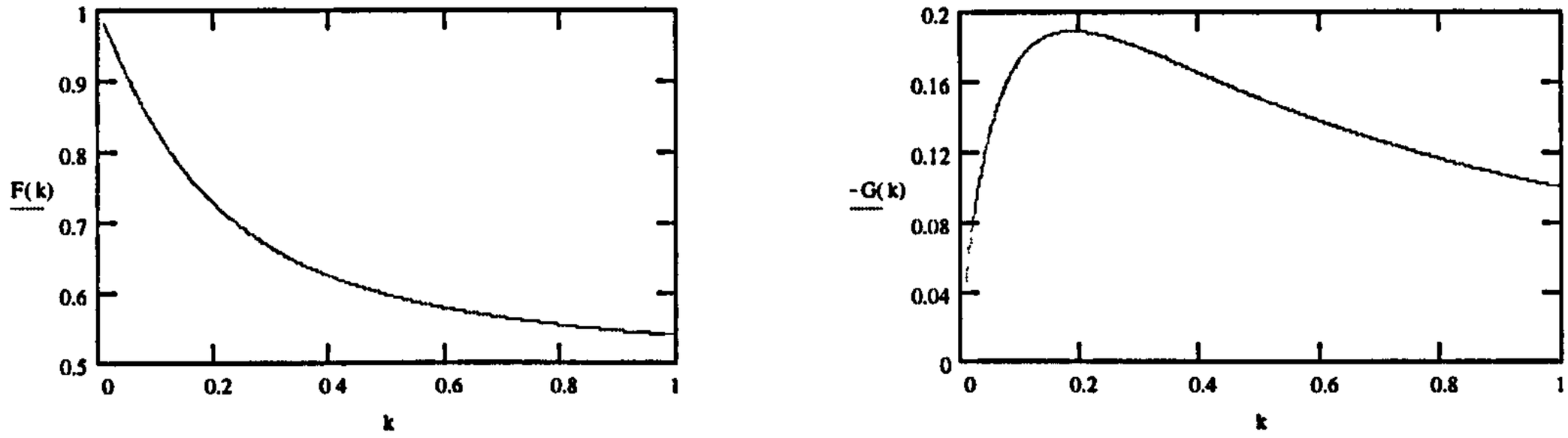


Figure 2-6 Re & Im parts of the Theodorsen Circulation Function.

If the airfoil is assumed to execute a purely harmonic motion at a frequency ω, then we can assume that $\alpha(t)=\overline{\alpha}\cdot e^{i\omega t}$ and $h(t)=\overline{h}\cdot e^{i\omega t}$. Similarly the aerodynamic lift and moment are transformed into the time domain as follows $L(t)=\overline{L}(\omega)\cdot e^{i\omega t}$ and $M(t)=\overline{M}(\omega)\cdot e^{i\omega t}$. Converting the above expressions from the frequency domain into the time domain and setting the elastic centre eccentricity, a, equal to 0, which is the case for a symmetric section, we arrive at the following expressions.

$$L(t)=\pi\rho b\left\{2VF(k)\dot{h}+\left(\frac{2G(k)V^2}{\omega}+bV(1+F(k))\right)\dot{\alpha}-2V\omega G(k)h+\left(2F(k)V^2-\omega bVG(k)\right)\alpha+b\ddot{h}\right\}$$

$$M(t)=\pi\rho b^2\left\{VF(k)\dot{h}+\left(\frac{bV(F(k)-1)}{2}+\frac{G(k)V^2}{\omega}\right)\dot{\alpha}-V\omega G(k)h+\left(F(k)V^2-\frac{\omega bVG(k)}{2}\right)\alpha-\frac{b^2}{8}\ddot{\alpha}\right\}$$

Equation 2-8 & Equation 2-9

A point to be noted here is that of the sign conventions used in the derivation of these expressions and the convention used in the program ANSUSP are different. This will be discussed in Section 3.1.3.

Several new techniques for determining the aerodynamic forces acting on a bridge deck have recently been proposed. Most are based upon the Theodorsen circulation

function, though they are not expressed in terms of Bessel functions, but are expressed as Fourier series [11..14] which are proposed to be computationally more efficient. Following several discussions with Dr Anderson of the Department of Aerospace Engineering at Glasgow University, the decision was taken not to progress with these new methods but to continue to use the traditional method. Justification for this decision comes from the fact that the aeronautics industry in general still apply the Theodorsen formulation derived from Hankel functions which are in turn formed from Bessel functions of the first and second kind.

2.2.1.2 Other work

Theoretical expressions for flutter of a sinusoidaly oscillating airfoil have been developed since the 1930s by several prominent researcher. However the majority of these had been greatly influenced by the general theory of aerodynamic instability for wing flutter developed by Theodorsen.

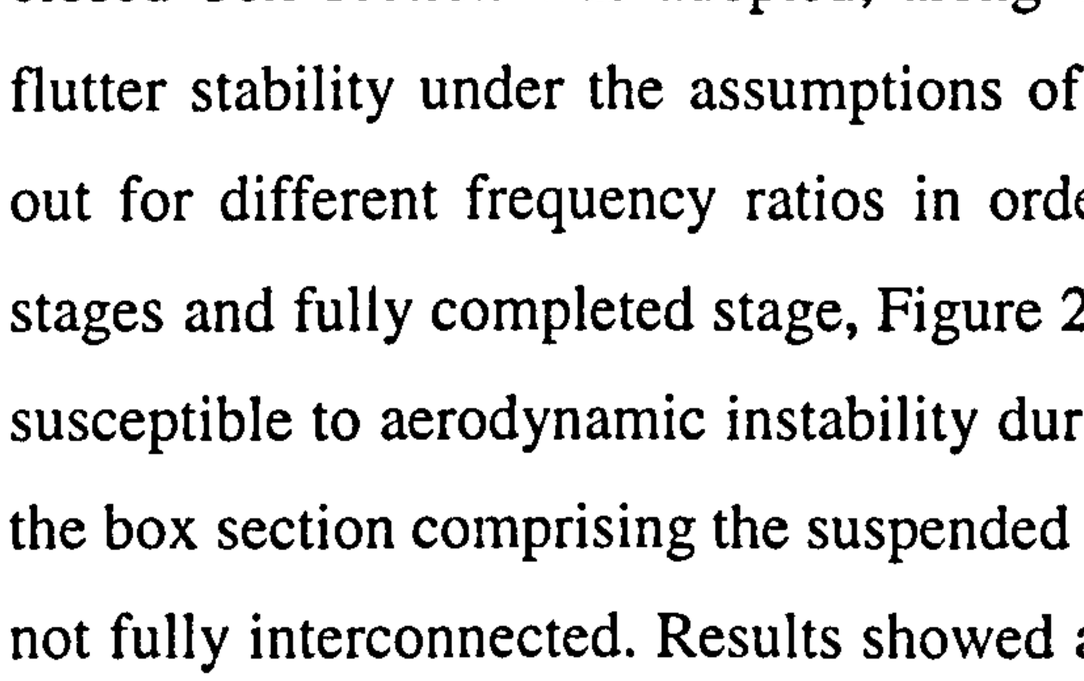
In 1948 Bleich [15] published his work applying aircraft theory to suspension bridge flutter. He developed the equations of motion for a suspension bridge deck idealised as a flat plate with vertical flexural and torsional degrees of freedom. He used Theodorsen aerodynamic force expressions and derived the equations of motion using the Ritz technique and Lagrange's equations. He showed that the flutter characteristics of the whole bridge were similar to those of an independent section of the bridge of unit length under the assumption of fundamental vertical and torsional mode shapes. It was further established that the lowest flutter speed is developed by a combination of the lowest flexural and torsional modes. The effect of structural damping had also been considered.

Bleich accepted that the theory for many real bridges needed further development to take in to account the marked effect of the shape of the cross section of the bridge and provided a formula to be added to Theodorsens which took in to account two parameters. These parameters depend on the profile of the cross section and the reduced frequency, k. These parameters were determined from wind tunnel tests on section models. Bleich applied these modified formulae to structures where the vortex shedding effect was significant and to truss stiffened suspension bridge sections where the effect was comparatively small, from which it was found that

these additional parameters improved the accuracy of the predictions for bluff sections.

In 1946 Scruton [16], began to investigate flutter with the objective of assisting the consulting engineers designing the proposed crossings for the Firth of Forth and the River Severn. In the latter stages of the investigations, the tests were supplemented by experiments using sectional models. In these investigations only the aerodynamic stability of the completed bridges was considered. However, the configurations of the suspended structures and the natural frequencies of oscillation of the bridges differed appreciably during construction from those of the completed structure. Aerodynamic investigations of the erection stages were also made which enabled procedures to be formulated to avoid wind-induced oscillations during construction.

In 1964 Smith [17] published the report on an investigation for the proposed Severn Suspension Bridge. The study included wind tunnel tests on a two dimensional sectional model of the proposed cross section of the bridge for which a streamlined closed box section was adopted, along with a parallel theoretical investigation of flutter stability under the assumptions of Theodorsen flat plate. Tests were carried out for different frequency ratios in order to simulate the structure at its erection stages and fully completed stage, Figure 2-7. The structure was supposed to be more susceptible to aerodynamic instability during the erection stages when the lengths of the box section comprising the suspended structure have been raised into position but not fully interconnected. Results showed a close agreement between Theodorsen flat plate theory and the two-dimensional model tests on the box girder sections proposed for the Severn Bridge.



Variation of critical wind-speed with length of erected centre span for the proposed Severn suspension bridge.
- *Calculated points*
1 *The critical speed is infinite*
2 *Completion of structural steelwork*
3 *Final state of structure (with added mass due to road, pavement, railings etc.)*

Figure 2-7 Variation of critical wind-speed with length of erected centre span for the proposed Severn suspension bridge.

Besides achieving the primary objective, the results of these investigations included design recommendations to avoid wind-excited oscillations of bridges in general, and established the relevant stability parameters, together with methods of test [18]. A special wind tunnel was built for examining the behaviour in steady winds of a full-span aeroelastic model of the proposed suspension bridges. The Severn Bridge was the first faired box section suspension bridge. The cantilevered walkways extending from the box edge contributed in a positive way to the aerodynamic stability of the road deck (Figure 2-8.a).

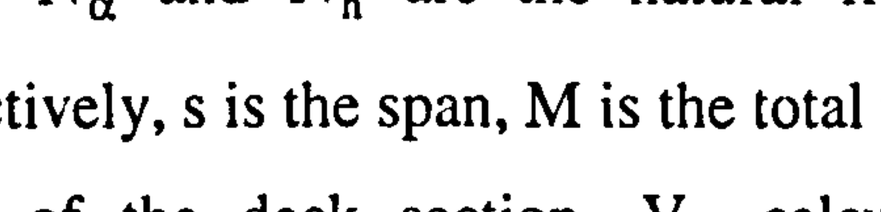
In 1961 Selberg [19], on the basis of wind tunnel tests on two dimensional sectional models, published a remarkably simple formula fcr the determination of flutter speeds V_{cr}, and at the same time gave reduction coefficients for different typical deck sections investigated experimentally by him. For streamlined deck sections for which aerodynamic forces can be assumed to be those for a flat plate, the Selberg formula can be expressed as

$$\frac{V_{cr}}{N_\alpha B} = 3.7\sqrt{\left[\left(\frac{r}{\rho B^3}\frac{M}{s}\right)\left\{1-\frac{N_h}{N_\alpha}\right\}\right]}$$

Equation 2-10

where N_α and N_h are the natural frequencies in torsion and vertical bending respectively, s is the span, M is the total mass, r is the radius of gyration, and B is the width of the deck section. V_{cr} calculated from Equation 2-10 for values of $N_h/N_\alpha < 0.5$ $(N_\alpha/N_h < 2.0)$ the formula is accurate to ~1.5%. For values of $N_h/N_\alpha > 0.7$ $(\sim N_\alpha/N_h < 1.5)$ the formula rapidly becomes inaccurate, and its prediction of $V_{cr} = 0$ when $N_\alpha = N_h$ is at total variance with $V_{cr} = \infty$ predicted by the exact theory.

(a)



(b)

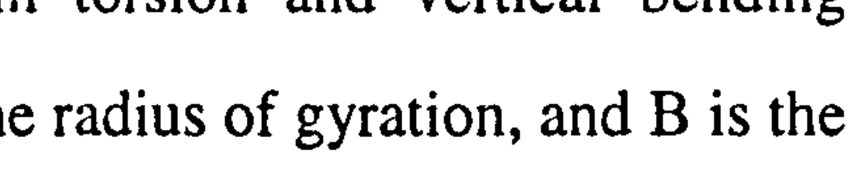


Figure 2-8 Deck profiles of the Severn bridge(a) and Lillaebelt bridge (b)

Work on the stability of suspension bridges continued mainly using scaled oscillatory sectional models equipped with large end plates to assure two-dimensional flow on the model. An innovative design that was geometrically similar to the Severn bridge was the Lillaebelt Bridge in Denmark. Both truss stiffened girders and box girder designs were studied. The final box girder design selected is shown in Figure 2-8b, and was the result of wind tunnel testing undertaken by Selberg at the Technical University of Trondheim

2.2.2 Experimental Methods

2.2.2.1 The Experimental work of Scanlan *et al*

The experimental method developed by Scanlan *et al* [20..27] for the extraction of aerodynamic force information from model tests, is a widely used method in practice, and has undergone significant refinement in the past three decades. The main distinguishing feature of Scanlan's method compared to previously developed methods is that the model deck section is allowed to freely oscillate in the air stream, while previous methods tended to use a mechanically driven section model. Thus the comparative results extracted for similar sections are quite markedly different.

The theoretical basis of the model is that a typical bridge deck section of unit span is allowed to oscillate freely in a steady wind stream. The system is idealised as a two degree of freedom system, one vertical, the other torsional about the longitudinal axis, suspended in a constant cross wind. This is realised in the physical model by placing large endplates at the ends of the deck section during the wind tunnel test, thus insuring the wind stream is two dimensional. The equations of motion for this system are given by

$$\ddot{h} + 2\zeta_h \omega_h \dot{h} + \omega_h^2 h = H_1 \dot{h} + H_2 \dot{\alpha} + H_3 \alpha \equiv \frac{L_h}{m}$$

$$\ddot{\alpha} + 2\zeta_\alpha \omega_\alpha \dot{\alpha} + \omega_\alpha^2 \alpha = A_1 \dot{h} + A_2 \dot{\alpha} + A_3 \alpha \equiv \frac{M_\alpha}{I_p}$$

Equation 2-11 & Equation 2-12

Where

$H_i \& A_i$	General Aerodynamic Coefficients, which are functions of the reduced velocity.
$\omega_h \& \omega_\alpha$	The natural undamped frequencies for the vertical and torsional degrees of respectively, (Rads/sec)
$\zeta_h \& \zeta_\alpha$	Structural damping ratios for the vertical and torsional degrees of freedom respectively.
m	Mass of bridge deck per unit span.
I_p	Mass moment of inertia of bridge deck per unit span.

H_i & A_i are referred to as flutter derivatives and are extracted from wind tunnel tests of scaled bridge deck section. The flutter derivatives fall into two main categories namely direct-flutter derivatives and cross-flutter derivatives. Direct-flutter derivatives are the derivatives associated with displacements or velocities in the same plane as the aerodynamic force, that is for the first of Equation 2-11 & Equation 2-12, the direct-flutter derivatives associated with lift are those in terms of h and $\dot{h}$. Cross-flutter derivatives on the other hand are associated with displacements or velocities corresponding to the lift forces that arise from pitching motions and vice versa. The method originally proposed by Scanlan for determining these derivatives was as follows. The torsional and vertical motions of the model would be restrained independently, from these tests the direct-flutter derivatives H_1 and A_2, A_3 were obtained respectively. Then the cross-flutter derivatives H_2, H_3 and A_1 were calculated from analytical expressions, as functions of the direct-flutter derivatives.

However in a recent paper Scanlan [27] acknowledged that flutter derivatives are only truly accurate if extracted from fully coupled motion wind tunnel tests. To address the short falls of his earlier method, a new extraction method was developed called the Modified Ibrahim Time Domain (MITD). This method allows all the derivatives, both direct-flutter and cross-flutter to be calculated simultaneously from a freely oscillating model, thus removing the necessity of restraining single degrees of freedom to determine individual flutter derivatives. This improved method also reduces the possibility of significant errors and bias in the extracted flutter derivatives.

Once all the flutter derivatives have been determined, they are normalised to become general flutter derivatives. This also ensures that they are non-dimensional, thus allowing the extracted data sets to be used in the investigation of any bridge that has the same general cross sectional shape.

$$H_1^* = \frac{m}{\rho b^2 \omega} H_1 \qquad A_1^* = \frac{I_p}{\rho b^3 \omega} A_1$$

$$H_2^* = \frac{m}{\rho b^3 \omega} H_2 \qquad A_2^* = \frac{I_p}{\rho b^4 \omega} A_2$$

$$H_3^* = \frac{m}{\rho b^3 \omega} H_3 \qquad A_3^* = \frac{I_p}{\rho b^4 \omega} A_3$$

Equation 2-13

where

$H_i^* \& A_i^*$ Non-dimensional Flutter Derivative Coefficients, which are functions of the reduced velocity.

ρ Density of the air stream.

Note: The above expressions have been normalised with respect to the semi-chord width b. However, they are also sometimes normalised with respect to B the full chord width.

These non-dimensional flutter derivatives are then used in the linearised expressions originally proposed by Scanlan and Tomko in 1971[22], Equation 2-14 & Equation 2-15. The original expression was derived in terms of the full chord length B, with corresponding reduced frequency $K = \omega B/V$. Examples of some non-dimensional flutter derivative coefficients are shown in Figure 2-9. The flutter derivatives, coefficients which multiply the displacement and velocity components of the two degrees of freedom in the aerodynamic forces, are postulated to be dependent on the geometry of the bridge deck and reduced velocity ($V/\omega b$).

$$\overline{L}(K) = \frac{1}{2}\rho V^2 (2B)\left[KH_1^*(K)\frac{\dot{h}}{V} + KH_2^*(K)\frac{B\dot{\alpha}}{V} + K^2 H_3^*(K)\alpha \right]$$

$$\overline{M}(K) = \frac{1}{2}\rho V^2 (2B^2)\left[KA_1^*(K)\frac{\dot{h}}{V} + KA_2^*(K)\frac{B\dot{\alpha}}{V} + K^2 A_3^*(K)\alpha \right]$$

Equation 2-14 & Equation 2-15

Where

$\overline{L}(K) \& \overline{M}(K)$ Frequency dependent Aerodynamic Lift and Moment.

V Steady wind speed.

This expression has been significantly modified throughout the past three decades, increasing the number of flutter derivatives from 6 to 8, and most recently to 18[27],

with the addition of an extra expression, defining the out of plane sway motion of the deck of long span bridges.

$$\overline{L}(K)=\frac{1}{2}\rho V^2 B\left[KH_1^*(K)\frac{\dot{h}}{V}+KH_2^*(K)\frac{B\dot{\alpha}}{V}+K^2H_3^*(K)\alpha+K^2H_4^*(K)\frac{h}{B}+KH_5^*(K)\frac{\dot{p}}{V}+K^2H_6^*(K)\frac{p}{B}\right]$$

$$\overline{M}(K)=\frac{1}{2}\rho V^2 B^2\left[KA_1^*(K)\frac{\dot{h}}{V}+KA_2^*(K)\frac{B\dot{\alpha}}{V}+K^2A_3^*(K)\alpha+K^2A_4^*(K)\frac{h}{B}+KA_5^*(K)\frac{\dot{p}}{V}+K^2A_6^*(K)\frac{p}{B}\right]$$

$$\overline{P}(K)=\frac{1}{2}\rho V^2 B\left[KP_1^*(K)\frac{\dot{p}}{V}+KP_2^*(K)\frac{B\dot{\alpha}}{V}+K^2P_3^*(K)\alpha+K^2P_4^*(K)\frac{p}{B}+KP_5^*(K)\frac{\dot{h}}{V}+K^2P_6^*(K)\frac{h}{B}\right]$$

Equation 2-16, Equation 2-17 & Equation 2-18

Where

$\overline{L}(K),\overline{M}(K)\&\overline{P}(K)$ Aerodynamic Lift, Moment and Sway forces respectively.

However due to difficulties in obtaining the out of plane derivatives from a two-dimensional wind tunnel model these expressions are currently simplified and used with only 8 Flutter derivatives.

$$\overline{L}(K)=\frac{1}{2}\rho V^2(2B)\left[KH_1^*(K)\frac{\dot{h}}{V}+KH_2^*(K)\frac{B\dot{\alpha}}{V}+K^2H_3^*(K)\alpha+KH_4^*(K)\frac{h}{B}\right]$$

$$\overline{M}(K)=\frac{1}{2}\rho V^2(2B^2)\left[KA_1^*(K)\frac{\dot{h}}{V}+KA_2^*(K)\frac{B\dot{\alpha}}{V}+K^2A_3^*(K)\alpha+KA_4^*(K)\frac{h}{B}\right]$$

Equation 2-19 & Equation 2-20

It is worth noting that these expressions can be expressed with respect to the semi-chord length b. Since the theoretical solution proposed by Theodorsen was derived using aeronautical conventions, it is useful to convert the above expression to provide a rational basis for comparison. These modified expressions are coded into the ANSUSP program, refer to Section 3.1.3)

$$\overline{L}(k) = \frac{1}{2}\rho V^2(2b)\left[k\tilde{H}_1^*(k)\frac{\dot{h}}{V} + k\tilde{H}_2^*(k)\frac{b\dot{\alpha}}{V} + k^2\tilde{H}_3^*(k)\alpha + k\tilde{H}_4^*(k)\frac{h}{b}\right]$$

$$\overline{M}(k) = \frac{1}{2}\rho V^2(2b^2)\left[k\tilde{A}_1^*(k)\frac{\dot{h}}{V} + k\tilde{A}_2^*(k)\frac{b\dot{\alpha}}{V} + k^2\tilde{A}_3^*(k)\alpha + k\tilde{A}_4^*(k)\frac{h}{b}\right]$$

Equation 2-21 & Equation 2-22

Where

$\tilde{H}_i^* \& \tilde{A}_i^*$	Non-Dimensional Flutter Derivative Coefficients, normalised with respect to the semi-chord length b.
k	Reduced frequency ($\omega b/V$)

The flutter derivatives measured and normalised with respect to the full chord length can be converted for use with the above expression using the conversion factors below, Equation 2-21 & Equation 2-22. In wind engineering it is common to use K ($\omega B/V$) instead of k ($\omega b/V$), the difference being that K is defined in terms of the full chord length of the bridge deck, while k is defined in terms of the semi-chord length, b=B/2. For consistency, the aeronautical notation k will be used. $\tilde{H}_i^*$ and $\tilde{A}_i^*$ are associated with k, while H_i^* and A_i^* are related to K.

$$\tilde{H}_1^* = 4H_1^* \qquad \tilde{H}_2^* = 8H_2^* \qquad \tilde{H}_3^* = 8H_3^* \qquad \tilde{H}_4^* = 4H_4^*$$
$$\tilde{A}_1^* = 8A_1^* \qquad \tilde{A}_2^* = 16A_2^* \qquad \tilde{A}_3^* = 16A_3^* \qquad \tilde{A}_4^* = 8A_4^*$$

Equation 2-23

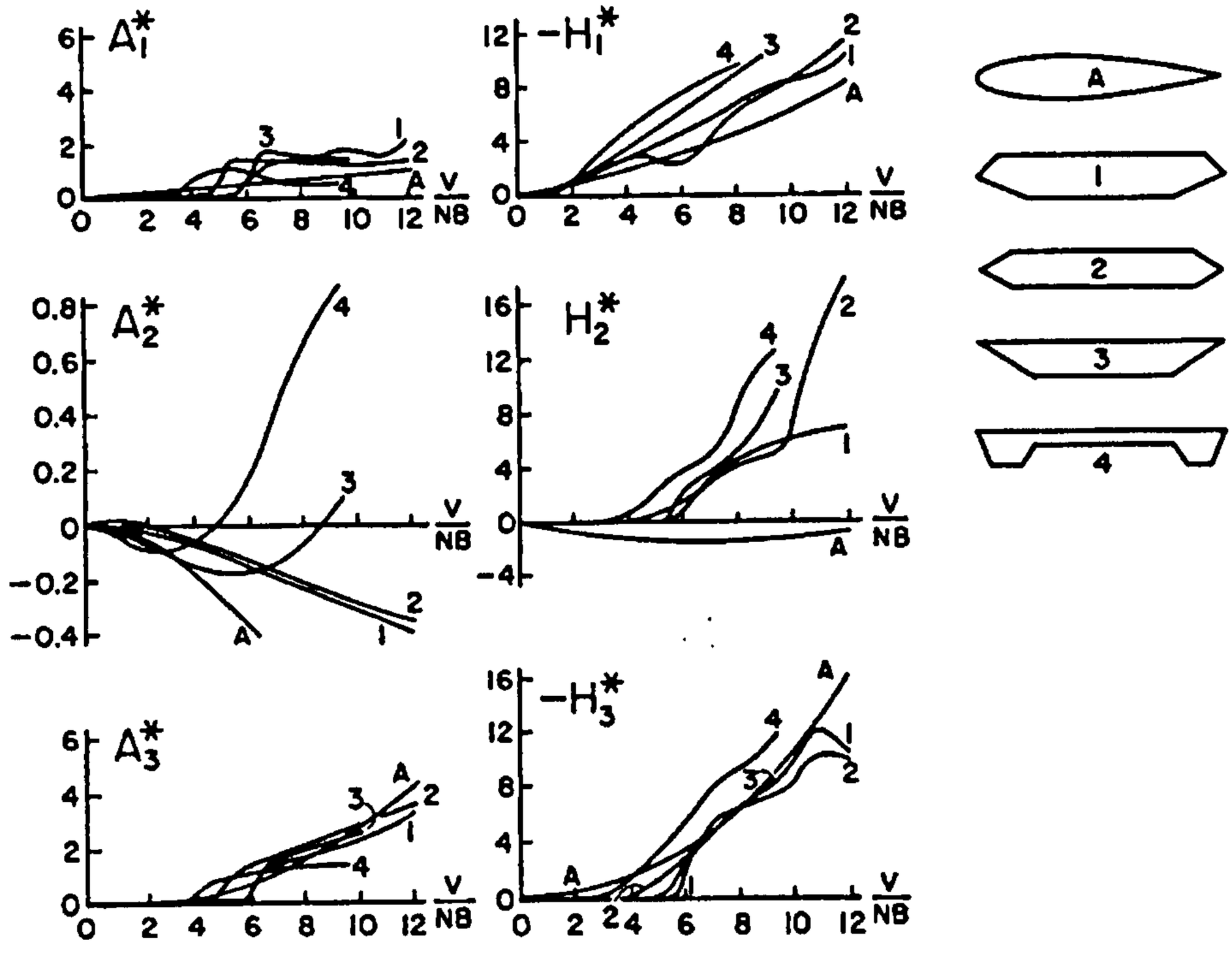


Figure 2-9 Typical Flutter Derivatives

In comparing the Theodorsen expressions Equation 2-8 & Equation 2-9 with the modified linearised expressions proposed by Scanlan, the corresponding flutter derivative coefficients can be defined as continuous functions, as shown graphically in Figure 2-10

$$\tilde{H}_1^* = \frac{-2\pi F(k)}{k} \qquad \tilde{A}_1^* = \frac{\pi F(k)}{k}$$

$$\tilde{H}_2^* = \frac{-\pi\left[1+F(k)+\frac{2G(k)}{k}\right]}{k} \qquad \tilde{A}_2^* = \frac{\frac{\pi}{2}\left[1+F(k)+\frac{2G(k)}{k}\right]}{k}$$

$$\tilde{H}_3^* = \frac{-2\pi\left[F(k)-\frac{kG(k)}{2}\right]}{k^2} \qquad \tilde{A}_3^* = \frac{\pi\left[F(k)-\frac{kG(k)}{2}\right]}{k^2}$$

$$\tilde{H}_4^* = \frac{2\pi G(k)}{k} \qquad \tilde{A}_4^* = \frac{-\pi G(k)}{k}$$

Equation 2-24

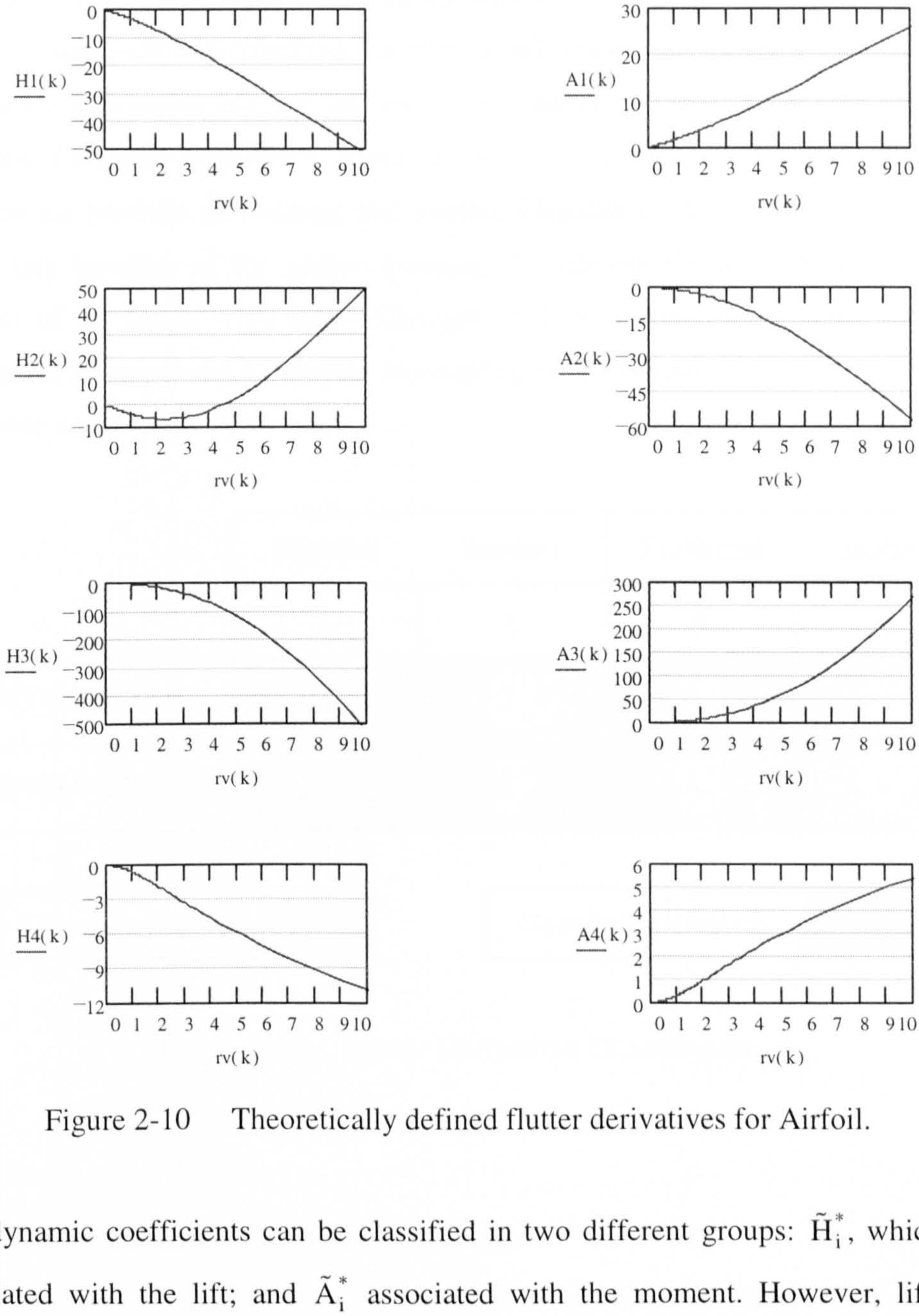


Figure 2-10 Theoretically defined flutter derivatives for Airfoil.

Aerodynamic coefficients can be classified in two different groups: $\tilde{H}_i^*$, which are associated with the lift; and $\tilde{A}_i^*$ associated with the moment. However, lift and moment expressions depend on both degrees of freedom vertical and torsional, so we can talk about terms associated with vertical motion which multiply h and $\dot{h}$, and terms associated with torsional motion which multiply α and $\dot{\alpha}$. For each group direct derivatives are those not involving the motion in the other degree of freedom and coupling terms are those affected by torsional degrees of freedom in the lift

expression, and those affected by vertical degrees of freedom in the moment expression. Figure 2-11 summarises this classification.

Physically, the direct flutter derivatives manifest themselves by modifying the structural damping and structural stiffness in all degrees of freedom as a result of aeroelastic interaction and are included in the aerodynamic damping and stiffness matrices. Coupling derivatives affect the aerodynamic forces indicating coupling between the motions of bending and torsion. Changes in flutter derivatives which imply any increase of the global damping or stiffness should correspond to an increase of the flutter wind speed. Changes in flutter derivatives which implies an increase in aerodynamic forces due to coupling motion should imply an increase of the flutter wind speed.

	Vertical motion		**Torsional motion**	
	$\dot{h}$	h	$\dot{\alpha}$	α
Lift expression	$\tilde{H}_1^*$	$\tilde{H}_4^*$	$\tilde{H}_2^*$	$\tilde{H}_3^*$
Moment expression	$\tilde{A}_1^*$	$\tilde{A}_4^*$	$\tilde{A}_2^*$	$\tilde{A}_3^*$

Direct derivatives | Coupled derivatives

Figure 2-11 Flutter Derivatives Classification

2.2.2.2 Discussion of Scanlan Methods.

In his early paper Scanlan [20] noted that the distinguishing feature, with regards to the aerodynamic flutter of airfoils using potential flow theory, was that because of the smooth trailing edge of the airfoil, no vortices are shed unless the airfoil is in motion. This one important feature quite clearly differentiates between the aerodynamics for the case of many bridge decks (and other bluff objects) on the one hand and airfoils on the other. He also raised questions about attempting to apply

Theodorsen's flutter aerodynamics expressions to flutter analysis of bridge deck sections.

However, it is important to consider the context in which this was written. The vast majority of suspension bridges being built in the early 1960s were truss-stiffened girders, which generally have very poor aerodynamic characteristics, but are not particularly susceptible to flutter.

In a later paper (Scanlan [21]) concerning the aerodynamic behaviour of box-girder bridges, scanlan's conclusions about the suitability of the Theodorsen expressions, were quite different. This moderation of attitude has continued in his more recent papers, where he has stated that the aerodynamic instability of suspension bridge decks and the flutter of airfoils are closely paralleled, thus allowing the Theodorsen expressions to be used as a guide to the stability of bridge decks.

Generally the theoretical expression as a guide to the stability of suspension bridges is accepted to be quite accurate, especially for faired box-girders. The suitability of the theoretical expressions for truss-stiffened girders and plate-girders reduces to the point where the solution is clearly no longer applicable, thus experimental flutter derivatives must be obtained for these sections.

The experimentally obtained flutter derivatives by Poulsen *et al* [4] for wind tunnel tests on the proposed Great Belt, East Bridge, which is an aerodynamic faired box-girder, the curves are smooth and continuous with no discontinuities. Comparing these experimental derivatives with the theoretically defined derivatives, given by Equation 2-24 the comparison is excellent, justifying the use of either the theoretically defined derivatives or the Theodorsen expressions, Equation 2-8 & Equation 2-9 for flutter analysis of suspension bridges.

As was mentioned previously, the expressions proposed by Scanlan have changed continuously, thus demanding a great deal of care when selecting a data set of flutter derivatives. The main inconsistency in these expressions is notation. The early expressions were normalised with respect to the semi-chord width, however, this gradually changed to the full-chord width, Equation 2-14 & Equation 2-15. The

implication of this is that the flutter derivatives are significantly different due to the different definitions of the reduced frequency K and k, the former being defined with respect to full chord width, while the latter with respect to the semi chord width.

Secondly, the magnitude of the multiplying factor has changed from $\frac{1}{2}\rho V^2(2B)$; Equation 2-14, to $\frac{1}{2}\rho V^2 B$; Equation 2-16. The significance of this is that the magnitudes of the flutter derivative coefficients for the latter case are twice those of the former. Hence great care is needed in using a new set of flutter derivatives.

2.2.2.3 Other Experimental Work

In 1981 a draft copy of the design rules for aerodynamic effects on bridges [7] was published. Modifications to these are presented in the current UK code guidance in 1993 [9]. This code is available for structures with individual spans not exceeding 200m. For spans greater than this it is advised that the stability should be verified by wind-tunnel tests. There are two different types of wind model tests.

2.2.2.3.1 Full Model

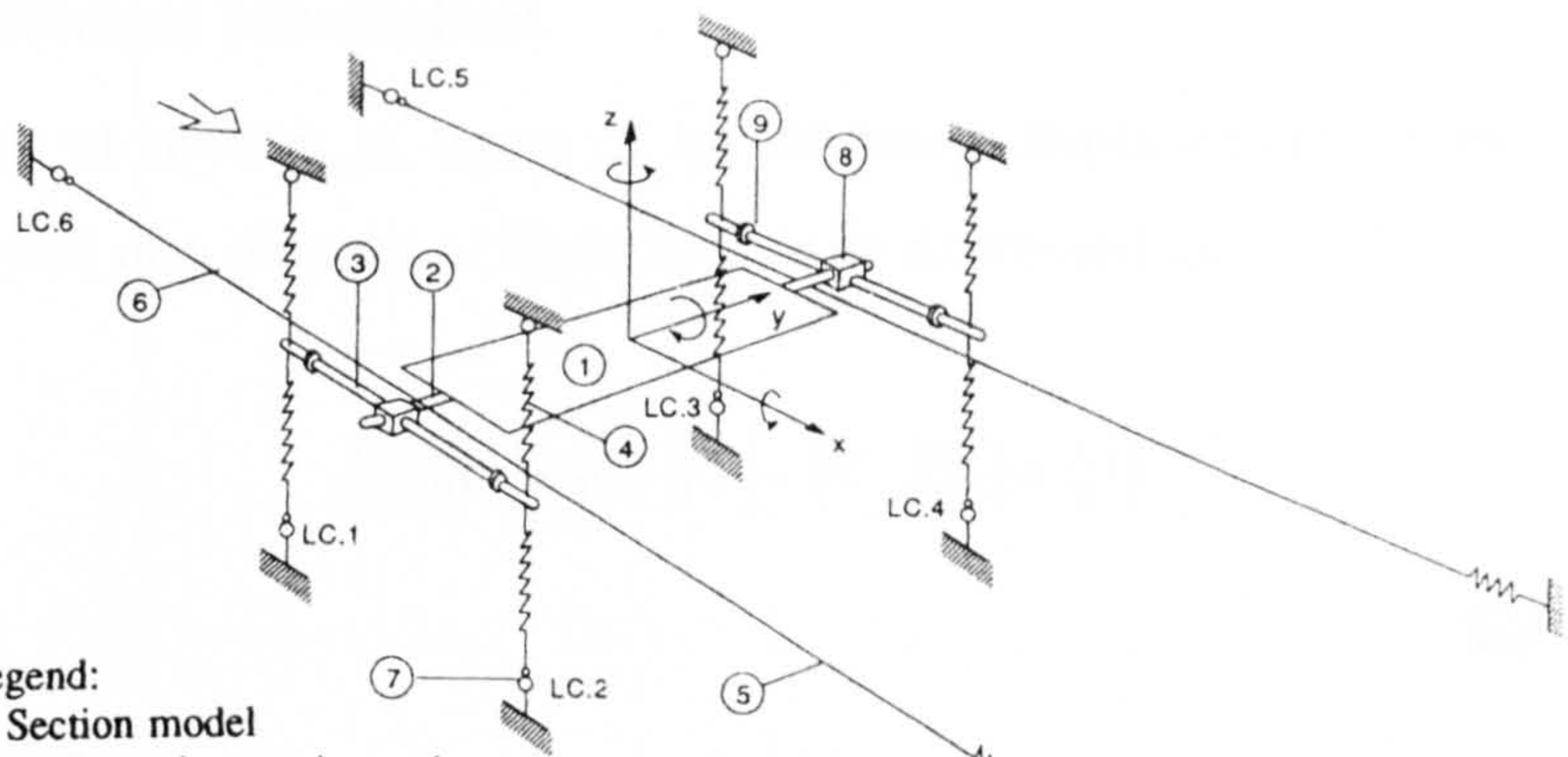
The aerodynamic behaviour of a structure in wind is most accurately represented on a model scale by a "full model". A full aeroelastic model [28] of the structure is constructed to be geometrically similar to the prototype at least with regard to the external shape. It should strictly have the same mass, stiffness and damping distributions and the same values of some non-dimensional parameters of the prototype. While the mass and stiffness distribution may be reasonably well reproduced, an aeroelastic model does not automatically reproduce the correct structural damping distribution and it may be necessary to incorporate devices to increase the damping with in the model.

Wind tunnel testing of a full aeroelastic model has seen a revival with the number of record breaking spans currently under design and construction. Tests on full aeroelastic bridge models are expensive and time consuming but reduce the possibility of encountering unexpected phenomena on the completed structure. However, its cost is relatively small in relation to the cost of the bridge itself. The aeroelastic model is a part of an overall programme that always includes sectional model tests and analytical studies.

2.2.2.3.2 Sectional models

Sectional models [29] are rigid geometrical copies of a typical length of the full-structure. The sectional model is supported in the wind tunnel by springs that provide the required stiffness. Damping is reproduced by an electromagnetic device consisting of a copper plate fixed to a tube between the poles of electromagnets. The suspension allows vertical and pitching motions to occur separately or in a coupled

motion. Sectional models are now used to investigate the aerodynamic stability of the actual long and slender suspension bridges structure. The methods have changed with the advent of computer-based data acquisition and analysis techniques, to commonly include buffeting measurements in simulated turbulent flows and sometimes in the extraction of aerodynamic derivatives, section 2.2.2.2.



Legend:

1. Section model
2. Horizontal extension rod
3. Arm
4. Helical springs
5. Leeward drag wire with helical spring
6. Windward drag wire
7. Load tranducers LC1 through 6
8. Centre piece between arm and extension rod
9. Dummy mass

Figure 2.8 The Experimental set up of a section model.

2.2.3 Analytical Methods

There are two main numerical methods that can be employed to solve the flutter problem. The equations of motion can be explicitly numerically integrated, or a modal synthesis method in the frequency domain can be used. Both methods will be outlined in the following section.

2.2.3.1 Numerical Integration

The equations of motion in terms of an arbitrary displacement vector {U} for a structural system of n degrees of freedom can be expressed as:

$$[M]\{\ddot{U}\}+[C_s]\{\dot{U}\}+[K_s]\{U\}=\{P\}$$

Equation 2-25

Where:

[M] Mass matrix (n x n).

$[C_s]$ Structural damping matrix.

$[K_s]$ Stiffness matrix sum of K_e and K_g, the elastic and structural geometric stiffness respectively.

{P} Force vector. In general the elements of the vector {P} contain the effects of: dead loads, self weight, static imposed loads, static wind loads, and dynamic wind loads (gusting forces and flutter forces).

{U} vector of displacement of the structure.

Time-History Method is used to solve the entire response of the system subjected to any arbitrary loading of any duration. This technique solves the equations of forced vibration directly, in an incremental or step-by-step manner. The time derivatives are replaced by differences of displacement and velocity at various instants of time. For each increment all the terms involved in Equation 2-25 are calculated which allows the response of the structure for that period of time to be followed.

Finite difference methods for approximately solving these have been developed. However in the past two decades, methods that are particularly efficient for transient

finite element problems have flourished. There are two categories of numerical integration methods. These are explicit and implicit methods. The explicit methods have the form;

$$\{U\}_{n+1} = f\left(\{U\}_n, \{\dot{U}\}_n, \{\ddot{U}\}_n, \{U\}_{n-1}, ...\right)$$

thus $\{U\}_{n+1}$ is determined in terms of the complete time history of displacements and time derivatives for time $n\Delta t$ and earlier. Implicit methods have the form;

$$\{U\}_{n+1} = f\left(\{\dot{U}\}_{n+1}, \{\ddot{U}\}_{n+1}, \{U\}_n, ...\right)$$

hence $\{U\}_{n+1}$ is computed in terms of the unknown time derivatives of $\{U\}_{n+1}$.The two methods however have markedly different characteristics.

Most implicit methods are unconditionally stable and have no restriction on the time step size Δt other than as required for accuracy. A popular unconditionally stable implicit method is the Trapezoidal Rule or Average Acceleration Method. This method relates the displacements, velocities and accelerations as shown in Equation 2-26. The method is numerically stable for any size of time step Δt, but the size is governed by the requirement for accuracy.

$$\{U\}_{n+1} = \{U\}_n + \frac{\Delta t}{2}\left(\{\dot{U}\}_n + \{\dot{U}\}_{n+1}\right)$$

$$\{\dot{U}\}_{n+1} = \{\dot{U}\}_n + \frac{\Delta t}{2}\left(\{\ddot{U}\}_n + \{\ddot{U}\}_{n+1}\right)$$

Equation 2-26

There are many other implicit methods such as the Houbolt Method. However, even though the method is unconditionally stable, it introduces numerical or algorithmic damping that is too high for low-frequency response, thus excluding it for the present research, where the natural frequencies of the structure are generally low.

The majority of the remaining implicit routines are based on the Newmark Method, which will be dealt with in detail. The basis of this method is that the analyst can

choose the parameters β and γ in Equation 2-27 to control the numerical accuracy and stability of the numerical integration with time.

$$\{U\}_{n+1} = \{U\}_n + \Delta t\{\dot{U}\}_n + \Delta t^2\left[\left(\frac{1}{2}-\beta\right)\{\ddot{U}\}_n + \beta\{\ddot{U}\}_{n+1}\right]$$
$$\{\dot{U}\}_{n+1} = \{\dot{U}\}_n + \Delta t\left[(1-\gamma)\{\ddot{U}\}_n + \gamma\{\ddot{U}\}_{n+1}\right]$$

Equation 2-27

the method is unconditionally stable when

$$2\beta \geq \gamma \geq \frac{1}{2}$$

or conditionally stable when

$$\gamma \geq \frac{1}{2}, \quad \beta < \frac{1}{2} \quad \text{and} \quad \Delta t \leq \frac{\xi\left(\gamma - \frac{1}{2}\right) + \sqrt{\gamma/2 - \beta + \xi^2\left(\gamma - \frac{1}{2}\right)^2}}{\omega_{max}\left(\gamma/2 - \beta\right)}$$

where ξ is a percentage of the critical damping of the structure.

ω_{max} is maximum natural frequency of interest.

Table 2-1 below gives a summary of main Newmark methods with the limits for the time step Δt and the corresponding accuracy. It should be noted that the Fox-Goodwin method has accuracy of the fourth order when the structure is considered to be undamped. Generally $\Delta t < 2/\omega_{max}$ which is known as the Courant condition [30].

Method	β	γ	Δt_{max}	Accuracy
Artificially damped	$>\gamma/2$	>1/2	∞	$O(\Delta t)$
Average Acceleration	1/4	1/2	∞	$O(\Delta t^2)$
Linear Acceleration	1/6	1/2	$\approx 3.464/\omega_{max}$	$O(\Delta t^2)$
Fox-Goodwin	1/12	1/2	$\approx 2.449/\omega_{max}$	$O(\Delta t^4)^{(u)}$

Table 2-1 Summary of Newmark methods

The existing ANSUSP program [31..34] was developed using the Linear Acceleration Method, which is a conditionally stable implicit integration scheme with second order accuracy. This was replaced by the Fox-Goodwin method, which like the previous method is a derivative of the Newmark family of integration techniques. It is also conditionally stable but has the benefit of fourth order accuracy. However this is conditional on the fact that the structure is undamped. If structural damping is present the accuracy reduces to be the same as for the Linear Acceleration Method.

The Fox-Goodwin method was found to be beneficial when used for flutter time history analysis that has no structural damping applied. The solutions obtained using the Linear Acceleration and the Fox-Goodwin Methods compared well, with the Fox-Goodwin method proving to be the more efficient method, requiring less iterations to solve the same problem.

An explicit integration scheme based on Central Difference was also implemented into ANSUSP. This is a conditionally stable, second order accuracy method, whose time step is defined by the Courant condition. This method was tested against several problems previously solved using the Linear Acceleration Method.

The explicit method proved to be stable for the first several time steps but then began to diverge rapidly from the expected solution. This was repeated using progressively smaller time steps until a totally stable solution was achieved. The resulting time step was found to be one tenth of that used in the implicit methods.

The reason for this significant reduction in the size of the time step was found to be due to the style of problem being attempted. The basic principle is that equilibrium must be achieved at the end of each time step. Conventionally this is achieved by constructing the structural stiffness matrix for the whole structure. However, in the present approach stiffness matrices are calculated only for the deck as a substructure and the towers as separate substructures. The forces within these substructures are calculated using the current displacement field. The hanger and cable members are each solved individually using the current displacement field within the structure to calculate their individual internal structural forces. Static equilibrium within the structure is obtained by iterating to a solution within a time step, where any out-of-balance in structural forces at a joint is applied as an additional external force for the next iteration.

Thus for implicit methods a large time step requires several iterations to soften the solution, while explicit method on the other hand will only yield a meaningful solution if the time step is restrictively small, due to this method's inability to iterate.

Finally, an attempt was made at introducing a mixed integration method based upon the Central Difference and Newmark family of methods. The initial iteration for each time step was obtained using the explicit Central Difference Method, this initial solution being iterated by the implicit Fox-Goodwin Method, until satisfactory accuracy is obtained. However, as expected this mixed method proved to be no more efficient than the implicit methods already implemented.

2.2.3.2 Modal synthesis

Modal synthesis is a commonly used technique for linear elastic structure systems where the response {U} is basically assumed to be a linear combination of the natural mode shapes of the structure. The number of modes included is usually relatively small compared to the total number of degrees of freedom of the system.

This method has the computational advantage over numerical integration methods in that for a system of n degrees of freedom, the response can be calculated using only

m degrees of freedom ($m<<n$), where m is the number of mode shapes required for acceptable accuracy.

The most time consuming and computationally intensive part of this method is the initial extraction of the natural frequencies and associated mode shapes. However, once these have been calculated, it is comparatively inexpensive to consider the behaviour of the system over a large range of wind speeds or general loading.

The numerical idealisation used to model a suspension bridge assumes a non-linear relationship between forces and displacements due to the changes in the geometry of the structure as it deflects. The modal synthesis method however is developed using the theory of superposition that is appropriate only for linear elastic structures, with no non-linearities. Thus using modal synthesis is not strictly appropriate, however it has been found that the errors associated with using this method are negligibly small.

3 The ANSUSP Program

The ANSUSP (ANalysis of SUSPension bridges) program is a three-dimensional suspension bridge dynamic analysis program capable of performing six different forms of analyses. These analyses are all based on two main numerical methods, namely the direct solution of the equations of motion by numerical integration or the use of eigenvalue methods, that yield the natural frequencies and normalised relative responses (modes) of the structure. The main forms of analysis will be discussed below, with the detailed aspects of each type of analysis being outlined. For further details refer to the ANSUSP Users Guide [31..34].

3.1 Time History Analysis

3.1.1 Numerical Modelling procedure

The program ANSUSP developed by Agar [35..38] and Beith [39..44] idealises a suspension bridge as a three-dimensional framework in a manner similar to Iwegbue *et al* [45,46], Figure 3-1. The program has the capability to analyse the structure in either its fully constructed geometry or at an erection stage. The program numerically idealises the suspension bridge as a two cable structure that comprises Tower, Cable, Hanger and Deck Elements.

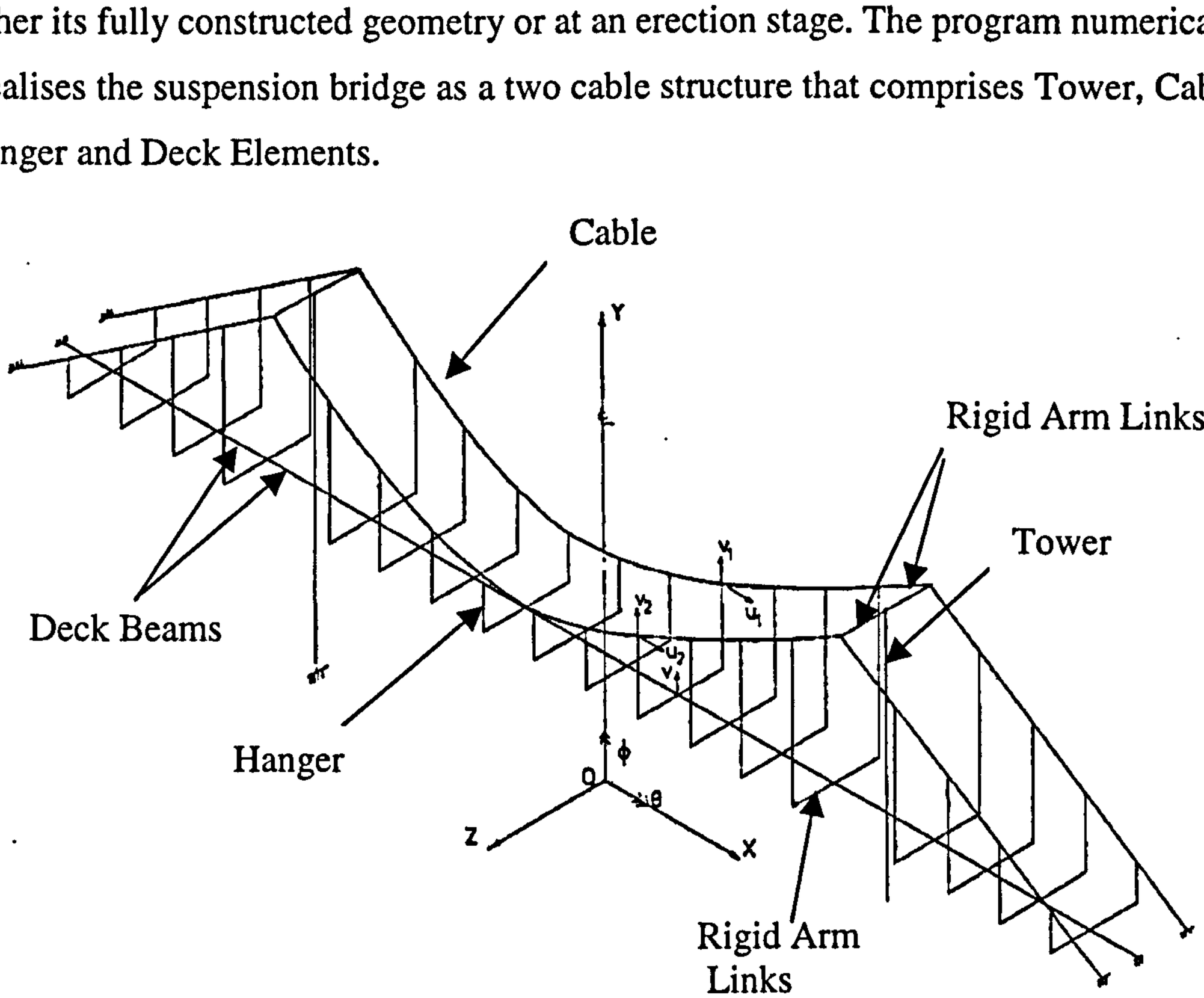


Figure 3-1 Suspension Bridge Framework from Iwegbue [45]

The Deck elements are modelled as beams spanning along the centre line of the bridge. They are modelled with the gross mass, rotational inertial and stiffness properties of full deck cross section. They are modelled in this way so as to be capable of properly accounting for the aerodynamic force that are applied to the deck cross sections when performing a Flutter or Gust analysis.

Each Deck element is capable of resisting lateral and vertical loads as well as being able to resist longitudinal torsional effects, again a crucial aspect when performing flutter analysis. The Hanger elements are connected to the deck using rigid offsets perpendicular to the centroid of the deck.

The Cable elements span between the tops of the Hanger elements and are only capable of carrying axial tensile loads. They are modelled as bar elements and as such are not capable of developing end moments, since in reality they are primarily axial force members and carry very little bending moment, the moment is neglected to simplify the analysis. However they have a considerable effect on the dynamic response of a bridge structure due to their considerable inertial effect.

The Hanger elements connect the Cable elements to the deck elements. Similar to the cable elements, these are also capable of carrying axial tensile loads. The consequence of this is that if the force in the hanger is compressive, then the program sets the force within the elements to zero and redistributes the loading to the rest of the structure until equilibrium is achieved.

The Tower elements are modelled as vertical cantilever beams, fully restrained at their base. These again are modelled to reflect the gross cross sectional properties of the real structure. The towers are considered to have lateral, longitudinal and torsional stiffness, but are consider to be infinitely axially rigid. The cable elements are once again connected to the top of the towers by rigid offsets.

The active degrees of freedom associated with each of the structural elements are shown in summary below

Cable Nodes	Vertical (y)
	Longitudinal (x)
	Lateral (z)
Deck Nodes	Vertical (y)
	Lateral (z)
	Rotation (θ_x)
Deck Nodes	Longitudinal (x)
	Lateral (z)
	Rotation (θ_y)

The inertial characteristics of the structure are modelled by simple lumping of the member element mass at the nodes. This is chosen in preference to any type of consistent mass representation because lumping mass effectively means that the dynamic equation of motion may be solved explicitly, using the numerical integration methods presented in Section 2.2.3.1. If the structure is considered to be undamped the structural masses are uncoupled from one another, thus allowing each equation to be solved individually, if however, a consistent mass matrix was used the equations would need to be solved simultaneously due to the coupling effect of the mass matrix.

The numerical integration method requires that at each time step, the vector of structural restoring forces be calculated. Since in general there is at least some degree of geometric non-linearity, the structural stiffness matrix is not formed explicitly. Rather the resisting structural forces are calculated from the current member end positions so that resolution of the forces are appropriate to their current deflected positions. This is done taking account of the special load carrying limitations of the cable and hanger elements, using a method similar to that presented by Iwegbue and Brotton [45], but which has been expanded to develop a fully three-dimensional solution.

3.1.2 Static Analysis

Static analyses in ANSUSP are obtained using a process initially proposed by Otter[47] and Day[48] called Dynamic Relaxation. This form of static analysis varies from the traditional stiffness method for geometrical non-linear analysis, which is essentially matrix techniques requiring the formation of a global structural stiffness matrix. In contrast dynamic relaxation is a step-by-step procedure of dynamic analysis based on satisfying the dynamic equation of forced vibration and is capable of performing static analyses on the suspension bridge structure during its construction stage, when the structure is highly non-linear. The method of obtaining the static solution is to integrate the equation of motion Equation 3-1 with respect to time until the velocities and accelerations are so small as to be negligible. During this process the stiffness matrix $[K]$ is amended to be consistent with the current deflections and member forces

$$[M]\{\ddot{U}\}+[C]\{\dot{U}\}+[K]\{U\}=\{P\}$$

Equation 3-1

$$[K]\{U\}=\{P\}$$

Equation 3-2

It should be noted that, when the velocities and accelerations are small the inertial and structural damping force terms are negligible, the mass and damping matrices used do not affect the final values of the deflection, they only affect the path by which it is attained. To obtain the solution the structure is critically damped to ensure the structure does not vibrate during the solution procedure but monotonically approaches the static equilibrium position. The value of critical damping is calculated by monitoring the response of the structure during an undamped analysis from which the period of the structure is determined. The critical damping is calculated using the relationship $C_{cr}=4\pi\cdot M/T$, where M is the mass of the structure and T is the period of vibration.

3.1.3 Flutter Analysis

This analysis is essentially the same as for static analysis except that the forces in this case are derived from the aerodynamics of the bridge deck sections. The forces are calculated from either the Theodorsen functions[11] or by the method of Flutter Derivatives, proposed by Scanlan [20..27], (Section 2.2.) Since the Flutter phenomenon is considered to mainly involve motion in the vertical plane, unaccompanied by any considerable transverse motion, no transverse (z) translations are considered in a flutter analysis.

The Theodorsen's and Scanlan's expressions developed in Section 2.2.1.1, were set up using aeronautical sign conventions and notations. Table 3-1 below shows the significant disparities in the positive sign conventions used by the aeronautical fraternity and as coded within the ANSUSP program.

	Aeronautics	ANSUSP
Vertical Displacement	DOWN	UP
Rotation	CLOCKWISE	CLOCKWISE
Lift	UP	UP
Moment	CLOCKWISE	CLOCKWISE†

† this is due to the direction of the applied wind load.

Table 3-1 Sign conventions.

Thus converting the Theodorsen expressions to have consistent notation with ANSUSP, and removing the inertial terms due to their negligible effect, we obtain the expressions that were coded into the ANSUSP program, Equation 3-3 & Equation 3-4

$$L(t) = -\pi\rho b\left\{2VF(k)\dot{h} - \left(\frac{2G(k)V^2}{\omega} + bV(1+F(k))\right)\dot{\alpha} - 2V\omega G(k)h - \left(2F(k)V^2 - \omega bVG(k)\right)\alpha\right\}$$

Equation 3-3

$$M(t) = -\pi\rho b^2\left\{VF(k)\dot{h} + \left(\frac{bV(1-F(k))}{2} - \frac{G(k)V^2}{\omega}\right)\dot{\alpha} - V\omega G(k)h + \left(\frac{\omega bVG(k)}{2} - F(k)V^2\right)\alpha\right\}$$

Equation 3-4

The modified Scanlan expressions coded into ANSUSP are shown in Equation 3-5 & Equation 3-6.

$$\overline{L}(k) = \frac{1}{2}\rho V^2 (2b)\left[-k\tilde{H}_1^*(k)\frac{\dot{h}}{V} + k\tilde{H}_2^*(k)\frac{b\dot{\alpha}}{V} + k^2\tilde{H}_3^*(k)\alpha - k\tilde{H}_4^*(k)\frac{h}{b}\right]$$

Equation 3-5

$$\overline{M}(k) = \frac{1}{2}\rho V^2 (2b^2)\left[-k\tilde{A}_1^*(k)\frac{\dot{h}}{V} + k\tilde{A}_2^*(k)\frac{b\dot{\alpha}}{V} + k^2\tilde{A}_3^*(k)\alpha - k\tilde{A}_4^*(k)\frac{h}{b}\right]$$

Equation 3-6

3.1.3.1 Solution Procedure

The Flutter response contains contributions corresponding to the deck's vertical h and rotational α displacements and their corresponding first time derivatives, the wind speed V being considered and the oscillation frequency ω occurring at that time. These contributions can be evaluated from Equation 3-3 & Equation 3-4, for decks aerodynamically equivalent to flat plates or from Equation 3-5 & Equation 3-6 if flutter derivatives have been measured experimentally for a geometrically similar bridge cross section.

To perform an analysis, the procedure is to choose successively larger wind speeds and examine the characteristics of the response produced. Identifying the lowest wind speed, V_{cr} were the deck oscillates vertically and torsionally at a common frequency, the oscillation changes from being convergent (amplitude decreasing with time, net positive global damping) in nature to divergent (amplitude increasing, net negative global damping).

The circular frequency of deck motion is evaluated by monitoring the cyclic response of a deck node, the time at which maximum amplitudes occur. Figure 3-2, the period of oscillation is approximated as

$$T_i = t_{i+2} - t_i$$

Equation 3-7

And circular frequency

$$\omega_i = \frac{2\pi}{T_i}$$

Equation 3-8

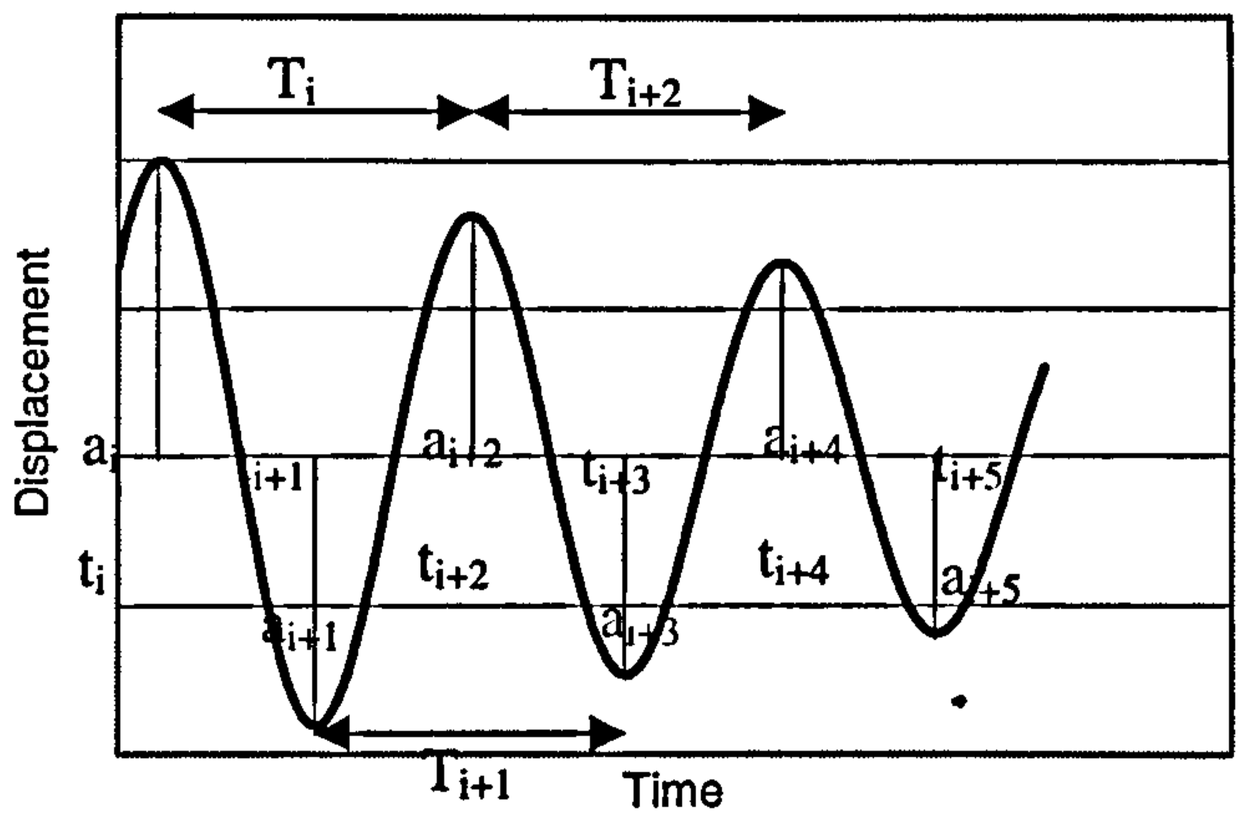


Figure 3-2 Response Monitoring

And these estimates are updated every half cycle and are used in subsequent calculations of aerodynamic forces. Figure 3-2, the systems global damping in terms of logarithmic decrement is evaluated from the magnitudes of successive peaks as

$$\delta_i = 2 \ln \left| \frac{a_i - a_{i+1}}{a_{i+1} - a_{i+2}} \right|$$

Equation 3-9

For the multi-degree of freedom bridge idealisation the amplitude values a_i, a_{i+1}, a_{i+2}, are evaluated as RMS values of the displacement components at the deck nodes.

Given a particular wind speed to be investigated, the deck response to the effects of the aerodynamic forces acting on it have to be monitored as it oscillates. This requires the following approach:-

i. Initiation of an oscillating motion- although the most obvious means of starting a combined vertical and torsional oscillatory motion is to apply an initial set of displacements and then release the structure, there are practical difficulties in deriving compatible sets of displacements. Because of this the approach instead has been to apply a set of initial velocities (being approximately proportional to the displacements) to the structure, and this has worked quite satisfactorily. The program allows the choice of specifying the initial velocities at each node

explicitly or applying velocities proportional to one of the natural mode shapes previously calculated by the program. (Refer to Section 3.2.2).

ii. Estimation of oscillation frequency (on which the aerodynamic forces depend)- prior to the deck completing a full cycle of response oscillation (at which stage the circular frequency may be calculated from Equation 3-8) the program uses an approximation that must be user specified.

iii. As the response of the deck completes further cycles, the oscillation frequency is updated from Equation 3-8 every half cycle and is used to modify the aerodynamic terms until a constant response (equal to the input frequency of excitation in the aerodynamic terms) is attained. At this stage time integration is terminated and the convergent or divergent nature of the established equilibrium oscillation is evaluated in term of logarithmic decrement according to Equation 3-9. A positive value of global damping implies a stable oscillation where energy dissipates, whereas a zero or negative value indicates that flutter instability could exist at that wind speed.

In practice the procedure cannot be implemented as simply as intended above. For small wind speeds which necessarily involves little coupling between the equations of motion (vertical and torsional), global damping is relatively high and the starting oscillations degrade into virtually independent flexural and torsional motions. Because of this it is reasonable to perform the time integration using aerodynamic forces calculated on the basis of separate bending and torsional frequencies produced by their respective individual responses, Bell *et al*[49]. Consequently the ω and k, in Equation 3-3 & Equation 3-4 are replaced by ω_b, k_b, ω_t,, k_t respectively where the subscripts b and t correspond to bending and torsional values. However it should be noted that in doing so the equations are not strictly correct as they only apply to the case where flexural and torsional oscillations are occurring at a single common frequency. A typical set of two-dimensional results are shown in Figure 3-3 for the Severn Bridge deck section.

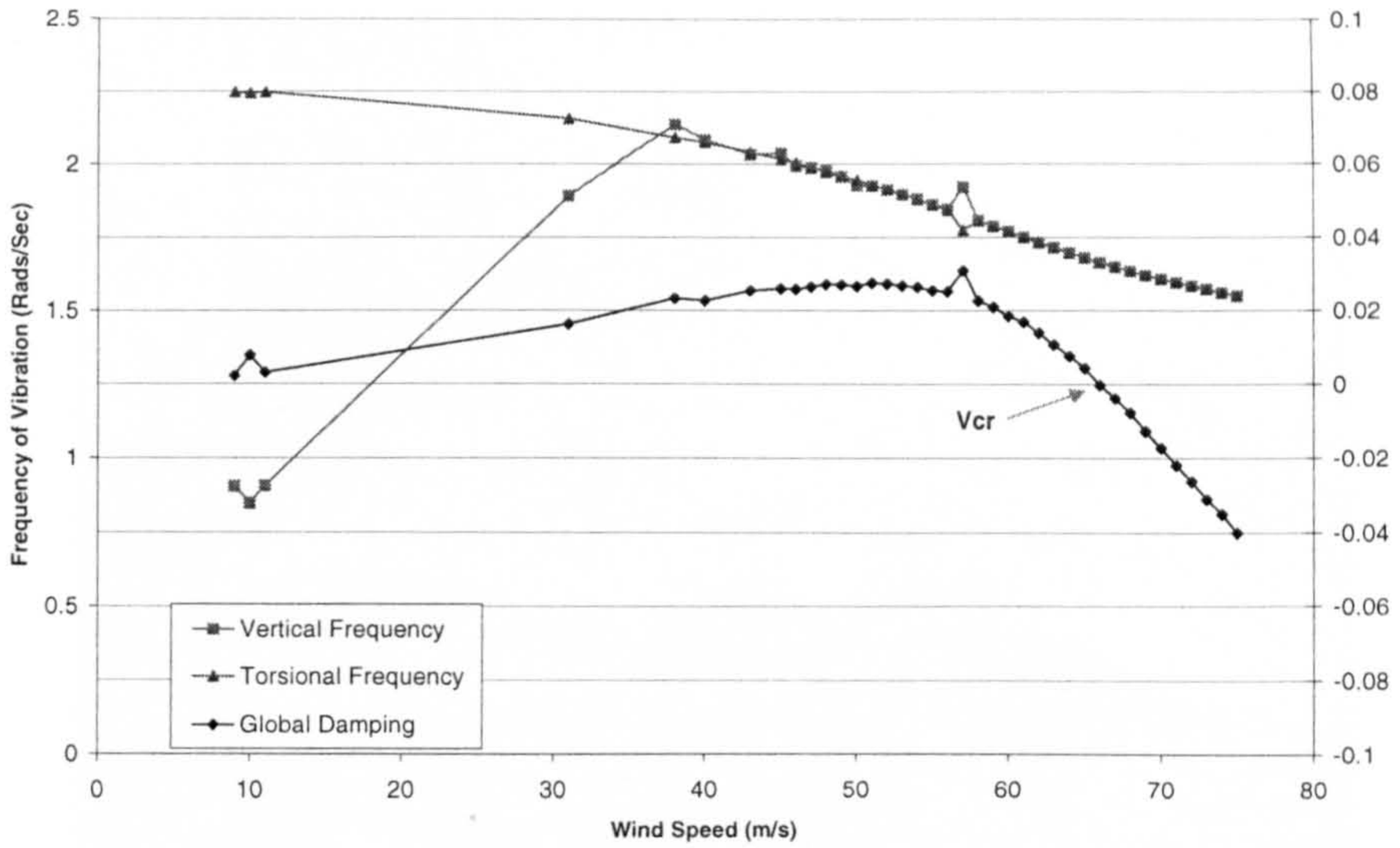


Figure 3-3 Typical Flutter Time-History Analysis Results.

As can be seen at low wind speeds there is no coupling between the flexural and torsional displacements, each responding individually. However, as the applied wind increases the level of coupling increases to a point were the two motions begin to coincide at a common frequency of excitation. This progresses until the point where the level of coupling causes the net global damping to become negative, thus indicating the structural response has become divergent, possibly ultimately leading to complete structural failure, in a manner similar to the Tacoma narrows bridge failure of 1940, Figure 3-4.

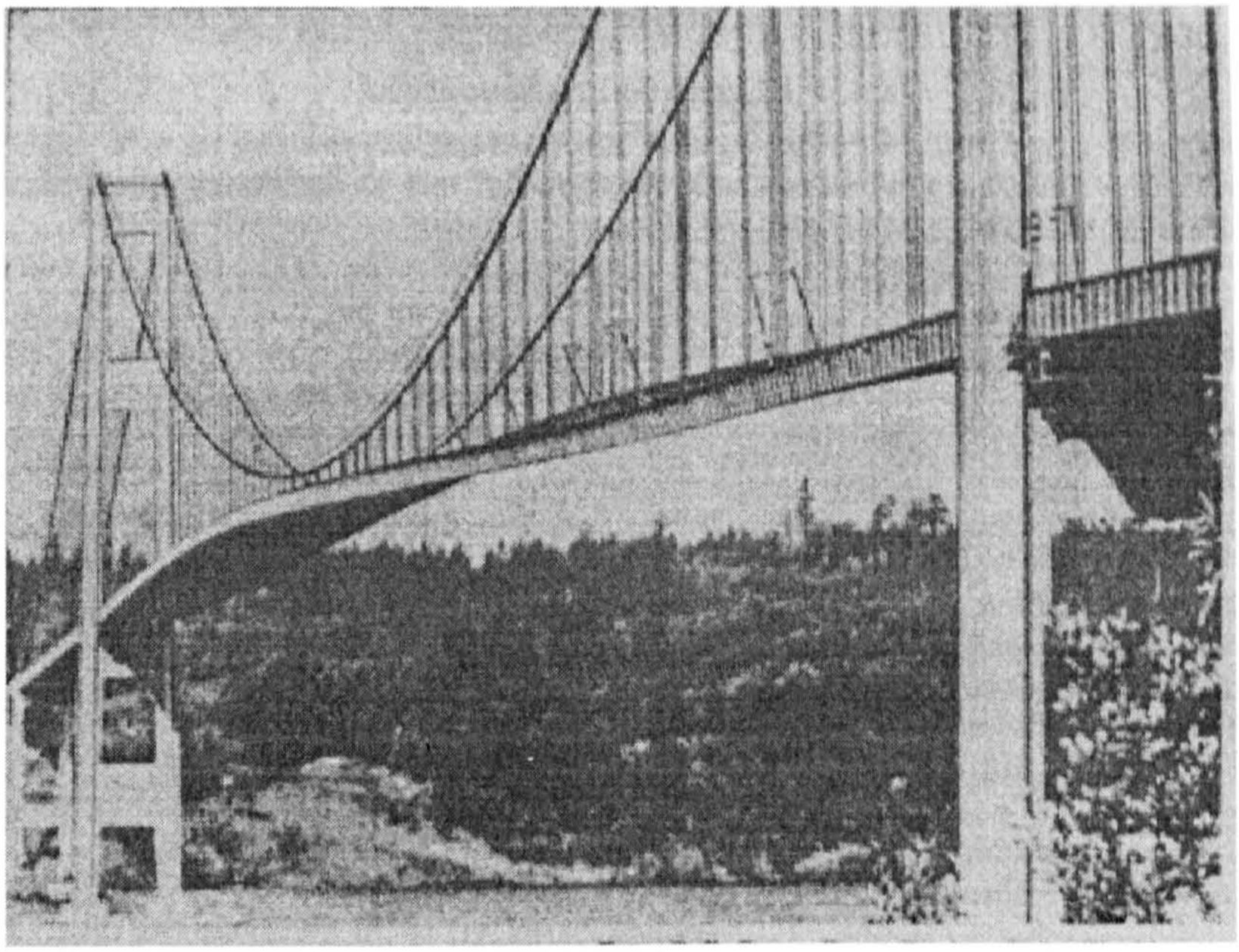


Figure 3-4 Tacoma Narrows Bridge failure of 1940.

The results obtained using this solution procedure on the Severn bridge are shown in Table 3-2. The results are presented for four different numerical models each representing a different level of refinement in the computational model. Figure 3-5, shows the numerical model used in the Severn 1:3 idealisation, the significance of the idealisation label being that 1 finite element represents 3 actual structural sections.

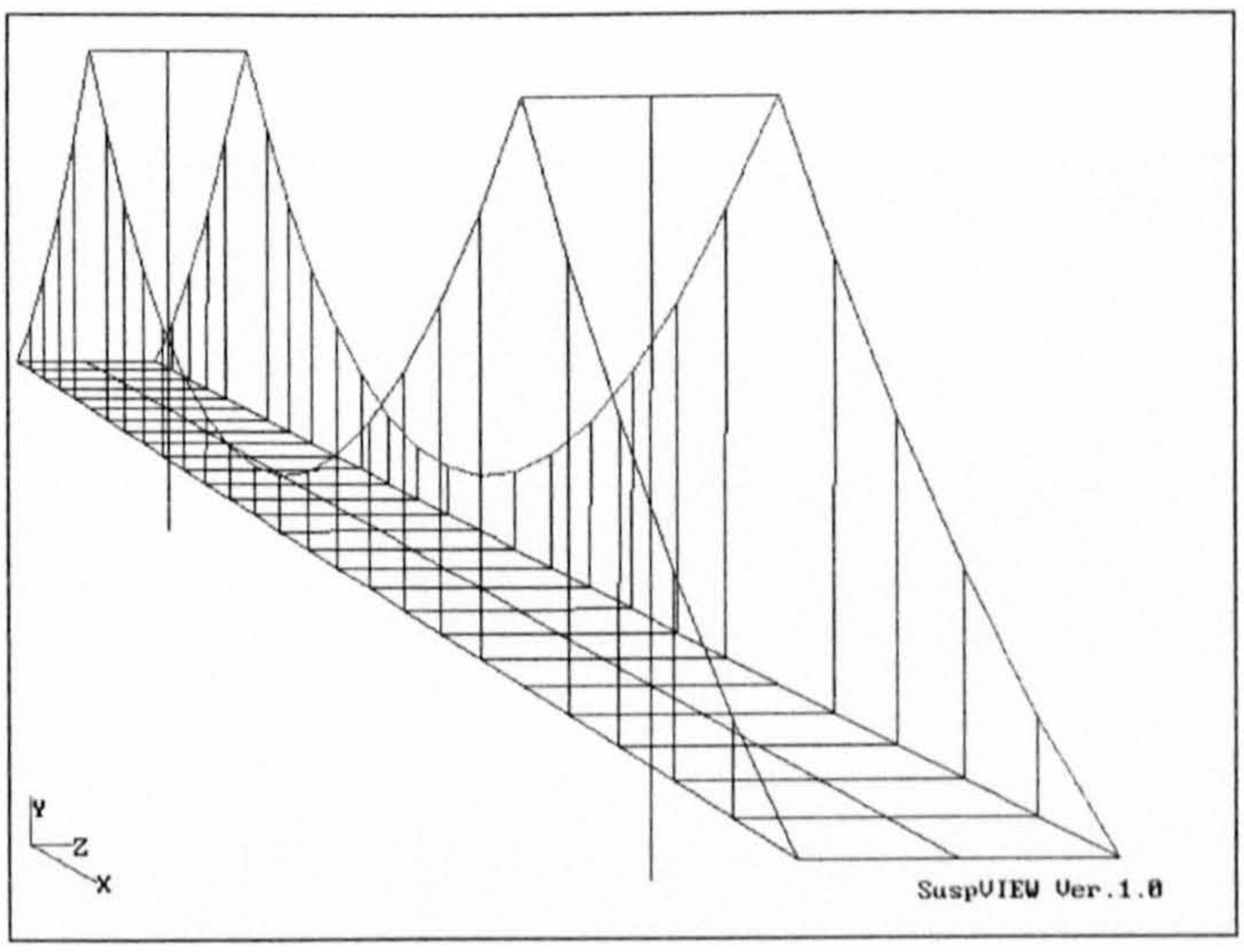


Figure 3-5 Computational Model of Severn Bridge (1:3 Idealisation).

Idealisation	V_{cr}	Error (%)	Analysis Time
1:1	77.12 m/s	-	4 hr
1:2	77.54 m/s	0.55	2.5 hr
1:3	77.92 m/s	1.04	1 hr
1:6	78.32 m/s	1.56	0.5 hr

Table 3-2 Flutter Time History Method Solutions for the Severn Bridge.

As can be seen from the above results the level of refinement in the model has no great influence on the final flutter wind speed prediction. However, it is evident that the more refined models such as the 1:1 and 1:2 idealisations require considerable computational time to perform an analysis on one wind speed.

It was this considerable calculation time that led to the implementation of the modal flutter method, explained in Section 3.2.3.

3.2 Eigenvalue Analysis

3.2.1 Introduction to Eigenvalue Methods

The ANSUSP program in capable of solving two distinct types of eigenvalue problem. The general method used to extract the natural frequencies and modal response of the structure is achieved using the Simultaneous Iteration method, while the Double QR method is used to solve the modal flutter problem. Both the methods implemented in the program have been selected for their suitability to the problem types they have to solve. The main points of both methods are discussed in the following sections.

3.2.2 Natural Frequency Analysis

This analysis is a typical eigensolution problem, which yields the elastic natural circular frequencies of the structure with the corresponding eigenvectors (mode shapes). The Simultaneous Iteration technique[62..65] is a computationally efficient solution process that takes full advantage of the sparsity of the structural stiffness matrix. The basis of this method is the Power Method which is used to identify the dominant eigenvalue of a matrix, which in terms of structural vibration, would be expressed as follows

$$\omega^2 MU = KU$$

Equation 3-10

Here the power method would return the maximum eigenvalue, which would be equal to the square of the maximum natural circular frequency. Since in vibration problems we are most interested in the lowest natural frequency the matrix is manipulated to yield the lowest eigenvalue.

$$MU = \lambda KU$$

Equation 3-11

where $\lambda = \frac{1}{\omega^2}$

In the Simultaneous Iteration method, the algorithm calculates the smallest *m* eigenvalues and corresponding eigenvectors, simultaneously. The solution procedure is

as follows. The stiffness matrix K is decomposed using the Cholesky factorisation method into a lower triangular matrix.

$$K = LL^T$$

Equation 3-12

If the vector Y is defined as shown in, Equation 3-13

$$Y = L^T U$$

Equation 3-13

Then pre-multiplying Equation 3-10 by L^{-1}, we obtain

$$\omega^2 L^{-1} M L^{-T} L^T U = L^{-1} L L^T U$$

Equation 3-14

Thus $$\omega^2 (L^{-1} M L^{-T}) Y = Y$$

Equation 3-15

This finally reduces to the desired form of a typical eigenvalue extraction problem.

$$AY = \Lambda Y$$

Equation 3-16

The practicality of using this method was outlined by Jennings[62,64,65]. This technique is computationally very efficient for large sparsely populated matrices compared with other similar techniques such as the Lanczos Method[66,67]. This method of eigenvalue analysis is the main method used within the ANSUSP program for the calculation of the natural frequency response of the structure. Again this may be done for either the completed structure or the partially completed structure during the construction phase. The resulting orthogonal eigenvectors from this method are mass normalised for subsequent use in other types of analysis.

3.2.3 Modal Flutter Analysis

Modal analysis is based on the principle of superposition, i.e. the response of a structure can be constructed from the natural modes of the structure. The main assumption in this technique is that the structure is linear elastic and that displacements are small. However, a suspension bridge structure especially during construction responds in a highly non-linear manner, thus the accuracy of the modal response prediction may vary. From analyses performed, it was found that the difference in modal and direct integration responses are minimal.

This technique involves the formation of a real skew-symmetric matrix $\overline{A}$ that is reduced to upper Hessenberg form by similarity transforms. The resulting eigenvalues and corresponding eigenvectors are complex in nature.

The problem is solved using the Double QR method which is a computationally more efficient extension of Francis[68] QR method. This method unlike the Simultaneous Iteration method calculates all the eigenvalues within the matrix, thus indicating that this method is only computationally efficient for relatively small order problems ($n \leq 30$).

Since the flutter forcing function resulting from a constant lateral wind is of the form

$$p_a(t) = F_1 u + F_2 \dot{u}$$

Equation 3-17

where F_1, F_2 are aerodynamic stiffness and damping matrices derived from either the Theodorsen's or Scanlan's expressions in Section 2.2. Using the theory of superposition as a co-ordinate transform to convert the response into a set of *N* decoupled equations of motion.

$$u = \Phi\xi$$

Equation 3-18

$$\Phi^T m \Phi \ddot{\xi} + \Phi^T c \Phi \dot{\xi} + \Phi^T k \Phi \xi = \Phi^T F_1 \Phi \xi + \Phi^T F_2 \Phi \dot{\xi}$$

Equation 3-19

Since the eigenvectors ϕ_i calculated in the natural frequency analysis are orthogonal and mass normalised

$$\phi_i^T m \phi_i = 1 \qquad \phi_i^T k \phi_i = \omega_n^2$$

Equation 3-20

The resulting expression is shown in Equation 3-21

$$\ddot{\xi} + \overline{C}\dot{\xi} + \overline{K}\xi = 0$$

Equation 3-21

where

$$\overline{C} = \Phi^T c \Phi - \Phi^T F_2 \Phi$$

$$\overline{K} = \Phi^T k \Phi - \Phi^T F_1 \Phi$$

Assuming a solution of the form $\xi = \xi_0 e^{\lambda t}$ the generalised problem is obtained in the form of the homogeneous equation:

$$\left(\lambda^2 I + \lambda \overline{C} + \overline{K}\right)\xi_0 = 0$$

Equation 3-22

Applying the second co-ordinate transformation

$$\dot{\xi} = \eta$$

Equation 3-23

thus

$$\ddot{\xi} = \dot{\eta}$$

and the generalised problem, Equation 3-21, reduces to

$$\dot{\eta} + \overline{C}\eta + \overline{K}\xi = 0$$

Equation 3-24

Using the co-ordinate transform in Equation 3-25, we note that

$$I\eta = \dot{\xi}$$

Equation 3-25

The problem is formed when the two identities of Equation 3-24 & Equation 3-25 are combined to give the eigenvalue problem, Equation 3-26.

$$\begin{bmatrix} 0 & I \\ -\overline{K} & -\overline{C} \end{bmatrix} \begin{Bmatrix} \xi \\ \eta \end{Bmatrix} = \begin{Bmatrix} \dot{\xi} \\ \dot{\eta} \end{Bmatrix}$$

Equation 3-26

This is analogous to the Equation 3-16, the eigenvalues and eigenvectors for this particular case being in the form of complex conjugates, Equation 3-27.

$$\lambda = \mu \pm i\omega \qquad \xi = p \pm iq$$

Equation 3-27

This form of analysis was employed by Ado-Hamd[69], and is known in the aeronautical industry as the p-k method. The significance of this method is that the magnitude of the real component extracted for each of the natural modes is the real level of global damping in that mode, for the particular combination of wind speed and frequency of excitation.

The converged solution to this problem is obtained when the real part of an eigenvalue reduces to zero while the complex component represents the natural circular frequency of excitation. The resulting eigenvectors are of the form, Equation 3-28, indicating that the eigenvector of the velocity components is $\pi/2$ radian out of phase from the corresponding displacement component.

$$z = \begin{Bmatrix} \xi \\ \lambda\xi \end{Bmatrix}$$

Equation 3-28

This point of instability can be visualised using expression, Equation 3-29, which gives the characteristic motion of the structure in the time domain formed from the complex conjugate components of the eigenvalue solution. The complete derivation of these expressions is shown in Appendix I.

$$\xi(t) = e^{\mu t}\left\{W_1 \cdot \mathrm{Cos}(\omega t) + W_2 \cdot \mathrm{Sin}(\omega t)\right\}$$

Equation 3-29

where

$$W_1 = (q - p) \qquad W_2 = (q + p)$$

The significance of different complex conjugate eigenvalues with respect to general dynamic excitations, are demonstrated graphically (Figure 3-6 to Figure 3-8). However for the flutter stability analysis, the critical solution is when a eigenvalue has a zero real part, signifying the system is undamped, and the corresponding imaginary part of the eigenvalue is equal to the frequency of excitation. This frequency is therefore the flutter frequency at the critical wind velocity, V_{cr}.

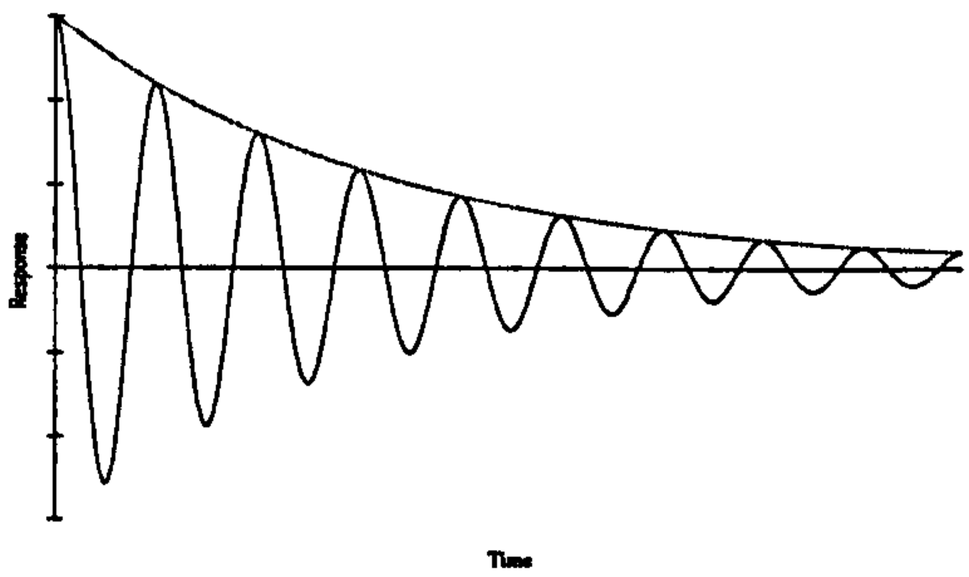


Figure 3-6 λ is complex with -ve real part

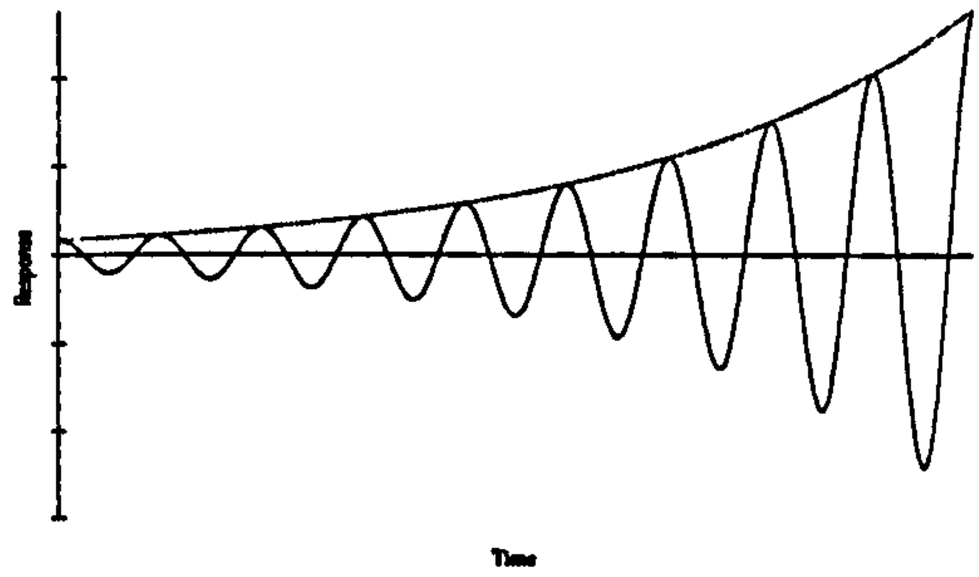


Figure 3-7 λ is complex with +ve real part

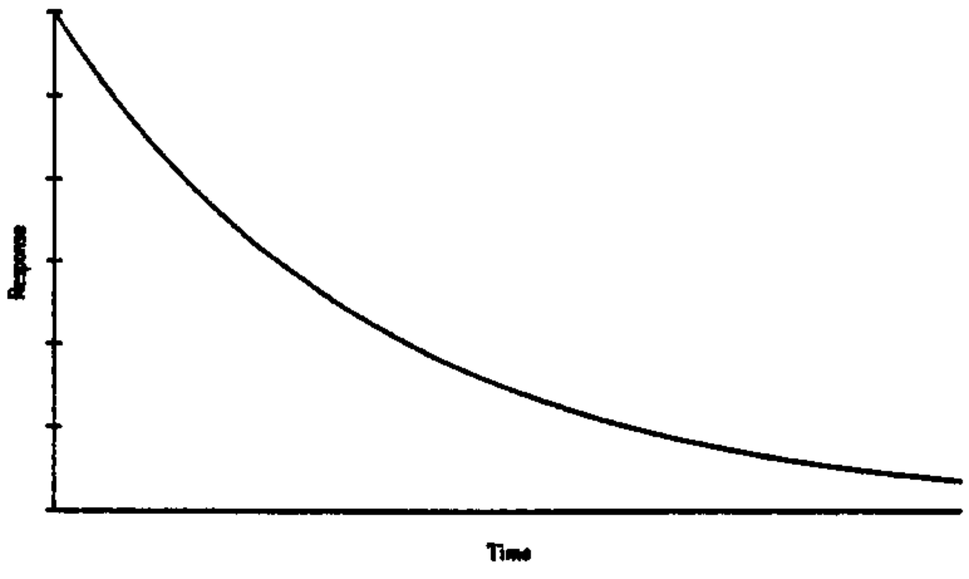


Figure 3-8 λ is real and -ve

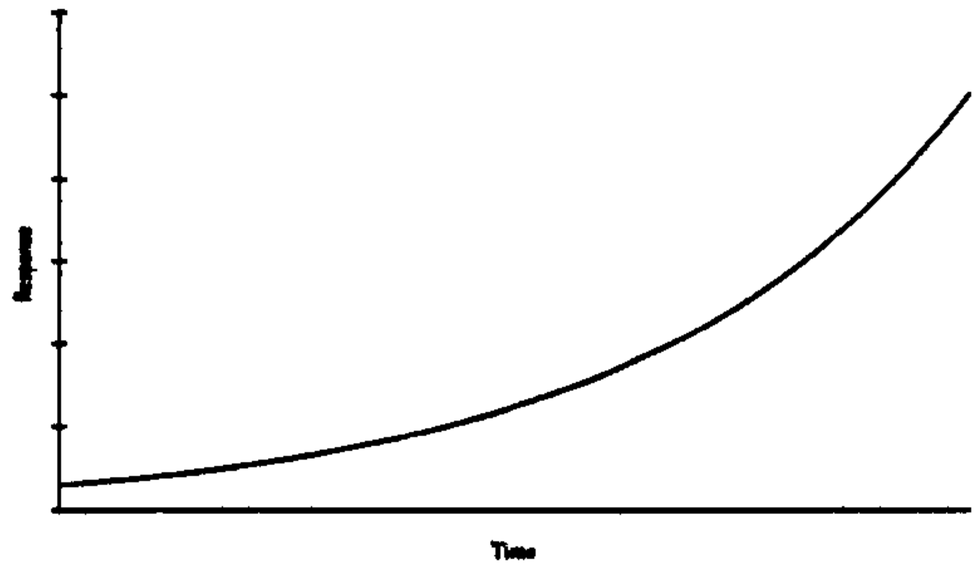


Figure 3-9 λ is real and +ve.

3.2.3.1 Solution Procedure

The solution procedure for this form of analysis has several aspects similar to the flutter time history analysis. The Flutter forcing functions used can be either similar Theodorsen's Expressions (Equation 3-3 & Equation 3-4), or Scanlan's Expressions (Equation 3-5 & Equation 3-6) for experimentally measured flutter derivatives.

This method also requires to know the reduced frequency for each analysis, which is a function of the wind speed V and frequency of oscillation ω.

The solution can be achieved in one of two ways:

i. The user can specify a specific wind speed and frequency of excitation, the solution only being correct if the complex component of one of the eigenvalues is equal to the frequency of excitation and the corresponding real component is zero. This process is repeated for a variety of combinations of wind speed and excitation frequency.

ii. The user can use an automated facility to scan a range of reduced frequencies. This is achieved by specifying a lower and upper limit to both the reduced frequency, k, and frequency of excitation, ω. Once again a converged solution is obtained only when the complex component of one of the eigenvalues is equal to the frequency of excitation and the corresponding real component tends to zero. However, in this case if no accurate solution can be obtained for a specific combination of reduced frequency and excitation frequency, the solution is returned as some arbitrary level of global damping, in this case 2 % critical damping.

The preferred method is the automated method (ii above), which is considerably faster than method (i above), Figure 3-10 & Figure 3-11 below show the results of the Severn bridge 1:2 model idealisation. Figure 3-10 shows the results of a broad range analysis, which has a range of frequency of excitation between the fundamental flexural and torsional frequencies. Figure 3-11 shows the results of the refined analysis, which is a more detailed investigation of the critical range of reduced frequency and frequency of excitation determined from the broad range results.

The critical solution is the one that occurs for the largest reduced frequency at the highest frequency of excitation, thus producing the lowest flutter wind speed prediction.

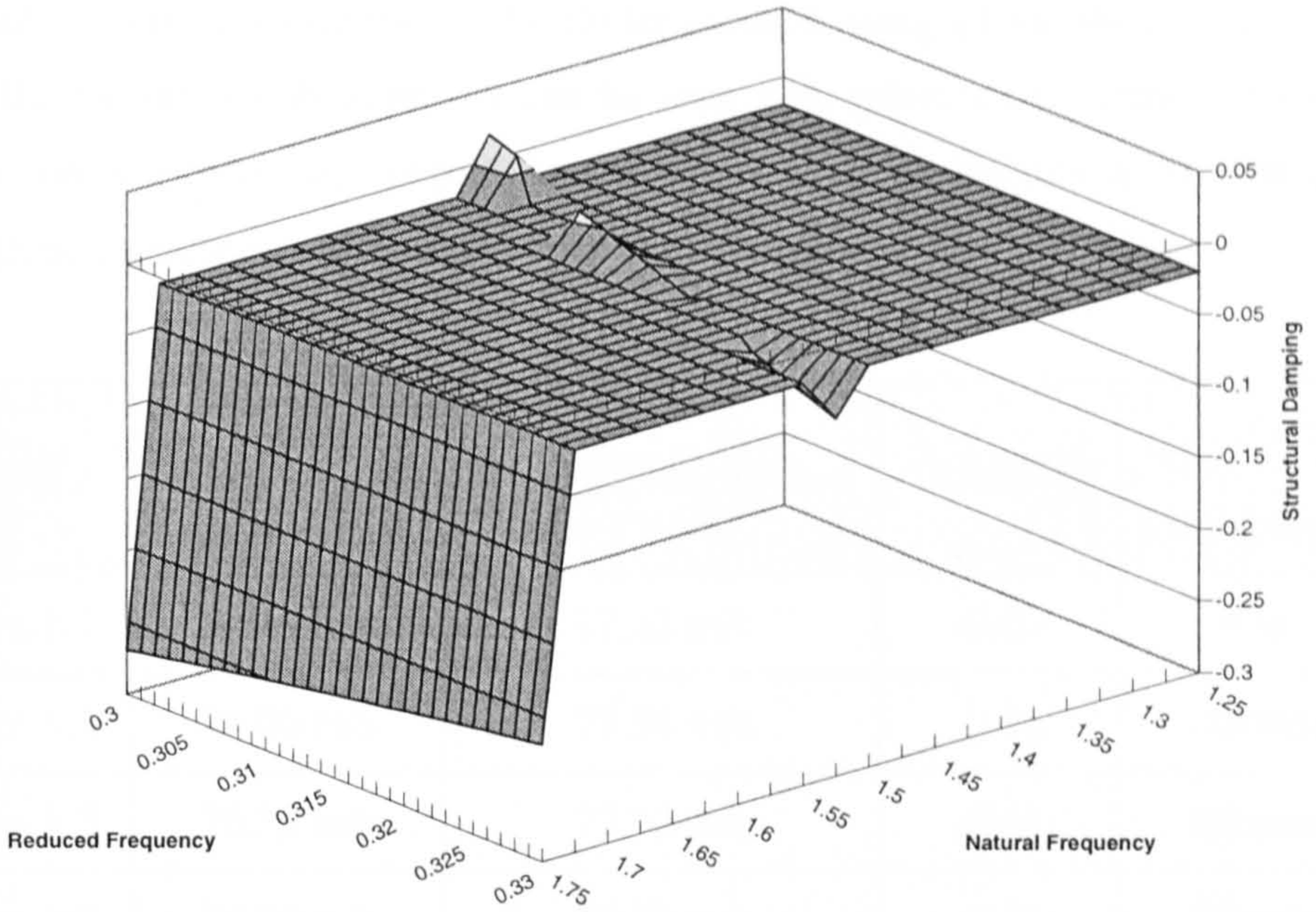


Figure 3-10 Severn 1:2 Idealisation, Broad Range Analysis.

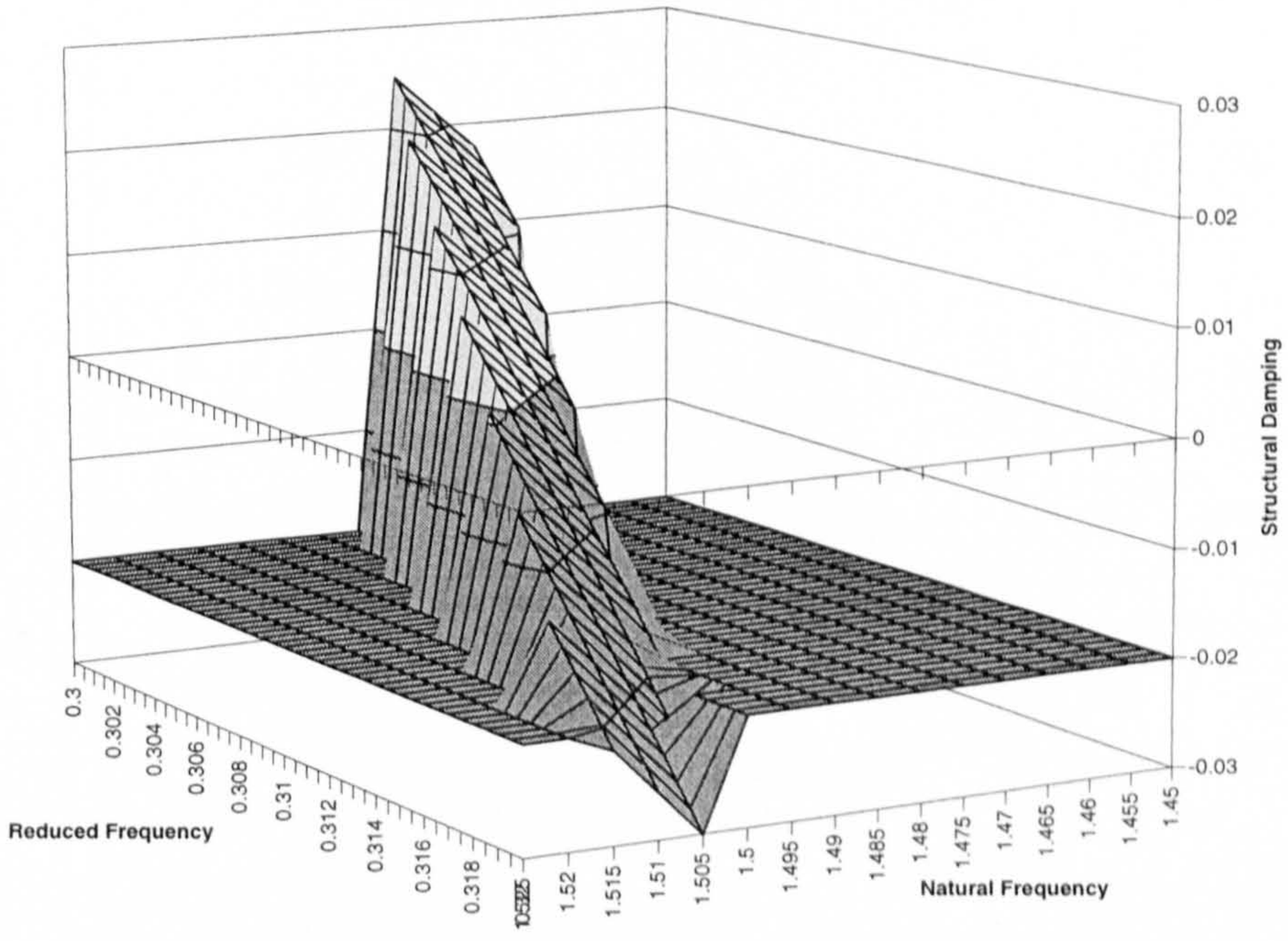


Figure 3-11 The Severn 1:2 Idealisation Refined Analysis.

The results obtained by this method are very similar to those achieved using the numerical integration method in the flutter time history method. Table 3-3, shows the results obtained using the modal flutter method using all the structural modes for each of the various idealisation. As can be seen with reference to Table 3-3 and Table 3-2, the computational time required to complete a single analysis is considerably reduced, with no significant reduction in solution accuracy.

Model	Modal Flutter	Time History Flutter	Error (%)	Analysis Time	
				Time History	Modal
Severn 1:1	77.10 m/s	77.12 m/s	-0.03	4 hr	1 hr
Severn 1:2	77.00 m/s	77.54 m/s	-0.16	150 min	20 min
Severn 1:3	76.80 m/s	77.92 m/s	-0.41	60 min	10 min
Severn 1:6	76.80 m/s	78.32 m/s	-0.41	30 min	1.5 min

Table 3-3 Comparative results for Modal Flutter and Numerical Integration Methods

4 The Theory of Mass Participation

To perform a full dynamic analysis involves determining the responses at each time step for a series of time intervals throughout the motion induced by an arbitrary external excitation. Generally the entire time history of a structure can be solved explicitly using a numerical integration technique. However, to obtain the entire time history of forces and displacements would be considerably time consuming and computationally expensive. A much more economic method of performing a dynamic analysis is the method of modal superposition, as discussed in Section 3.2.3. This method allows the response of the structure to be synthesised from the combination of the significant modes of the structure that each contribute a large amount to the total response.

One method of determining how significant each natural mode is in the total response is the Mass Participation Factor. The Mass Participation Factor calculates the percentage of the total structural mass active in any arbitrary mode shape. The work presented here will develop a method of determining the participation factor for a suspension bridge as well as the discussing the implication of this theory.

The general equation of motion for forced vibration of an *n* degree of freedom system is written in matrix notation as;

$$M\ddot{U} + C\dot{U} + KU = p(t)$$

Equation 4-1

From the theory of linear elastic modal superposition[70,71]; which assumes that the total displacement of a structure can be obtained from the sum of the individual mode shapes.

$$U = \phi_1 Y_1 + \phi_2 Y_2 \cdots + \phi_N Y_N = \sum_{n=1}^{N} \phi_n Y_n$$

Equation 4-2

Which can be expressed in matrix notation as

$$\{U\}=[\Phi]\{Y\}$$

Equation 4-3

Where

U	Total displacement vector.
Φ	Matrix of Eigenvectors.
Y	Generalised co-ordinate vector.

Equation 4-1 may be decoupled into *n* independent equations of motion by pre-multiplying by ϕ_n^T and imposing the mass orthogonality relationship $\phi_n^T m \phi_m = 0$. The result of this manipulation is *n* independent generalised equations of motion of the form

$$M_n \ddot{Y}_n + C_n \dot{Y}_n + K_n Y_n = P_n(t)$$

Equation 4-4

where Y_n is the generalised modal co-ordinate for mode n

$M_n = \phi_n^T M \phi_n$ is the generalised mass

$C_n = \phi_n^T C \phi_n$ is the generalised damping

$K_n = \phi_n^T K \phi_n$ is the generalised stiffness

$P_n = \phi_n^T p(t)$ is the generalised force

If the initial load vector $p(t)$ is defined as

$$p(t) = R \cdot f(t)$$

Equation 4-5

where R is the spatial distribution of the general force.

$f(t)$ is a scalar multiplier defining the magnitude of the temporal forcing function.

Thus the general forcing function for any arbitrary degree of freedom is given by;

$$P_n(t) = \phi_n^T R \cdot f(t)$$

Equation 4-6

It should be noted however, that in this case R, the spatial distribution of the forcing function is constant with time and only the scalar multiplier of the amplitude of the forcing function f(t) is temporal.

For simplicity this function can be written as

$$P_n(t) = \beta_n \cdot f(t) \qquad \text{Where} \qquad \beta_n = \phi_n^T R$$

Equation 4-7

The quantity β_n is the modal excitation factor, which gives a representation of the extent to which the spatial distribution of the forcing function tends to excite response in mode shape ϕ_n.

The first assumption in the following derivation is that the structure is subjected globally to a unit displacement. Thus the total displacement vector for the structure is unity.

$$\{1\} = \Phi Y$$

Equation 4-8

Each modal amplitude can be calculated by pre-multiplying by $\phi_n^T m$ and applying the mass orthogonality relationship $\phi_n^T m \phi_m = 0$.

Thus

$$\phi_n^T m\{1\} = \phi_n^T m \Phi Y$$

Equation 4-9

$$\phi_n^T m\{1\} = M_n Y_n$$

$$Y_n = \frac{\phi_n^T m\{1\}}{M_n}$$

Equation 4-10

re-arranging Equation 4-7 into term of ϕ_n^T and substituting into Equation 4-10

$$Y_n = \frac{\beta_n m\{1\}}{RM_n}$$

Equation 4-11

The total mass of a structure modelled as a lumped mass structure is given by

$$M_t = \langle 1 \rangle m\{1\}$$

Equation 4-12

Substituting Equation 4-8 into Equation 4-12 we obtain the total mass of the structure in terms of the modal matrix and modal amplitudes.

$$M_t = \langle 1 \rangle m \Phi Y$$

Equation 4-13

Substituting expression Equation 4-11 for the modal amplitude of degree of freedom *n* into Equation 4-13

$$M_t = \langle 1 \rangle m \Phi \begin{Bmatrix} \frac{\beta_1 m\{1\}}{RM_1} \\ \vdots \\ \frac{\beta_N m\{1\}}{RM_N} \end{Bmatrix}$$

Equation 4-14

Rearranging (5b) into terms of ϕ_n and substituting into Equation 4-14

$$M_t = \langle 1 \rangle m \left\langle \frac{\beta_1}{R^T} \quad \cdots \quad \frac{\beta_N}{R^T} \right\rangle \begin{Bmatrix} \frac{\beta_1 m\{1\}}{RM_1} \\ \vdots \\ \frac{\beta_N m\{1\}}{RM_N} \end{Bmatrix}$$

Equation 4-15

$$M_t = \langle 1 \rangle m \{1\} \sum_{1}^{N} \frac{\beta_n^2}{R^T m^{-1} R \cdot M_n}$$

Equation 4-16

This expression can be simplified to Equation 4-17, since the expression for the total mass Equation 4-12 can be cancelled from the right hand side of Equation 4-16. Also since all the eigenvectors are ortho-normalised with respect to mass, the generalised mass for each mode is unity.

$$1 = \sum_{1}^{N} \frac{\beta_n^2}{R^T m^{-1} R}$$

Equation 4-17

Defining the Mass Participation Factor χ_n as

$$\chi_n = \frac{\left(\phi_n^T R\right)^2}{R^T m^{-1} R}$$

Equation 4-18

Then the result of such a calculation, χ_n is the percentage of the total structural mass excited by that mode as part of the global structural response when subjected to an arbitrary external force. This expression has several useful features, the main implication of Equation 4-18 is that the sum of all the modal responses or mass participation factors is equal to unity. Thus the possible maximum active mass is the total mass of the structure being modelled. Another useful aspect of this formulation is that the final parameter is non-dimensional which is always a desirable characteristic.

This expression for the mass participation was developed for a general three dimensional spatial distribution for any arbitrary forcing function, and should not be confused with the more widely quoted expressions from Clough[70], Equation 4-19.

$$\chi_n = \frac{\phi_n^T R}{\phi_n^T m \phi_n} \qquad \text{or} \qquad \chi_n = \frac{\phi_n^T mr}{\phi_n^T m \phi_n}$$

Equation 4-19

where

r is the displacement transform vector for a unit displacement at the supports.

The form of the latter expression considers excitations with a spatial distribution in the same plane as the forcing excitation. The expression developed here Equation 4-18 can be applied to any general problem and will give the same solution as Clough for the case of the response being in the same plane as the excitation.

These forms are widely used in earthquake engineering, where it is common to find solutions showing 90-95% of the total mass of a structure being contained within the first six natural modes of the structure. These significant modes generally all have natural frequencies less than 33Hz or 40Hz, which is considered to be the zero period frequency, signifying that the structure responds rigidly with ground vibrations in United States and United Kingdom practices respectively.

4.1 Modal Flutter

In the present research on the dynamic characteristics of suspension bridges the results of modal analysis indicate that for an accurate solution it is necessary to include both the lower frequency modal response of the structure as well as several of the higher modes. However, the natural frequencies of these higher modes are less than 33Hz, which in general dynamic terms is still significant to the solution.

The forcing function resulting from a constant lateral wind is of the form

$$p_a(t) = F_1 u + F_2 \dot{u}$$

Equation 4-20

where F_1, F_2 are aerodynamic stiffness and damping matrices derived from either the Theodorsen's or Scanlan's expressions used for flutter analysis, Equation 2-8 & 2-9 or Equation 2-21 & 2-22 respectively.

Using the theory of superposition as a co-ordinate transform to convert the response into a set of *N* decoupled equations of motion.

$$u = \Phi\xi$$

Equation 4-21

Equation 4-20 is expanded using the superposition expression developed in Equation 4-21 above

$$p_a(t) = F_1\Phi\cdot\zeta(t) + F_2\Phi\cdot\dot{\zeta}(t)$$

Equation 4-22

Looking at each of the two product on the right hand side in detail

$$[F_1]_{(n\,x\,n)}\cdot[\Phi]_{(n\,x\,m)}$$

$$\begin{bmatrix} F_{1_{1,1}} & \cdots & F_{1_{1,n}} \\ \vdots & \ddots & \vdots \\ F_{1_{n,1}} & \cdots & F_{1_{n,m}} \end{bmatrix}\cdot\begin{bmatrix} \phi_{1,1} & \cdots & \phi_{1,m} \\ \vdots & \ddots & \vdots \\ \phi_{n,1} & \cdots & \phi_{n,m} \end{bmatrix} = \begin{bmatrix} \left(\sum_{i=1}^{n} F_{1_{1,i}}\cdot\phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{1_{1,i}}\cdot\phi_{i,m}\right) \\ \vdots & \ddots & \vdots \\ \left(\sum_{i=1}^{n} F_{1_{n,i}}\cdot\phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{1_{n,i}}\cdot\phi_{i,m}\right) \end{bmatrix}$$

Equation 4-23

$$[F_2]_{(n\,x\,n)}\cdot[\Phi]_{(n\,x\,m)}$$

$$\begin{bmatrix} F_{2_{1,1}} & \cdots & F_{2_{1,n}} \\ \vdots & \ddots & \vdots \\ F_{2_{n,1}} & \cdots & F_{2_{n,n}} \end{bmatrix}\cdot\begin{bmatrix} \phi_{1,1} & \cdots & \phi_{1,m} \\ \vdots & \ddots & \vdots \\ \phi_{n,1} & \cdots & \phi_{n,m} \end{bmatrix} = \begin{bmatrix} \left(\sum_{i=1}^{n} F_{2_{1,i}}\cdot\phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{2_{1,i}}\cdot\phi_{i,m}\right) \\ \vdots & \ddots & \vdots \\ \left(\sum_{i=1}^{n} F_{2_{n,i}}\cdot\phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{2_{n,i}}\cdot\phi_{i,m}\right) \end{bmatrix}$$

Equation 4-24

The final formulation is of the form

$$\begin{Bmatrix} p_{a_1}(t) \\ \vdots \\ p_{a_n}(t) \end{Bmatrix} = \begin{bmatrix} \left(\sum_{i=1}^{n} F_{1_{1,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{1_{1,i}} \cdot \phi_{i,m}\right) \\ \vdots & \ddots & \vdots \\ \left(\sum_{i=1}^{n} F_{1_{n,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{1_{n,i}} \cdot \phi_{i,m}\right) \end{bmatrix}_{F_1} \cdot \begin{Bmatrix} \zeta_1(t) \\ \vdots \\ \zeta_m(t) \end{Bmatrix} + \begin{bmatrix} \left(\sum_{i=1}^{n} F_{2_{1,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{2_{1,i}} \cdot \phi_{i,m}\right) \\ \vdots & \ddots & \vdots \\ \left(\sum_{i=1}^{n} F_{2_{n,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{2_{n,i}} \cdot \phi_{i,m}\right) \end{bmatrix}_{F_2} \cdot \begin{Bmatrix} \zeta_1(t) \\ \vdots \\ \zeta_m(t) \end{Bmatrix}$$

Equation 4-25

The final algebraic expression for the flutter aerodynamic loading vector is

$$P_{a_i}(t) = \left\{ \sum_{k=1}^{m} \left(\left(\sum_{j=1}^{n} F_{1_{i,j}} \cdot \phi_{j,i} \right) \cdot \zeta_k(t) \right) \right\} + \left\{ \sum_{k=1}^{m} \left(\left(\sum_{j=1}^{n} F_{2_{i,j}} \cdot \phi_{j,1} \right) \cdot \dot{\zeta}_k(t) \right) \right\}$$

Equation 4-26

Substituting the general displacements and velocities into the previous expression we obtain

$$\begin{Bmatrix} p_{a_1}(t) \\ \vdots \\ p_{a_n}(t) \end{Bmatrix} = \begin{bmatrix} \left(\sum_{i=1}^{n} F_{1_{1,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{1_{1,i}} \cdot \phi_{i,m}\right) \\ \vdots & \ddots & \vdots \\ \left(\sum_{i=1}^{n} F_{1_{n,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{1_{n,i}} \cdot \phi_{i,m}\right) \end{bmatrix}_{F_1} \cdot \begin{Bmatrix} W_{1_1} \cdot \mathrm{Cos}(\omega t) + W_{2_1} \cdot \mathrm{Sin}(\omega t) \\ \vdots \\ W_{1_m} \cdot \mathrm{Cos}(\omega t) + W_{2_m} \cdot \mathrm{Sin}(\omega t) \end{Bmatrix}$$

$$+ \begin{bmatrix} \left(\sum_{i=1}^{n} F_{2_{1,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{2_{1,i}} \cdot \phi_{i,m}\right) \\ \vdots & \ddots & \vdots \\ \left(\sum_{i=1}^{n} F_{2_{n,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{2_{n,i}} \cdot \phi_{i,m}\right) \end{bmatrix}_{F_2} \cdot \begin{Bmatrix} \omega W_{2_1} \cdot \mathrm{Cos}(\omega t) - \omega W_{1_1} \cdot \mathrm{Sin}(\omega t) \\ \vdots \\ \omega W_{2_m} \cdot \mathrm{Cos}(\omega t) - \omega W_{1_m} \cdot \mathrm{Sin}(\omega t) \end{Bmatrix}$$

Equation 4-27

Substituting for the weighting functions W_{1_1}, W_{2_1} gives the following expression.

$$\begin{Bmatrix} p_{a_1}(t) \\ \vdots \\ p_{a_n}(t) \end{Bmatrix} = \begin{bmatrix} \left(\sum_{i=1}^{n} F_{1_{1,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{1_{1,i}} \cdot \phi_{i,m}\right) \\ \vdots & \ddots & \vdots \\ \left(\sum_{i=1}^{n} F_{1_{n,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{1_{n,i}} \cdot \phi_{i,m}\right) \end{bmatrix}_{F_1} \cdot \begin{Bmatrix} [2(ap_1 - bq_1)] \cdot \mathrm{Cos}(\omega t) + [-2(aq_1 + bp_1)] \cdot \mathrm{Sin}(\omega t) \\ \vdots \\ [2(ap_m - bq_m)] \cdot \mathrm{Cos}(\omega t) + [-2(aq_m + bp_m)] \cdot \mathrm{Sin}(\omega t) \end{Bmatrix}$$

$$+ \begin{bmatrix} \left(\sum_{i=1}^{n} F_{2_{1,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{2_{1,i}} \cdot \phi_{i,m}\right) \\ \vdots & \ddots & \vdots \\ \left(\sum_{i=1}^{n} F_{2_{n,i}} \cdot \phi_{i,1}\right) & \cdots & \left(\sum_{i=1}^{n} F_{2_{n,i}} \cdot \phi_{i,m}\right) \end{bmatrix}_{F_2} \cdot \begin{Bmatrix} \omega \cdot [-2(aq_1 + bp_1)] \cdot \mathrm{Cos}(\omega t) - \omega \cdot [2(ap_1 - bq_1)] \cdot \mathrm{Sin}(\omega t) \\ \vdots \\ \omega \cdot [-2(aq_m + bp_m)] \cdot \mathrm{Cos}(\omega t) - \omega \cdot [2(ap_m - bq_m)] \cdot \mathrm{Sin}(\omega t) \end{Bmatrix}$$

Equation 4-28

Upon expanding this expression and taking full advantage of trigonometric expansions we obtain

$$P_{a_i}(t) = \left\{\left(M2_i \cdot b - M1_i \cdot a\right) \cdot \mathrm{Cos}(\omega t) + \left(M1_i \cdot b + M2_i \cdot a\right) \cdot \mathrm{Sin}(\omega t)\right\}$$

Equation 4-29

Where

$$M1_i = \left(C1_i \cdot \omega + C2_i\right) \qquad M2_i = \left(S1_i \cdot \omega + S2_i\right)$$

And

$$C1_i = \left\{\sum_{k=1}^{m}\left(\left(\sum_{j=1}^{n} F_{2_{i,j}} \cdot \phi_{j,k}\right) \cdot q_k\right)\right\}$$

$$C2_i = \left\{-\sum_{k=1}^{m}\left(\left(\sum_{j=1}^{n} F_{1_{i,j}} \cdot \phi_{j,k}\right) \cdot p_k\right)\right\}$$

$$S1_i = \left\{-\sum_{k=1}^{m}\left(\left(\sum_{j=1}^{n} F_{2_{i,j}} \cdot \phi_{j,k}\right) \cdot p_k\right)\right\}$$

$$S2_i = \left\{-\sum_{k=1}^{m}\left(\left(\sum_{j=1}^{n} F_{1_{i,j}} \cdot \phi_{j,k}\right) \cdot q_k\right)\right\}$$

which reduces to the form

$$P_{a_i}(t) = \overline{A}_i \cdot \mathrm{Cos}(\omega t - \vartheta_i)$$

Equation 4-30

where

$$\overline{A}_i = \sqrt{\left(M1_i \cdot b + M2_i \cdot a\right)^2 + \left(M2_i \cdot b - M1_i \cdot a\right)^2}$$

or

$$\overline{A}_i = \sqrt{a^2 + b^2} \cdot \sqrt{M1_i^{\,2} + M2_i^{\,2}}$$

$$\vartheta_i = \mathrm{Tan}^{-1}\left(\frac{(M1_i \cdot b + M2_i \cdot a)}{(M2_i \cdot b - M1_i \cdot a)}\right)$$

Finally this can be expressed in the desired form:

$$P_{a_i}(t) = \left(\sqrt{M1_i^{\,2} + M2_i^{\,2}}\right) \cdot \left(\sqrt{a^2 + b^2} \cdot \mathrm{Cos}(\omega t - \vartheta_i)\right)$$

Equation 4-31

Which is similar to the expression in Equation 4-5.

$$p(t) = R \cdot f(t)$$

This expression has the desired form to determine the spatial distribution of the forcing function and the temporal function. The significance of the developed solution is that the arbitrary constants of integration a & b, (refer to Appendix I), introduced in the development of the characteristic motion of the structure, can be separated from the spatial distribution, thus allowing the above expression to be used to calculate mass participation factors. If this decoupling had not been achieved, the solutions obtained would have been directly related to the values of these integration constants, resulting in an initial value problem.

4.2 Modal Flutter Mass Participation Results

Several detailed analyses have been performed using the Severn Bridge model with different structural idealisations. Table 4-1 to Table 4-4 show the significant numerical results obtained from the 1:1, 1:2, 1:3 and 1:6 numerical idealisations respectively. The modal description S and C indicated that modes are predominately side-span mode or centre-span mode respectively. Figure 4-1 and Figure 4-2, show the distribution of mass participation and cumulative mass participation respectively for the 1:3 idealisation.

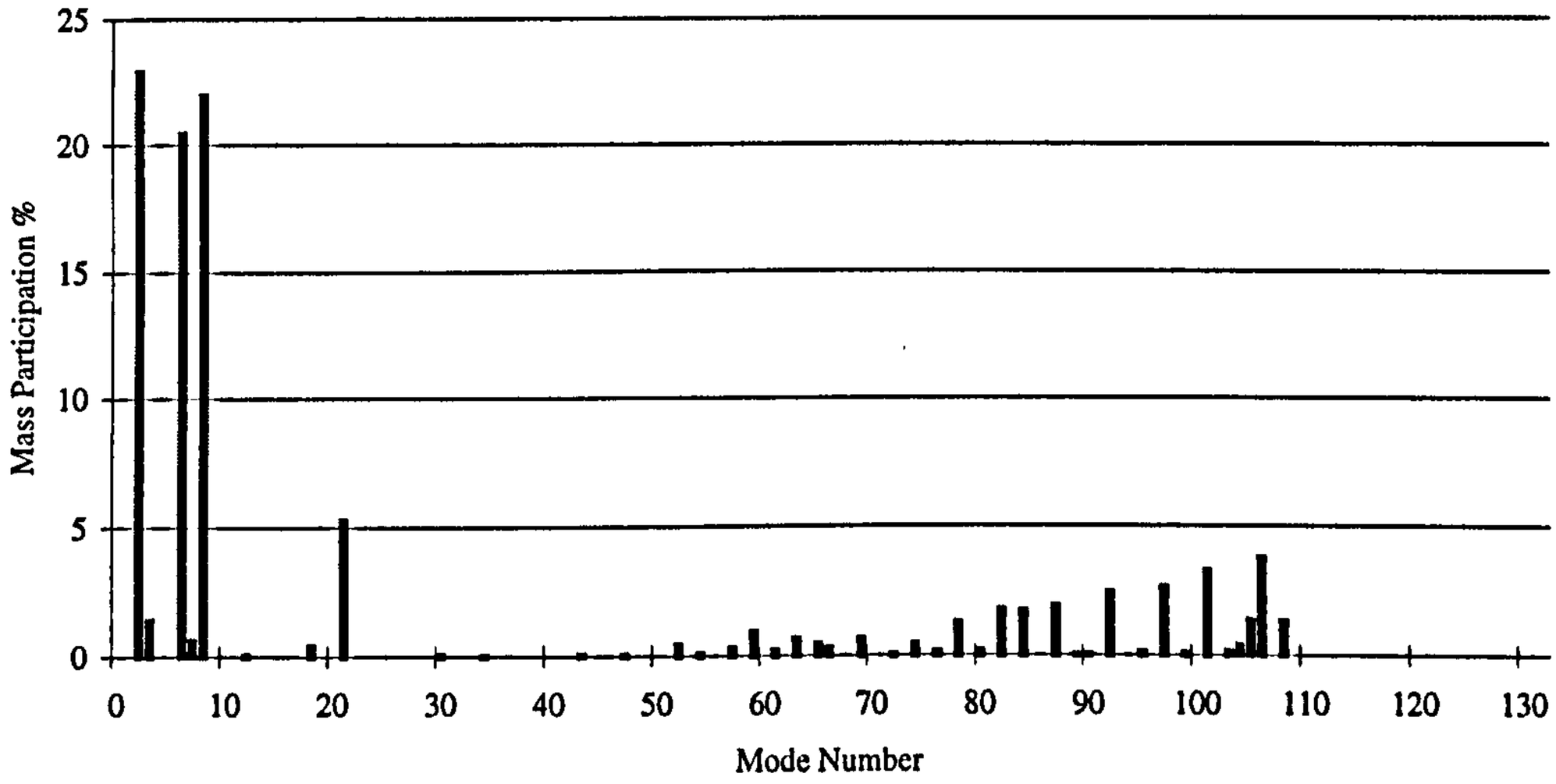


Figure 4-1 Modal Participation factors for Severn Bridge (1:3 Idealisation).

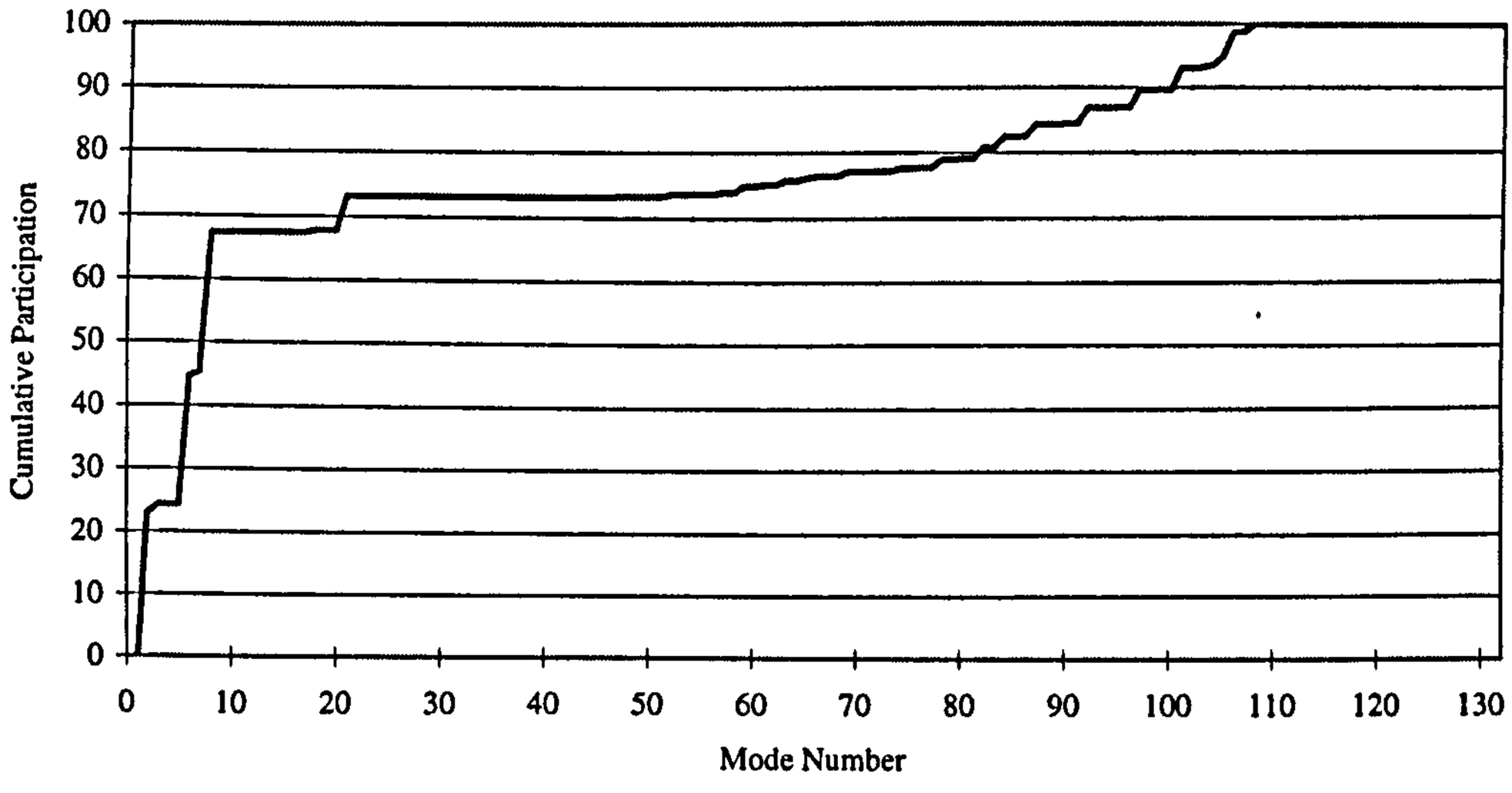


Figure 4-2 Cumulative Mass Participation for Severn Bridge (1:3 Idealisation).

By inspection the modal participation factors for the four most significant natural modes for all analyses are similar in magnitude and correspond to the same mode shape. The only exception to this being the 1:6 idealisation where the fourth most significant mode is displaced by the 59th natural mode to fifth position. This shift in order is due to the smaller number of natural modes that can be extracted from the 1:6 idealisation. The significance of this mode reduces with improvement in the mesh refinement. The cumulative total participation of these modes being around 70 % ± 1 %, with the corresponding flutter wind speed being predicted around 77.1 m/s±0.3%.

Mode *n*	χ_n (%)	F_n (Hz)	Modal Description
2	22.509	0.1377	1 st Symmetric Flexural (S,C)
8	22.174	0.3604	1 st Symmetric Torsional (C)
6	20.032	0.2732	3 rd Symmetric Flexural (S,C)
21	5.164	0.9263	1 st Symmetric Torsional (S)

Table 4-1 Results for Severn Bridge (1:1 Idealisation) Vcr = 77.10 m/s

Mode *n*	χ_n (%)	F_n (Hz)	Modal Description
2	22.815	0.1389	1 st Symmetric Flexural (S,C)
8	22.099	0.3615	1 st Symmetric Torsional (C)
6	20.261	0.2741	3 rd Symmetric Flexural (S,C)
21	5.222	0.9289	1 st Symmetric Torsional (S)

Table 4-2 Results for Severn Bridge (1:2 Idealisation) Vcr = 77.00 m/s

Note:- The modal description S and C indicated that modes are predominately side-span mode or centre-span mode respectively

Mode *n*	χ_n (%)	F_n (Hz)	Modal Description
2	22.902	0.1399	1 st Symmetric Flexural (S,C)
8	21.969	0.3623	1 st Symmetric Torsional (C)
6	20.446	0.2738	3 rd Symmetric Flexural (S,C)
21	5.280	0. 9217	1 st Symmetric Torsional (S)

Table 4-3 Results for Severn Bridge (1:3 Idealisation) Vcr = 76.80 m/s

Mode *n*	χ_n (%)	F_n (Hz)	Modal Description
2	22.080	0.1411	1 st Symmetric Flexural (S,C)
6	21.497	0.2676	3 rd Symmetric Flexural (S,C)
8	21.261	0.3604	1 st Symmetric Torsional (C)
59	6.114	8.4396	5 th Symmetric Torsional (C)
17	5.723	0.8950	1 st Symmetric Torsional (S)

Table 4-4 Results for Severn Bridge (1:6 Idealisation) Vcr = 76.80 m/s

In a practical engineering design situation this method presents a significant gain over the standard numerical integration methods. Table 4-5 to Table 4-7, show the solutions for the Severn bridge using both the numerical integration method and the Modal Method, Table 4-6 show the results of the modal method using all the structural mode, while Table 4-7 shows the solutions for the four most significant modes being used. The error associated with the modal method is negligibly small,

Idealisation	V_{cr}	Error (%)	Analysis Time
1:1	77.12 m/s	-	4 hr
1:2	77.54 m/s	0.55	2.5 hr
1:3	77.92 m/s	1.04	1 hr
1:6	78.32 m/s	1.56	0.5 hr

Table 4-5 Flutter Predictions Using Time History Method.

Idealisation	V_{cr}(All modes)	Error (%)	Analysis Time
1:1	77.10 m/s	-0.03	1 hr
1:2	77.00 m/s	-0.16	20 min
1:3	76.80 m/s	-0.41	10 min
1:6	76.80 m/s	-0.41	1.5 min

Table 4-6 Flutter Predictions Using Modal Method (All Modes).

Idealisation	V_{cr}(4 modes)	Error (%)	Analysis Time
1:1	77.11 m/s	-0.03	20 sec
1:2	76.97 m/s	-0.19	12 sec
1:3	76.83 m/s	-0.38	6 sec
1:6	76.90 m/s	-0.29	4 sec

Table 4-7 Flutter Predictions Using Modal Method (4 Significant Modes).

The most significant point to note when examining the results, is how considerably faster the modal technique is compared to the standard numerical integration method. The solution is achieved on average 630 times faster, if only the four most significant modes are used when compared to numerical integration, while the four mode method is also around 100 times faster that using all the structural modes. In practice, this allows a design engineer to examine more potential structural configurations at a preliminary stage.

5 SUSPview Graphic Interface (SVGI)

The ANSUSP package allows an analyst to solve several different types of arbitrary loading, as described in Chapter 3. The majority of these analyses employ a general numerical integration technique. As a consequence of using this form of solution a vast amount of computational results are generated for each time-step at considerable computational expense. The second analytical method used within ANSUSP is an eigenvalue method, used to extract the natural frequencies of the structure. This method like the numerical integration method generates a considerable volume of results that are complicated to interpret readily.

In an attempt to alleviate this situation, a comprehensive graphic interface has been developed to allow the analyst to animate the response of the structure, subjected to any arbitrary form of loading, from which a greater understanding of the structure's behaviour under that loading can be obtained.

The SUSPview Graphic interface is capable of viewing and interpreting the results of various types of analysis. Figure 5-1, shows the menu options available. The main forms being time history analysis, natural frequency mode shapes and modal flutter analysis.

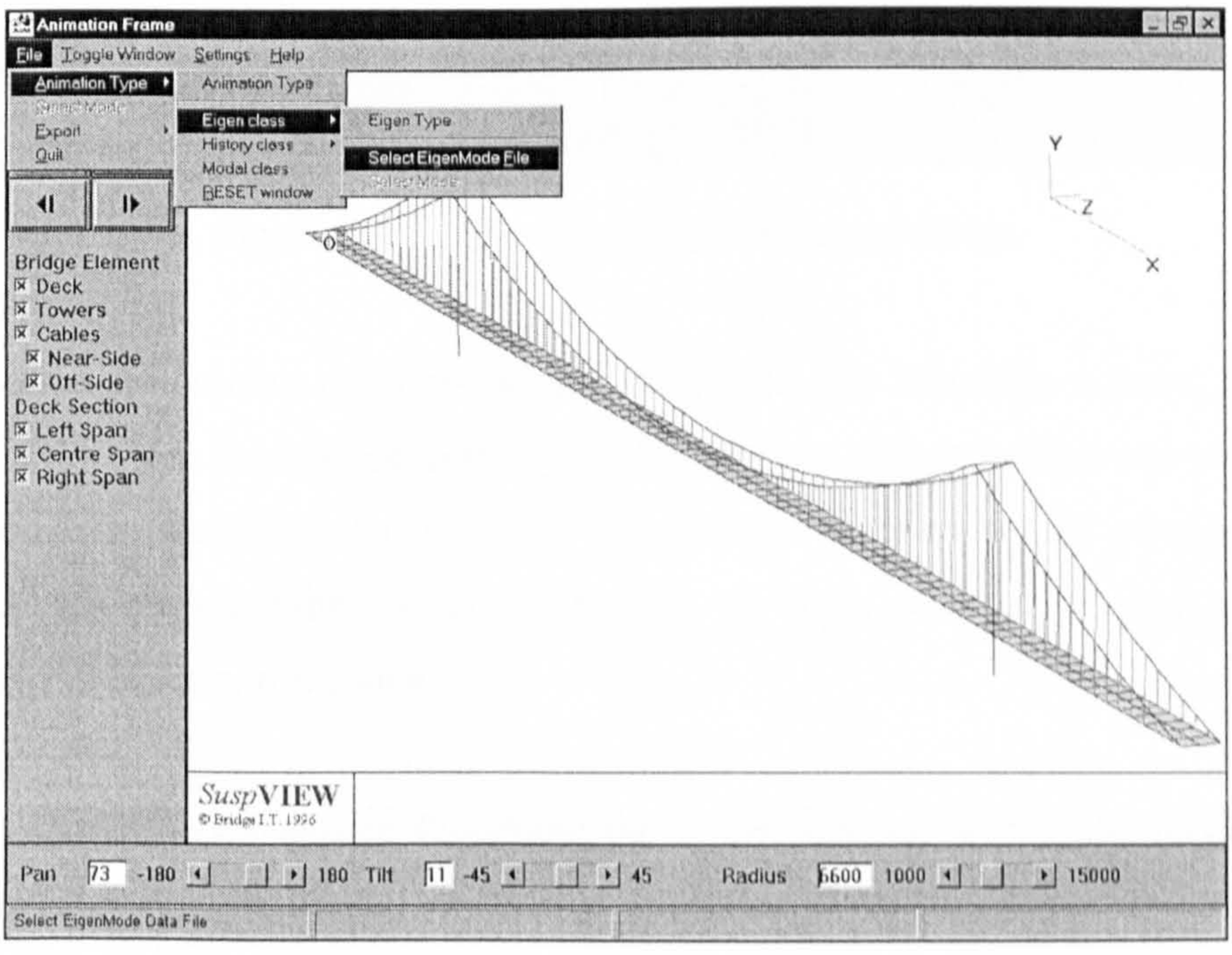


Figure 5-1 The SUSPview Graphical User Interface.

The SUSPview interface was initially developed to view the calculated eigenmodes, Figure 5-2, of the suspension bridge structure. The capabilities were increased to allow the calculated eigenmodes to be animated in pseudo time Figure 5-3 & Figure 5-4. This in itself was very useful, allowing the automatic determination and characterisation of the various natural modes, a task that could only be previously achieved by examining the considerable numerical printout and drawing the eigenmode manually, a very laborious and time consuming task.

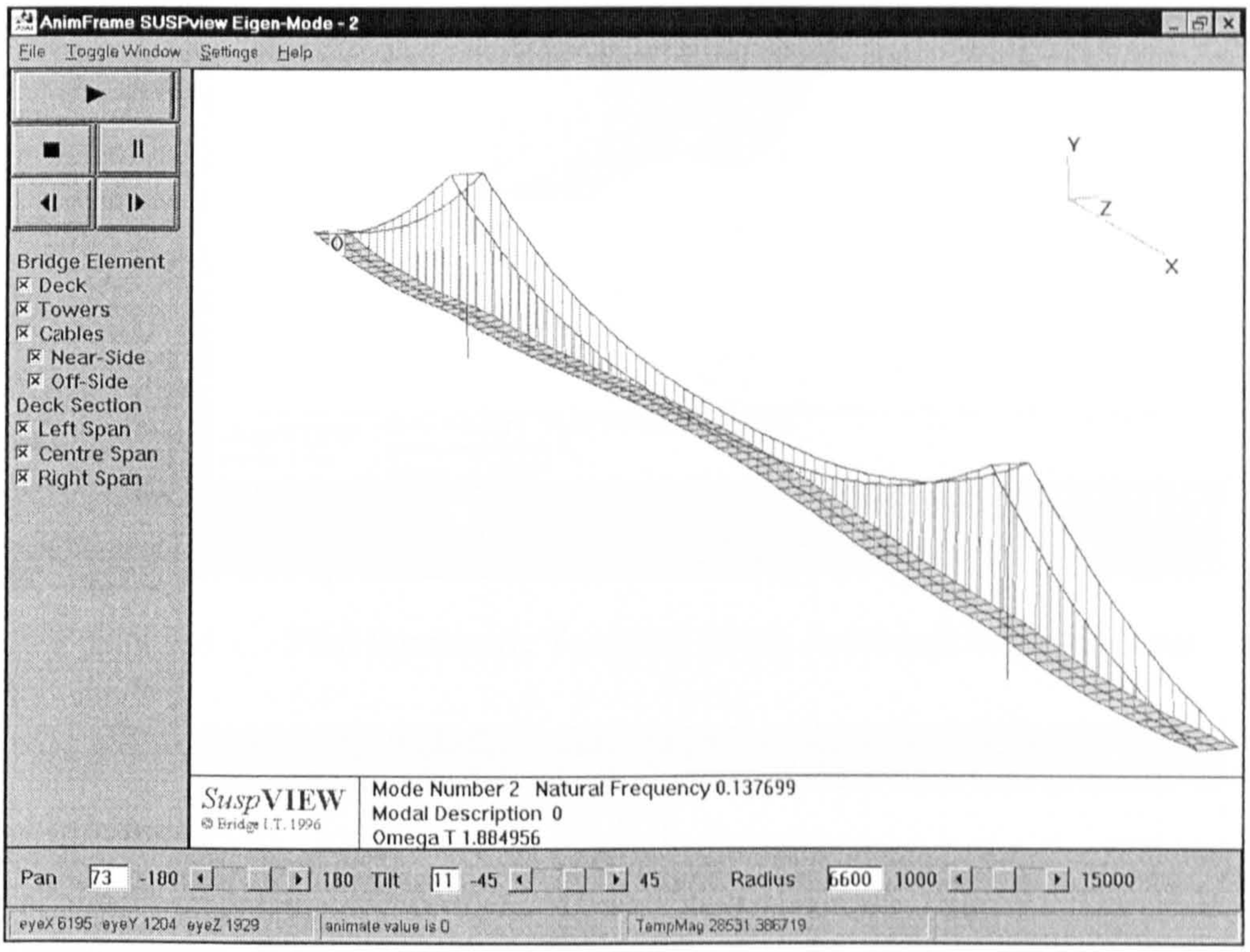


Figure 5-2 First Symmetric Flexural Mode.

To increase the amount of information gained from the interface, a basic form of colour coding was introduced, initially to the hangers, then extended into the cable elements. The interface calculates the strain state in each of the elements and displays the corresponding elements in one of three colours denoting tension, compression or zero strain state.

Thus the analyst can readily determine the stress state of all the elements in the structure. This function is also active during animation allowing a force/strain history

of any element to be easily obtained, which is of interest within the context of the fatigue life of a structural element.

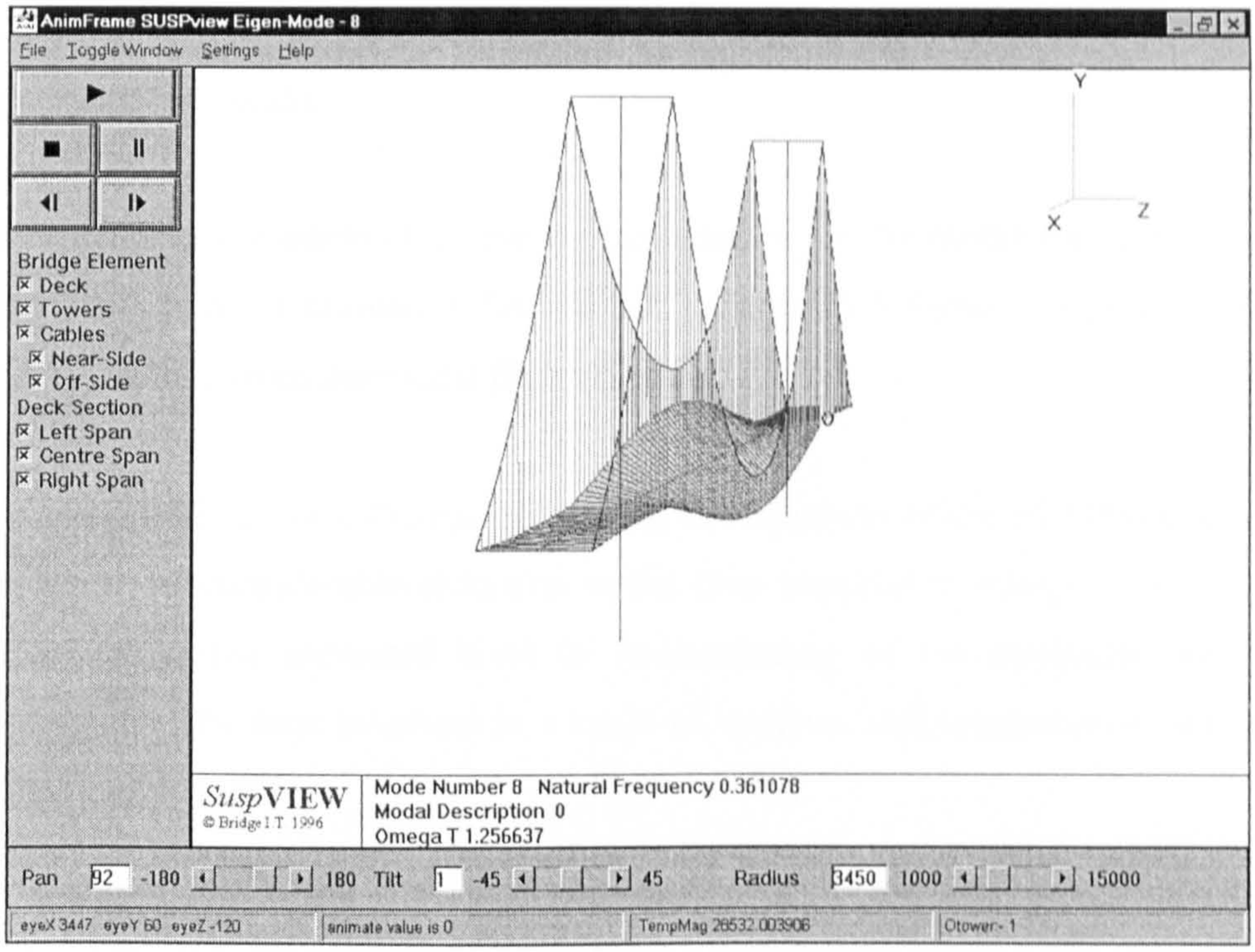


Figure 5-3 First Symmetric Torsional Mode Animated in pseudo time.

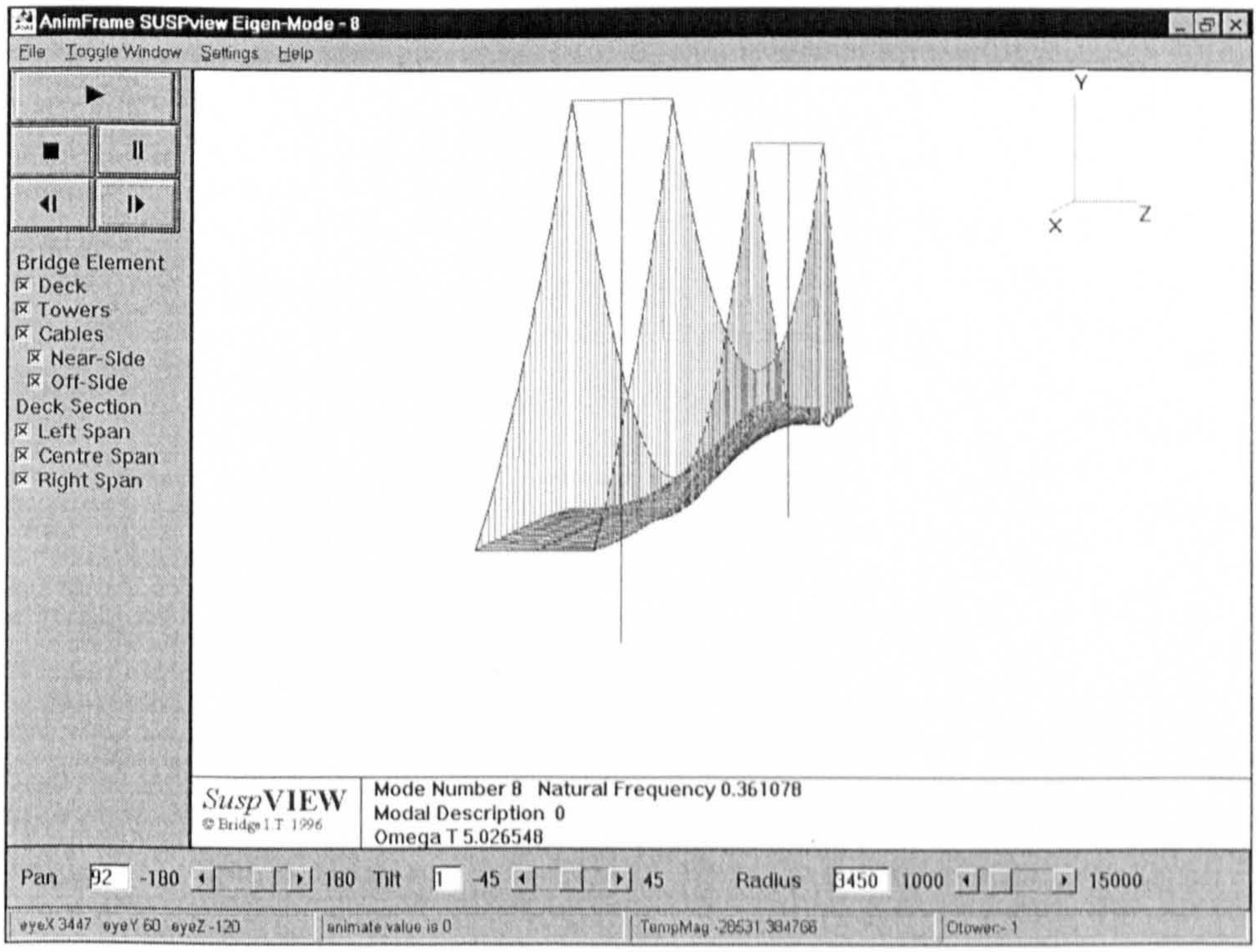


Figure 5-4 First Symmetric Torsional Mode Animated in pseudo time.

The interface also has the capability of animating the structure, when subjected to any form of loading when analysed using the numerical integration technique. The analyst has the option of scrolling forward through the response, or if required, stop the animation and go backwards. Presently this is only available at a much exaggerated time scale.

SUSPview is also capable of animating the solution for the modal flutter method of flutter analysis, as mentioned in Section 3.2.3. Figure 5-5 shows a typical animated solution obtained from the modal flutter method.

The most significant benefit arising from the development of the SUSPview graphic interface is the considerable reduction in the time required to interpret an analysis, with an associated increased level of understanding of the structural response. Consequently, the time involved in a cycle of analysis and interpretation has been considerably reduced.

Another significant benefit for the analyst is the reduction in the need to examine page after page of printout, possibly to a level where for an approximate analysis, the need to print the results has been totally removed.

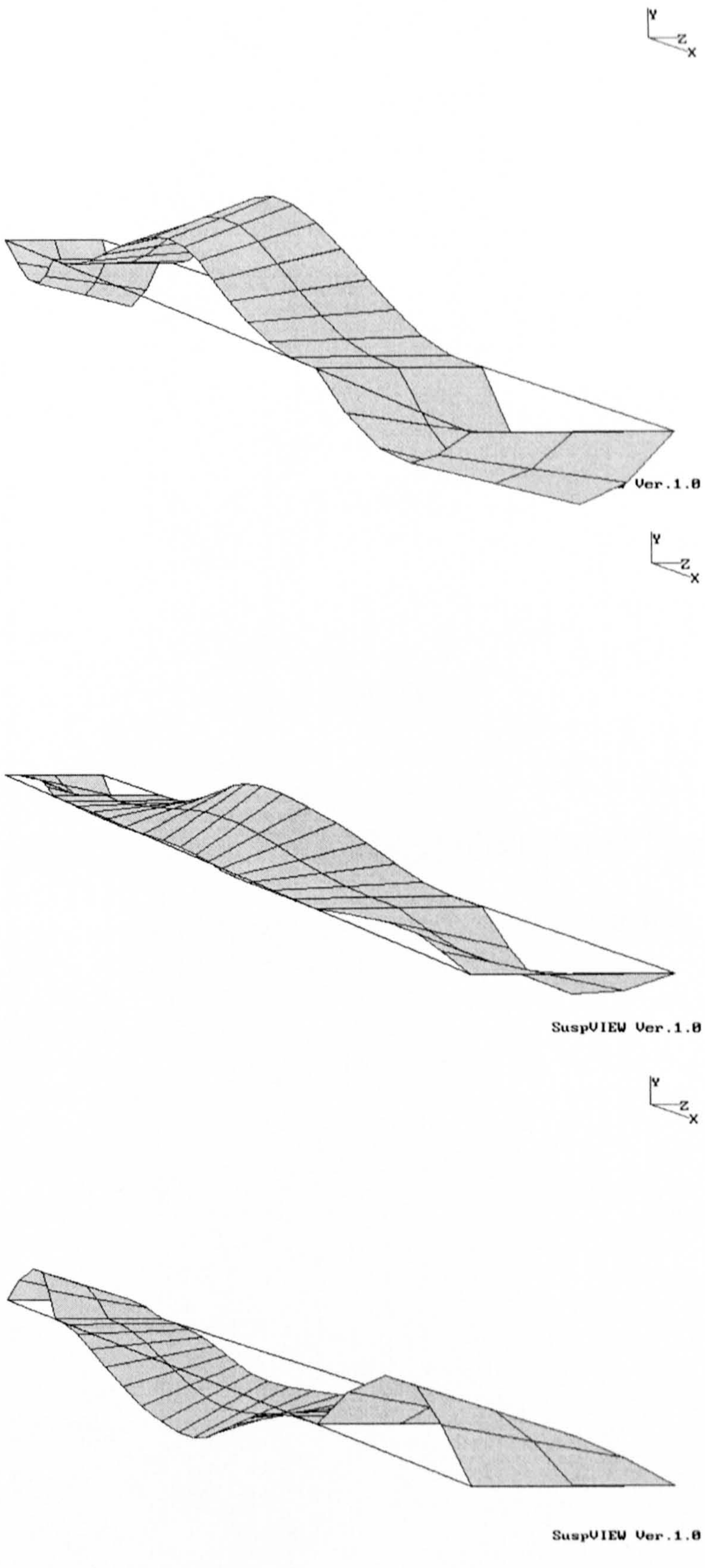


Figure 5-5 Animated Modal Flutter Solution

PART II

The Application of Distributed Computer Technology to Civil Engineering Aerodynamics

6 Introduction to Distributed Computing

Since the development of Colossus, the computer that broke the Enigma code in World War II, the development of computer technology has been inextricably linked with the military need for progressively faster computers.

In the sixties and seventies, supercomputer companies like CRAY research started constructing systems that would be capable of solving formidable scientific and engineering problems that extensively used floating point calculations. The intense computational demand of this type of work, much of it involving nuclear research or aerospace design was motivated by the arms race. This combined with the necessity of obtaining the result as quickly as possible, meant that cost was no object in obtaining the fastest hardware possible. The first CRAY-1 supercomputer shipped to the Los Alamos National Laboratory in 1976, had a peak speed of 167 MFLOPS (Millions of FLOating Point operations per Second) and cost upwards of $4 Million.

More recently in 1995 the U.S. Department of Energy paid $45 Million for the first super computer capable of sustaining 1 TFLOP (1 TFLOP equals 1 Million MFLOPS), to model the effects of ageing on nuclear warheads. The system designed and constructed by Intel contained 9072 processors each with 32 MB of dedicated RAM per processor.

The development of these cutting edge machines, has of course pushed back the frontiers of knowledge and led to the development of less expensive parallel computer systems. The main benefit of a parallel computer compared to a serial computer is that the parallel machine contains several processors that each performs part of the calculation given to that system.

This may be done in one of two ways. In the first, the parallel computer companies have developed several transparent operating systems that will implicitly parallelise a serial program run on a parallel computer. This can be done using PVM[72] or LSF[73], both of which allow a network of serial machines to be linked to form a 'virtual parallel machine', the results being the same as those if run on a single machine, but with the benefit of being performed significantly faster.

The second method is to explicitly parallelise the serial program codes. There are numerous languages that can be used for this such as High Performance FORTRAN[74] or MPI[75] which are both parallel extension of FORTRAN 77.

Along with the considerable effort devoted to the development of faster computers, considerable research has been devoted to the development of faster algorithms, that can perform a specific calculation faster and in a more numerically stable manner. Along with the development of the algorithms several communication topologies have been identified to allow data to pass around the parallel processor network with the minimum of delay.

Considerable research has been devoted to the development of general parallel matrix techniques, such as Gaussian Elimination, Cholesky Factorisation and General matrix reductions (Tridiagonal and Hessenberg forms), refer to [76..94].

In particular attention has been devoted to the development of efficient parallel eigensolution algorithms. The most prominent research being undertaken at the Oak Ridge National Laboratory, Tennessee led by Dongarra[95..97], into the parallelisation of the QR eigensolution method, with particular reference to solving the non-symmetric real matrix problem. However, research has been undertaken elsewhere into the Jacobi and Subspace Iteration Methods [98-109].

The objective of the research undertaken in this project was to implement the existing serial version of ANSUSP onto a distributed computer system. The following Sections will discuss the work undertaken in identifying and developing suitable communication topologies for the various algorithms within the program, as well as the development of specific solution techniques for the Numerical Integration and Simultaneous Iteration Algorithms.

7 Fundamentals of Distributed Computing

There are several classes of computer system available. They range from the common desktop portable computer to the commercial main frame computer, with several intermediate levels. The main distinction between these various computers is the hardware structure of the computer system. The general PC computer is a simple sequential computer that solves problems in a sequential manner, one step at a time, instruction after instruction.

Flynn[110] defined four main classes of computer systems, as shown in Table 7-1. The simplest computer is the Single Instruction Single Data (SISD) that obtains one instruction to process at a time and is equivalent to a PC. The next level of complexity being the Single Instruction Multiple Data (SIMD) class of system which includes Vector computers. These systems perform their calculations by assigning a controlling processor that instructs the rest of the processor within the system on how to perform their assigned task.

Single Instruction Single Data (SISD)	Single Instruction Multiple Data (SIMD)
Multiple Instruction Single Data (MISD)	Multiple Instruction Multiple Data (MIMD)

Table 7-1 Computer System Classification according to Flynn[110]

According to Flynn the most interesting class is that of the Multiple Instruction Multiple Data (MIMD) systems, which itself has two sub-classes of Distributed Global Memory and Shared Global Memory.

The most significant difference between these sub-classes is that the distributed global memory computer system requires some form of communication capabilities between the numerous processors of the system, while the shared memory system requires no such inter-communication. The two contrasting systems are shown schematically in Figure 7-1 & Figure 7-2 respectively.

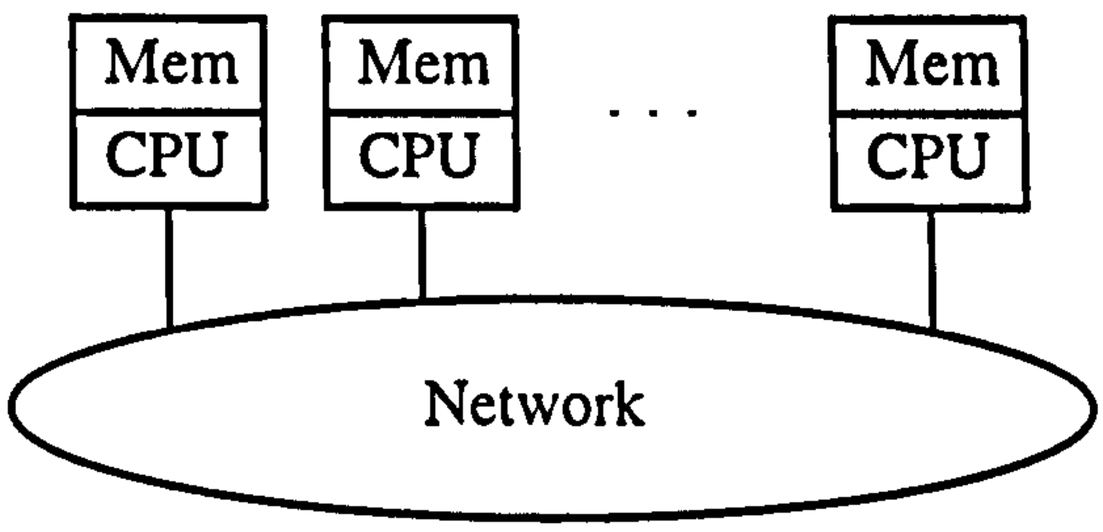


Figure 7-1 MIMD computer system with distributed memory.

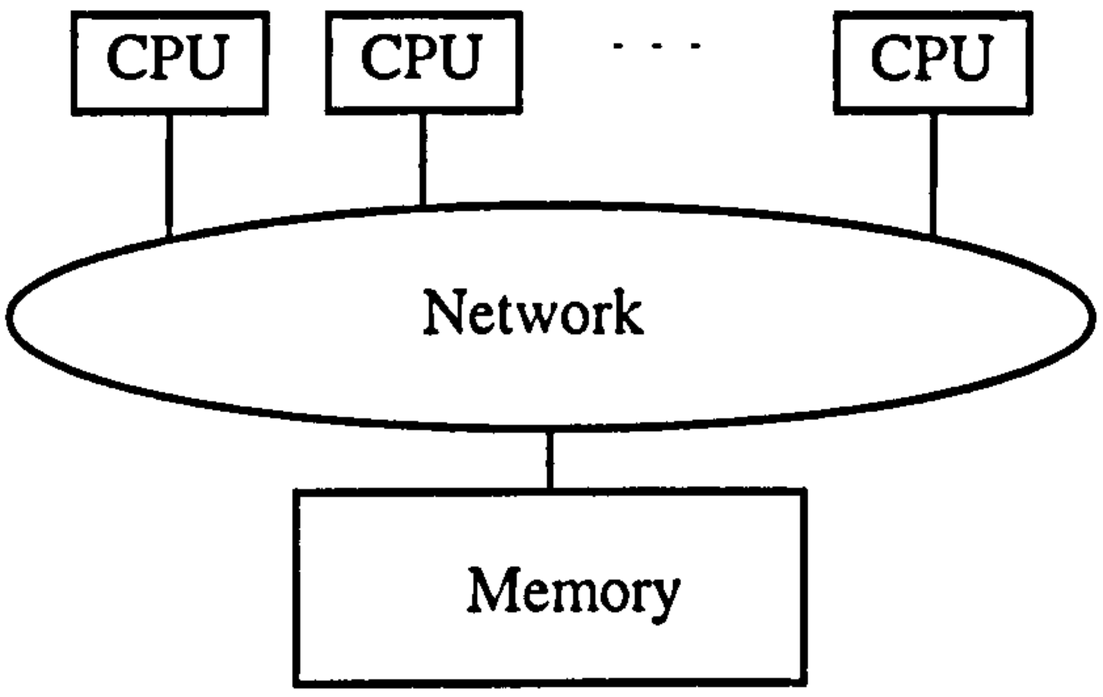


Figure 7-2 MIMD computer system with shared memory.

The distributed memory computer is constructed from a network of nodes that consist of a Central Processing Unit (CPU) with local Random Access Memory (RAM). The network is held together by physical communication links between adjacent processors, generally due to the physical two-dimensional construction of these computer systems, each processor has four physical communication links as shown in Figure 7-3.

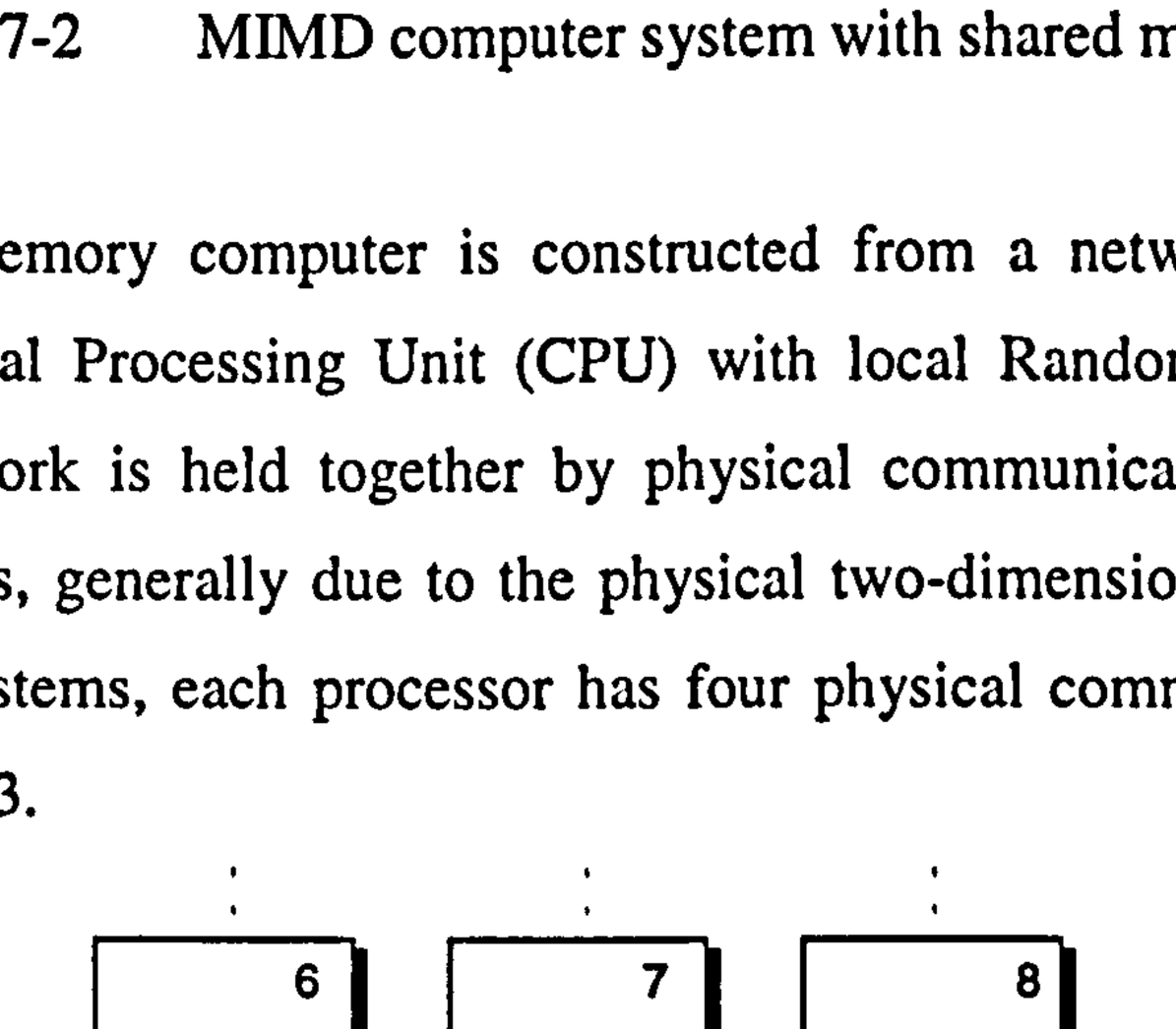


Figure 7-3 The physical communication links for a two-dimensional processor network.

Adjacent processors can communicate with each other directly using their physical communication link. While distant processors communicate with each other by forming what is termed a 'virtual link'. This is a link that is formed using the physical links of all the adjacent processors between the two communicating processors, as shown in Figure 7-4. The communication between processors 0 – 8, can be achieved by adjacent communications for 0 – 1 – 2 – 5 – 8. However, it could also be achieved by communicating from 0 – 1 – 4 – 7 – 8, 0 – 1 – 4 – 5 – 8, 0 – 3 – 4 – 5 – 8, 0 – 3 – 4 – 7 – 8 or 0 – 3 – 6 – 7 – 8. Generally however, the computer dynamically routes these communications, having the advantage that the system can respond to any changes in the network traffic patterns to help avoid congestion and minimise any communication delays.

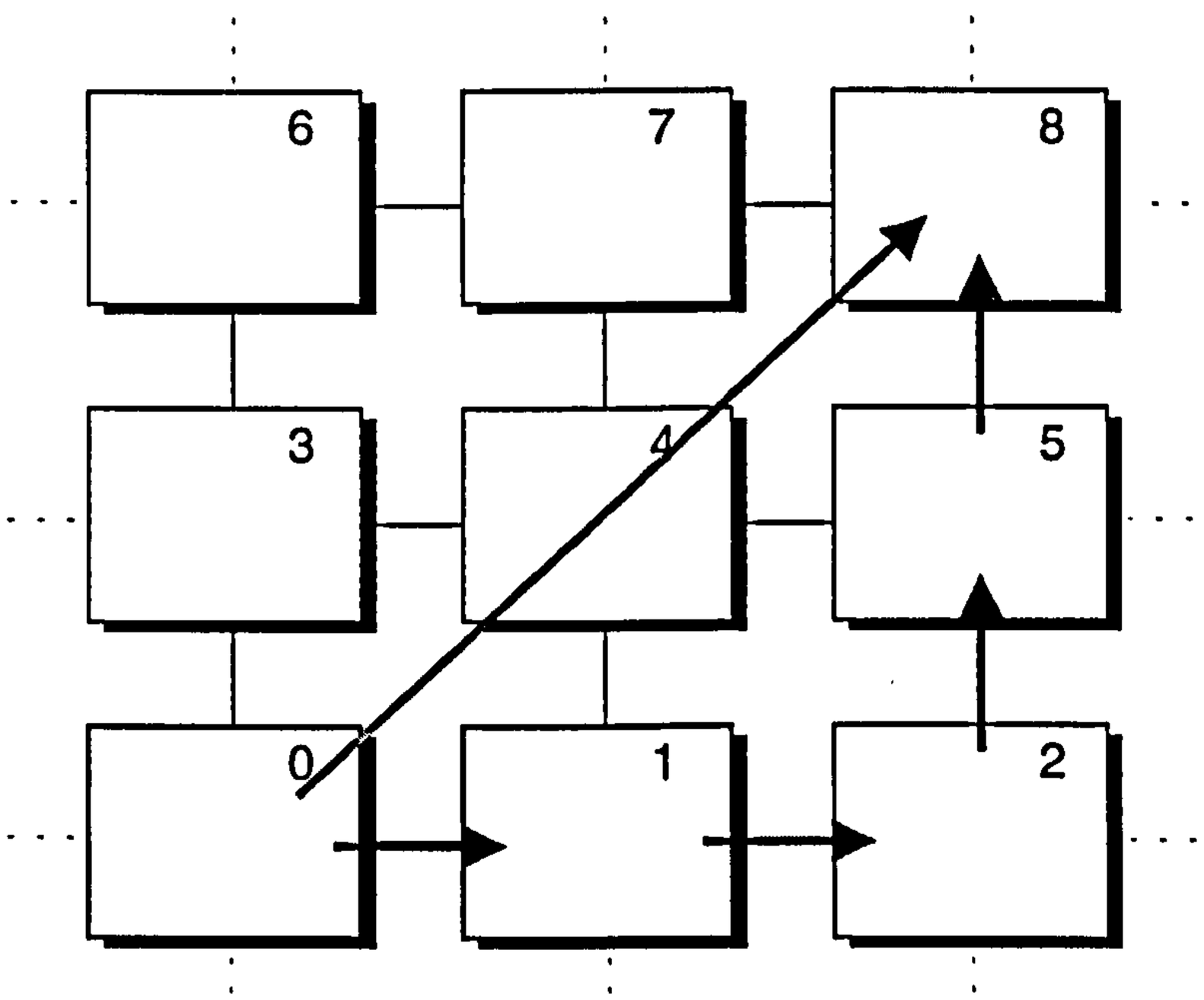


Figure 7-4 The physical communication links for a two-dimensional processor network.

Communicating around the network takes time and great care is required to ensure that any potential for delaying the execution of a set of instructions is minimised. As would be expected, the greater the distance between communicating processors the greater the communication time required, possibly due to the communication being routed along virtual links. Any communication between processors consists of two parts:

a) The communication overhead, which is the time required to prepare the network for the forthcoming communication.
b) The communication rate, which is the time required to transmit a single Byte of data between the master and slave processor.

An additional cause of delays within the communication network is the formation of communication bottlenecks. These occur when several slave processors try to simultaneously communicate with the master processor. Figure 7-5, shows the formation of a communication bottleneck when three processors try simultaneously to communicate with another processor. The result is that the network would recover the solution from processor 1, while delaying processors 2 and 3. They would both be delayed for the duration of the communication from processor 1. When this communication is complete the next processor in sequence will communicate its solution.

This cycle would be repeated several times until all the processors have communicated their solutions. The maximum delay being experienced by any single processor being equal to the sum of all the individual communications from all the preceding processors.

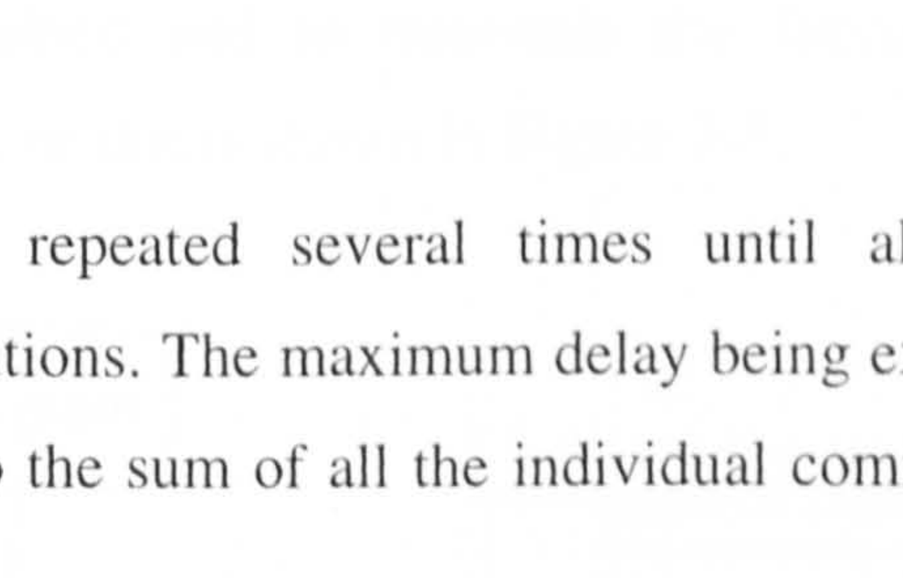


Figure 7-5 The Formation of a Communication Bottleneck

Thus the implication of a communication bottleneck forming within a parallelised computer program would be the development of considerable delay within the network, making the scheduling of computation within the network very difficult and thus significantly reducing the overall efficiency of the parallelised codes.

Distributed computer systems have the capability of controlling the execution of specific calculations in a pre-defined way. This is achieved due to the communication capabilities of the distributed memory network. The system can be programmed to mask a specific section of the program to run independently on specific processors.

This capability can be utilised to solve two specific algorithm topologies. Firstly, when the main program calls a subroutine, which then performs the same calculation N times for a series of elements. It is possible to distribute the elements to n different processors, each processor performing N/n calculations concurrently, each processor passing the solutions back to the master processor upon completion.

This requires a close control on message passing between processors, to ensure the correct solution is obtained and to minimise the formation of communication bottlenecks. A schematic of this is shown in Figure 7-5.

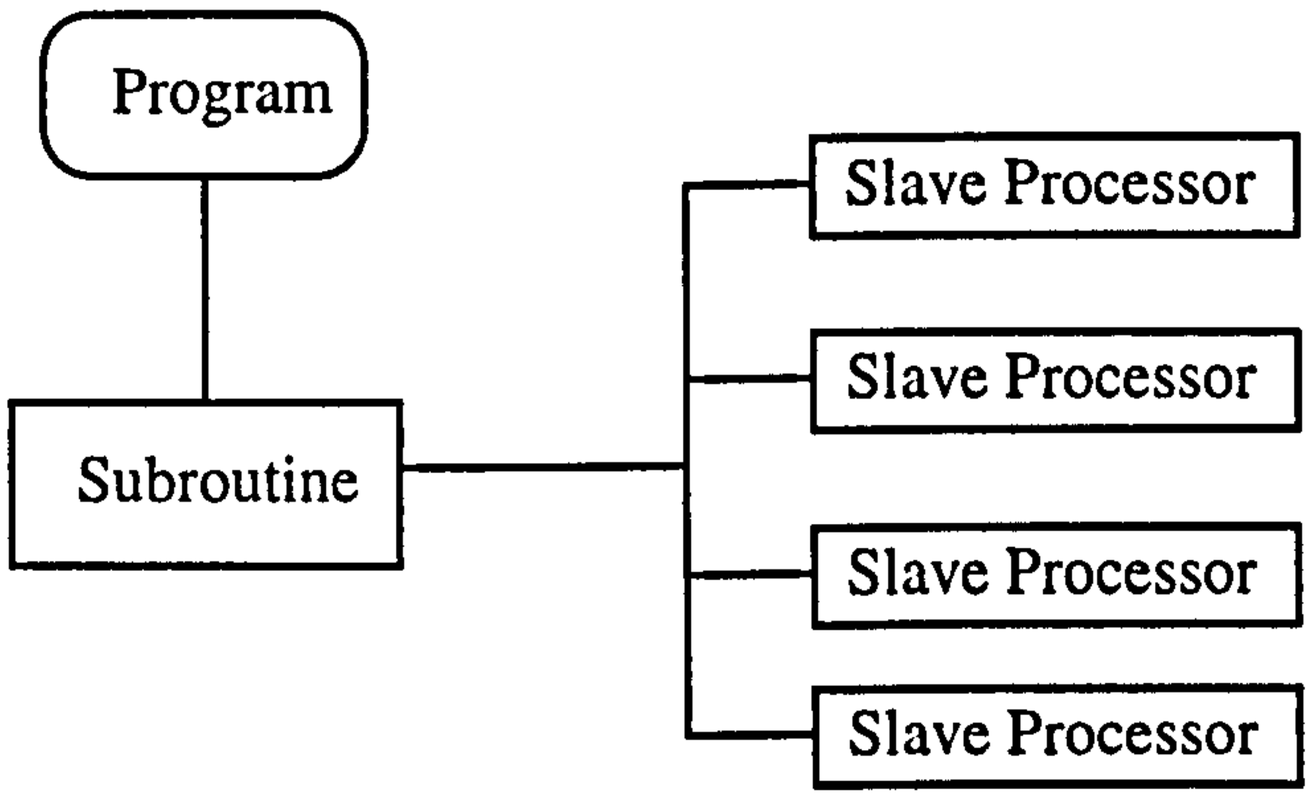


Figure 7-6 Idealisation of a subroutine being farmed out to several processors.

Secondly, when the main program calls a subroutine, which then in turn sequentially calls several other subroutines, each of which calculates a solution independent of the other subroutines. It is possible to distribute the calls to the subroutines onto individual processors, each processor only calculating the solution for a specific subroutine.

Again this allows the concurrent calculation of several solutions, the final solution being passed back to the master processor upon completion. Figure 7-7.

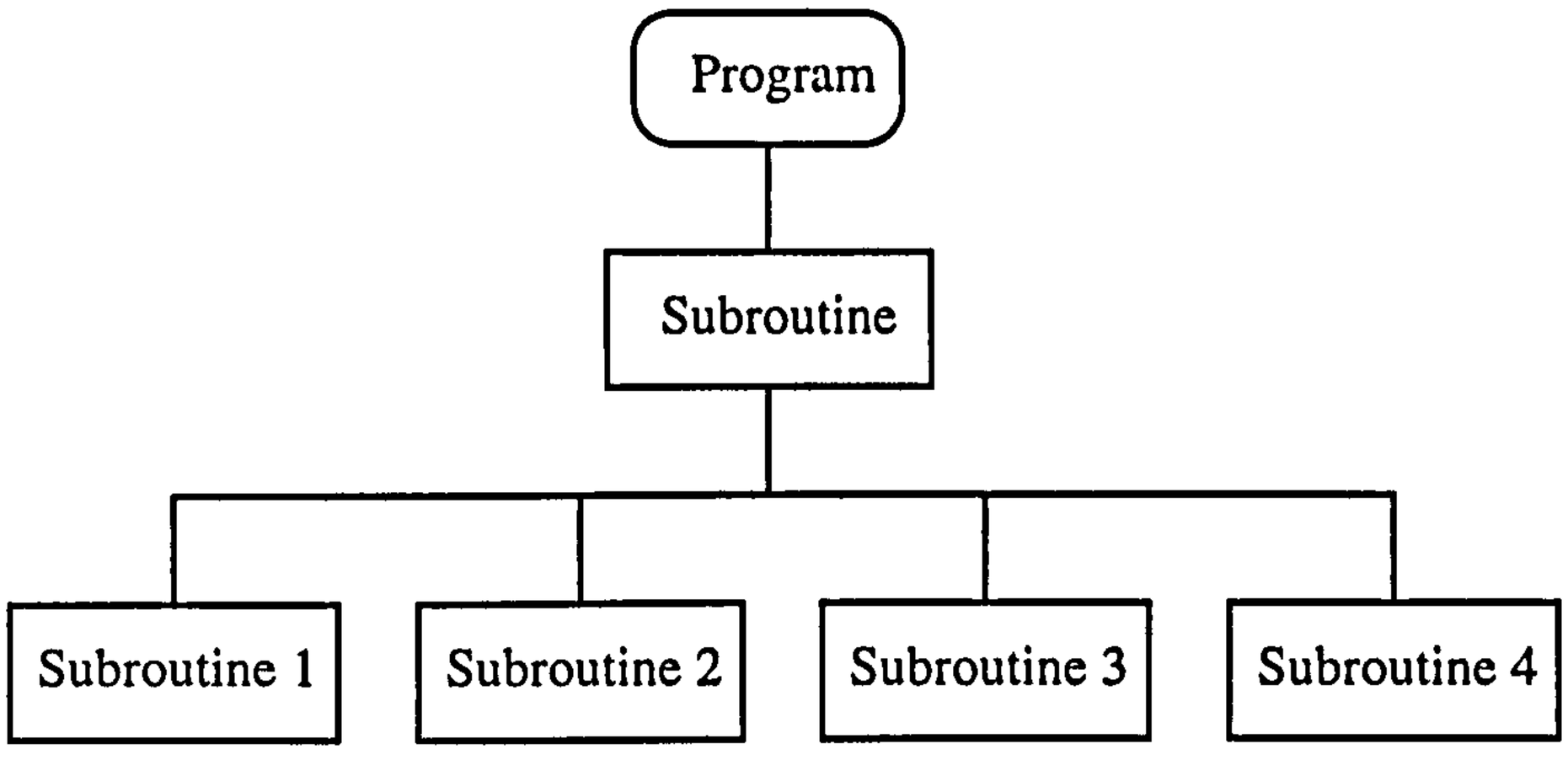


Figure 7-7 Idealisation of subroutines running concurrently.

Both the above strategies can be individually applied in the parallelisation of existing serial algorithms. However, there is the possibility of implementing a hybrid form of the two strategies that would allow large sections of the serial codes to be massively parallelised.

All the above strategies have been widely applied, in the parallelisation of the serial ANSUSP codes, as detailed in Chapter 9.

8 Distributed Computer Environment

To determine the limitations of the distributed computer system at hand, several system dependent measurements were taken to gain an insight into its calculation and communication performance characteristics. The results were subsequently employed to optimise the division of tasks among the processor network and the communication potential of the distributed computer system.

The distributed computer systems used for this project were a PARSYTEC Super-Cluster and Multi-Cluster, the former having 64 processors, the latter 32. The processors were T800 transputers, each processor having 8 MB of dedicated on-board RAM. These systems used a proprietary parallel operating system called PARIX[111], which allows a serial computer to control the execution of a distributed computer system.

The T800 transputers used in the systems are high powered microprocessors with on board RAM, which have been designed with the concept of linking a series of such processors in a parallel or distributed system.

Each transputer has a peak calculation performance of 30 Million Instructions Per Second (MIPS), which is comparable to a 486DX2 66MHz processors that was widely used in desktop computers in the early 1990's. However, due to their capability of linking the chips in a parallel manner, this means that a program can be divided into distinct components and executed concurrently on a series of individual transputers.

To determine the optimum configurations for both the distributions of tasks among a processor network and the optimum topology that should be used to communicate efficiently the partial solutions around the computer network, several specific system measurements were made. The main questions to be addressed were as follows;

i. What were the inter-processor communications rates around a network ?.
ii. What was the relative speed of performing a simple repetitive calculation compared to the inter-processor communication rates ?.

8.1 Measurement of System parameters

8.1.1 Determination of Communication Rates.

A basic 3 x 2 processor array was employed to measure the system characteristics. The network was mapped into the virtual topology of a farm communication network. This is a master processor communicating in sequence to a network of slaves. This network is demonstrated in Figure 8-1. The Master processor is denoted as having processor identification (procid) equal to 0, as will be the case for all subsequent discussions related to the development of communication networks.

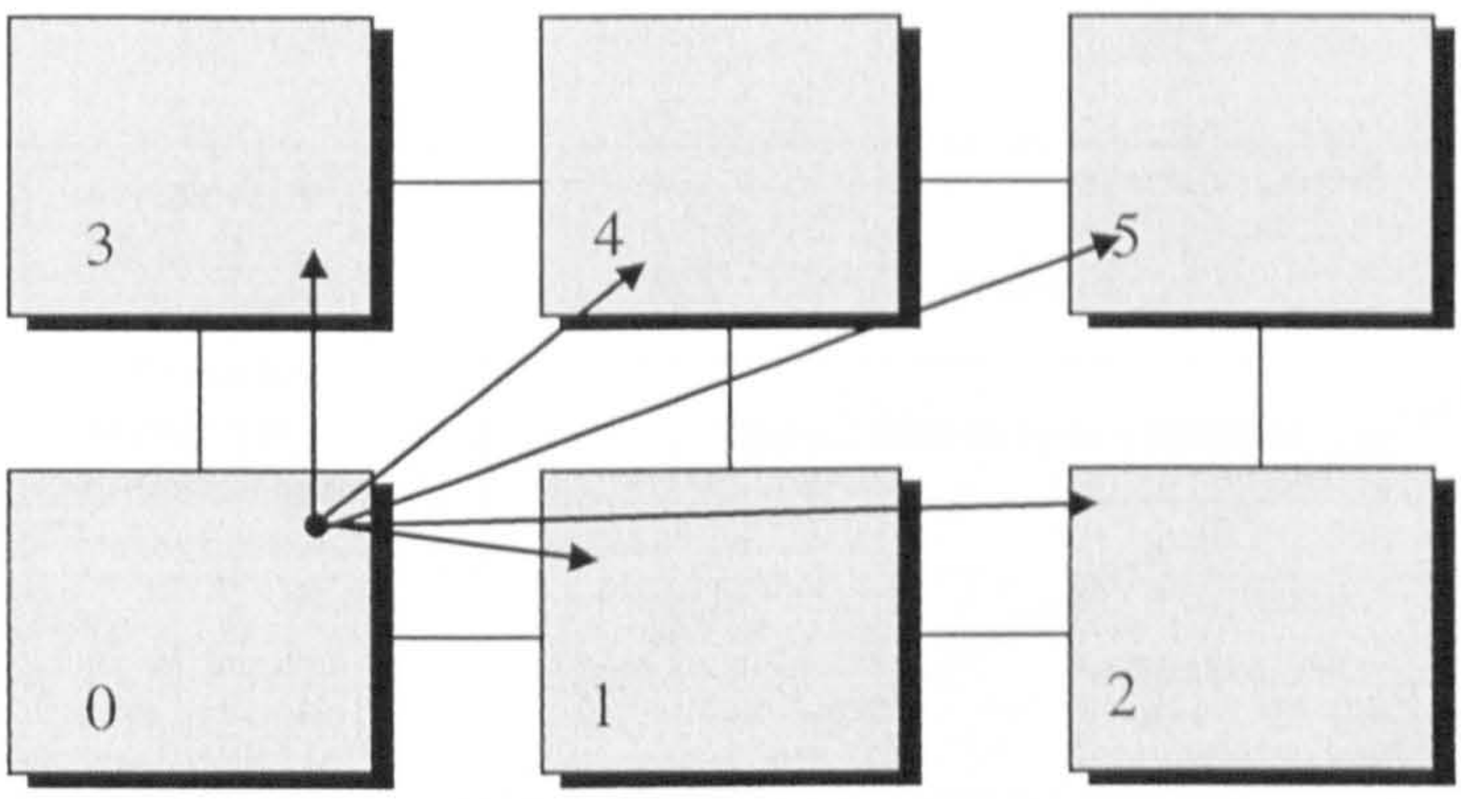


Figure 8-1 Processor network with Farm Communication Topology

The processor network communication characteristics were measured by sending messages of different lengths from the master processor to the slave processors in sequence. The messages ranged in size from 50 to 12800 Bytes in steps of 250 Bytes. The upper limit on message size was determined by the potential size of some of the larger problems that would be attempted using the parallelised program. This large amount of data being passed in one communication could be the results of several vectors of information.

The results of these measurements are shown in Table 8-1 and Figure 8-2.

Communication from Master to Processor	Communication Rate (μSeconds/ Byte)	Communication Overhead (μSeconds)
1	0.9831	86.95
2	1.6400	33.28
3	0.9823	87.03
4	1.6250	29.24
5	1.6260	65.62

Table 8-1 Absolute communication rates.

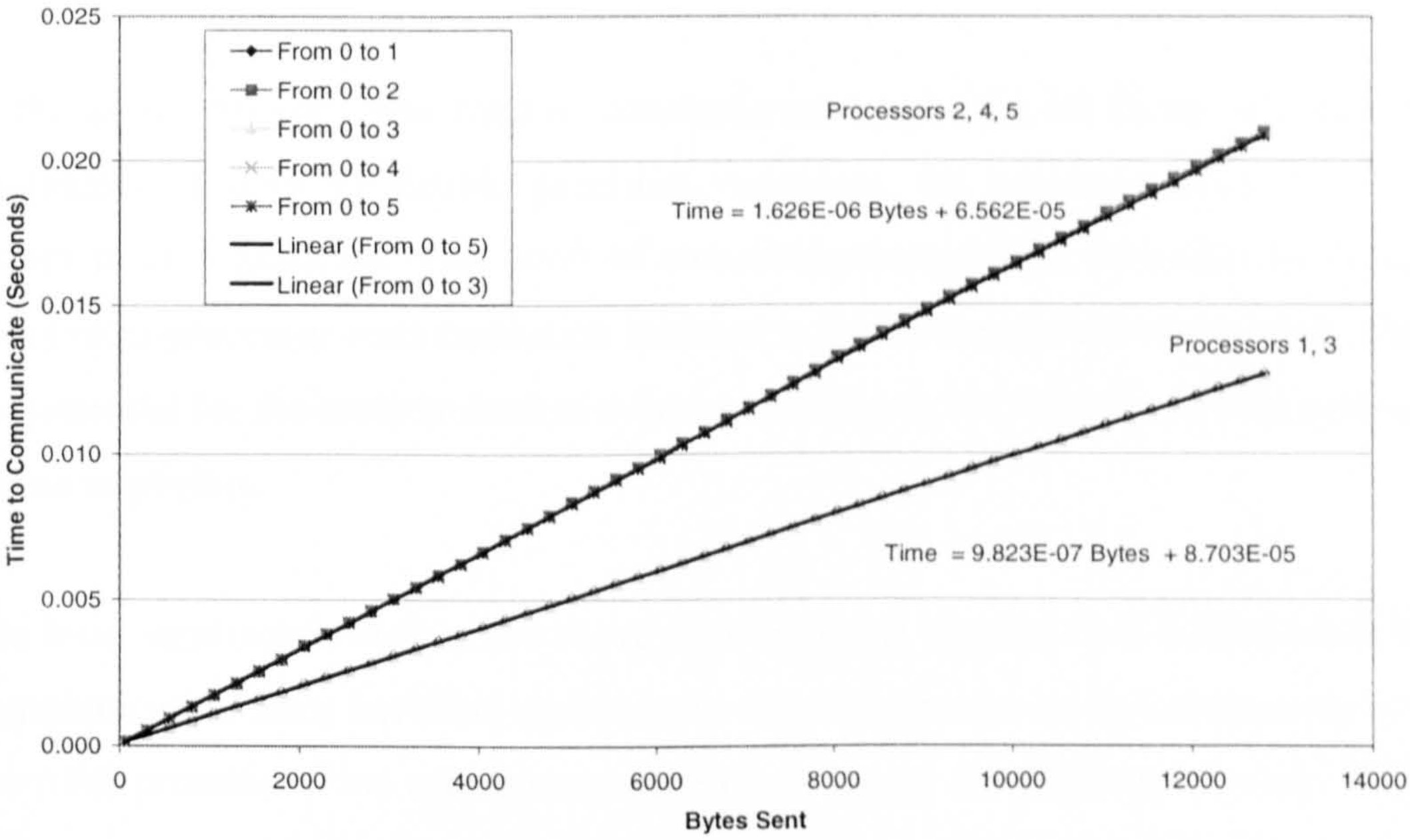


Figure 8-2 Measured inter-processor communication speeds.

As can be seen from Table 8-1, the time in seconds required to send x Bytes of data from the master processor to a specific slave processor is given by Equation 8-1,

$$\text{Transmission Time} = \text{Communication Rate} \cdot x + \text{Communication Overhead}$$

Equation 8-1

The Communication Overhead for any single inter-processor communication was around 29-87 μSeconds, while the actual Communication Rate, ranged from 0.9823

μSeconds/ Byte for the processors immediately adjacent to the master processor, up to 1.640 μSeconds/ Byte for the non-adjacent processors.

Thus if the network was used to send small amounts of data per iteration the communication overhead would prove to have a considerable effect upon the communication network and lead to a considerable data bottleneck forming at the Master processor.

However, if the network was used to distribute a large amount of information, the communication overhead would quickly become negligible compared to the actual communication rate of the communication link.

If the above network was used to communicate more than 88 Bytes of data (the equivalent of only 11 double precision variables), the communication overhead ceases to be significant. This level of communication will be exceeded by almost every inter-processor communication initiated in the final parallelised program. Thus the potential for the communication overhead leading to bottlenecking in the network seems negligible.

The most significant result in the above measurements was the clear difference in the communication rates between immediately adjacent processors in comparison with those for processors that communicate indirectly across the physical network. These indirect communications were achieved by forming a virtual link between the two processors.

A bottleneck forms when a communication from a slave processor to the master processor takes so long as to delay the subsequent communications from any other processor, thus resulting in a delay in the network. This situation would continue to build until such a time as the original communication was completed, allowing the waiting communications to be completed in sequence. This is a particular problem associated with programming parallel synchronous communication networks. The problem is effectively solved by using asynchronous communication. However in the current project no asynchronous communication was attempted due to the PARIX operating system not supporting this feature in the FORTRAN language.

8.1.2 Measurement of Simple Repetitive Calculation

The purpose of these measurements was to determine the optimal point at which the communication network should be used compared to calculating the solution locally. That is, at what point would it be faster to communicate a partial solution calculated on a remote processor and simply add that partial solution to the partial solution already contained in the target processor ?.

To this end a study of all the main routines used in the eigensolution module of ANSUSP were examined to locate a piece of program structure that was extensively utilised in one of the main iterative sections of the respective codes.

The program structure that was identified was extensively utilised in several subroutines such as VECSUM, ORTHOG and PREDIC, subroutines identified as benefiting most from being parallelised.

The purpose of the identified program structure was to calculate the values of a vector, the vector ordinates being determined from the sum of the previous value of the vector component and the product of two other vector components.

$$EL = EL + \hat{W} \cdot \hat{V}$$

```
      DO 4 I = LOCK,M
      DO 400 J = 1,N
      EL= EL+ W(I,J)*V(I,J)


  400 CONTINUE
    4 CONTINUE
```

The measurements were made for iteration counts ranging from 10 to 1000 in steps of 10. Again the upper limit was determined from possible problem sizes in the future.

The results of these measurements are shown in graphical form in Figure 8-3 and show that if the communication overhead were neglected, the calculation rate is 1.75 times faster than the fastest communication rate considering the calculation of a single double precision values (equivalent to 8 bytes). Thus if the total

communication time is compared to that of the processor calculating the values, the difference will increase with number of cycles, as is clearly demonstrated Figure 8-4.

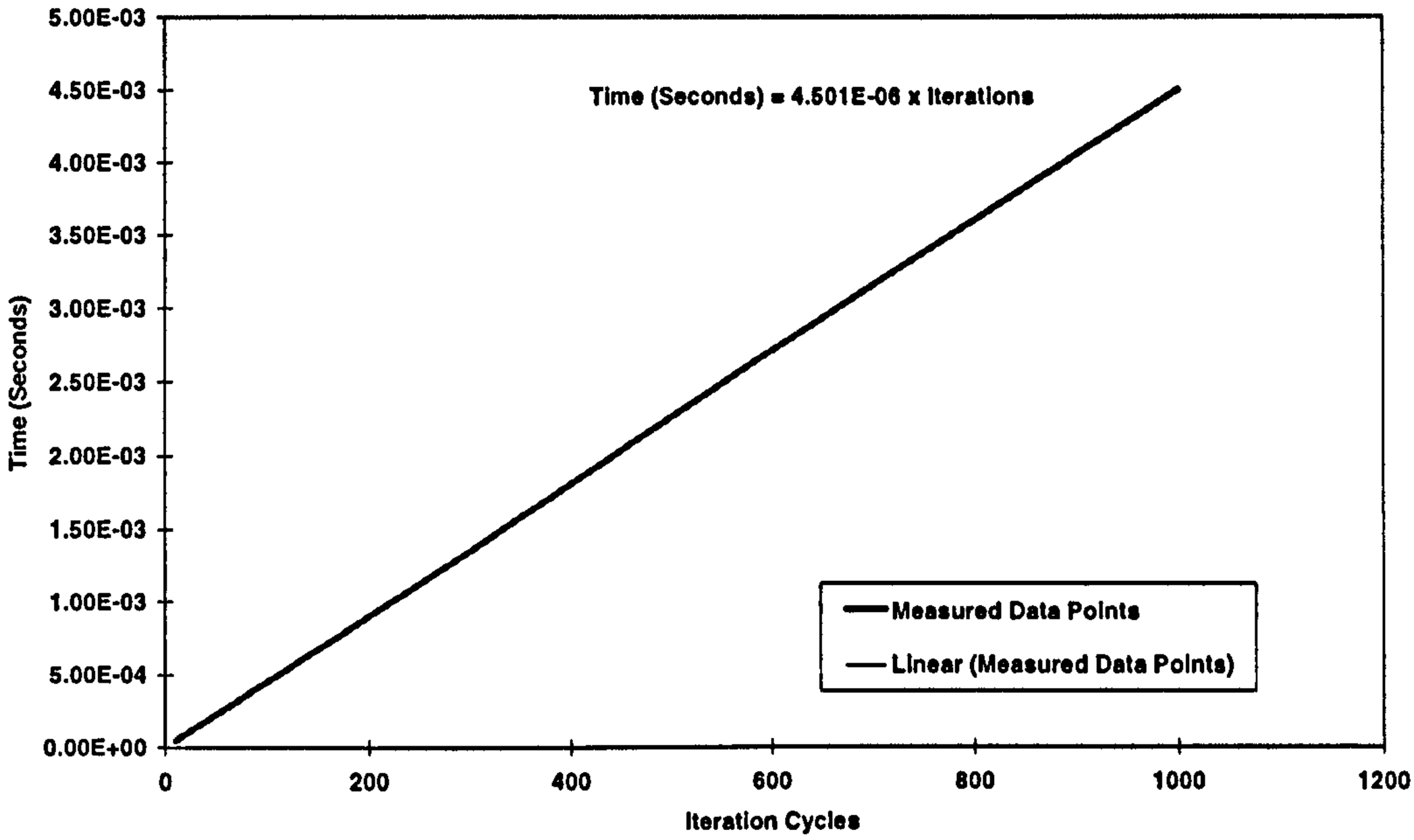


Figure 8-3 Measurement of simple calculation rate.

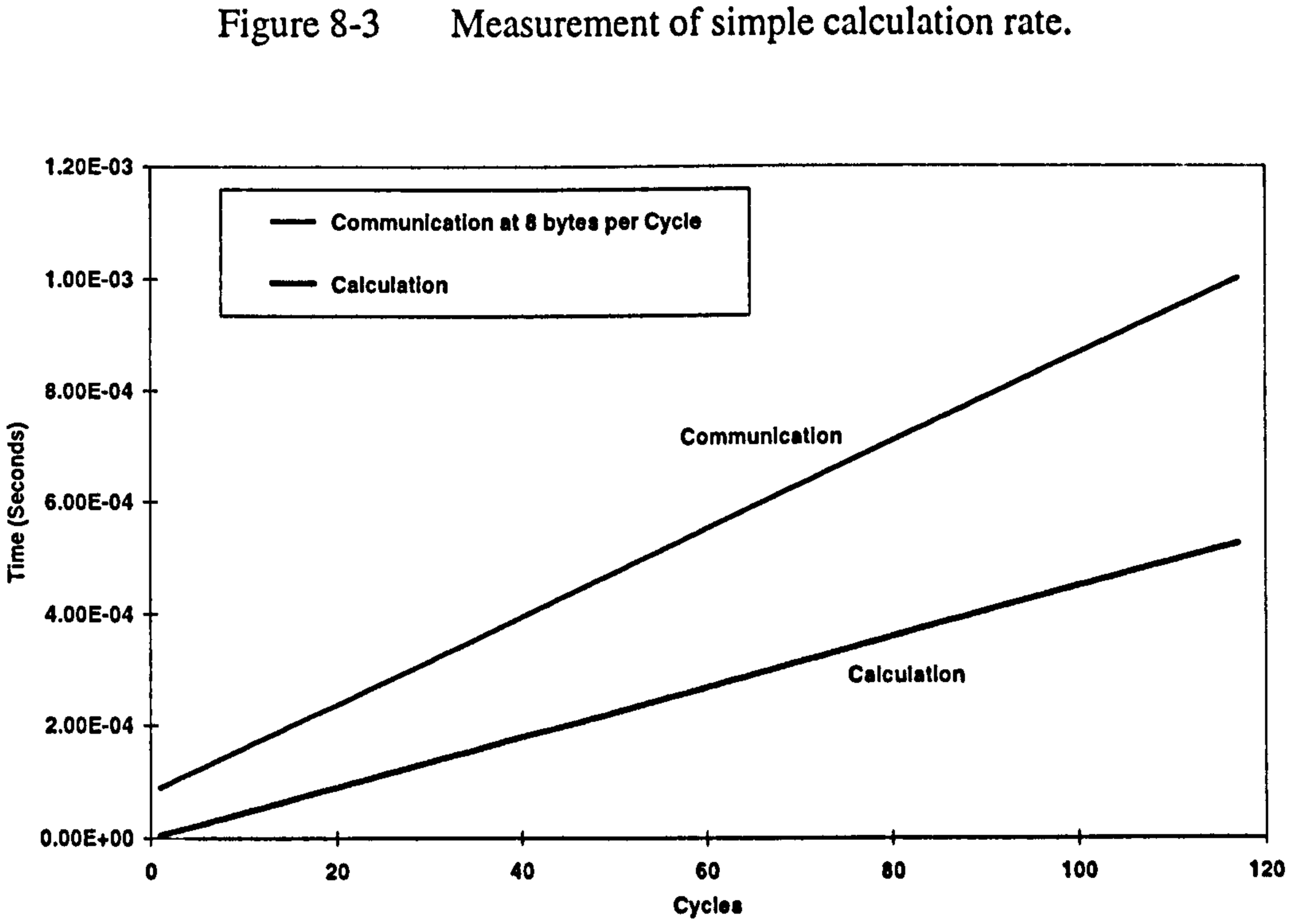


Figure 8-4 Comparison of calculation and fastest communication rates.

8.1.3 Conclusions

The main purpose of these measured results was to be able to develop an efficient inter-processor communication network which could minimise both the potential for data bottlenecks forming as well as the time required to perform a specific task, while maximising the level of work division for that particular task.

There are two main conclusions that can be drawn from the measurements made in this section. Firstly, with respect to the virtual topology of the communication network, it is obvious from Table 8-1, that the communication rate is nearly twice as high for a processor communicating to an adjacent neighbour compared to one communicating to a non-adjacent neighbour. Thus any topology that wishes to minimise the global duration of inter-processor communication, could do so by ensuring that the processors can only communicate with their adjacent neighbours.

Secondly, as was shown in Figure 8-4, the calculation rate of any single processor was substantially faster than the fastest inter-processor communication rate. Hence whenever possible inter-processor communications should be kept to a minimum, while on-processor calculation should be done as much as possible.

This has a considerable implication on the level of parallelisation that should be attempted for any individual computational task. If a task is parallelised to such a level as to reduce the onboard calculation to a negligible level, while the inter-processor communication demands increase, the task would actually take longer to complete than if it had been parallelised to a lesser extent.

8.2 Development of Generic Communication Networks

The main requirement for any concurrent program to operate efficiently is the development and successful implementation of an efficient generic inter-processor communication network.

This network strategy would allow the passing of information calculated and stored on one processor to all other processors in the network that require those pieces of information to progress with their own calculations. The significance of the network being generic is that it would be 'self-scaling'. The communication topology would be programmed with logic allowing it to determine the number of processors in the network and how they are connected. This internal logic would instruct the topology how it should map itself onto the available processor network and to whom each processor should communicate and in what sequence.

8.2.1 Farm Network

One of the most basic inter-processor networks is that of the Farm Network Topology, in which one processor acts as a Master processor and all others are treated as Slaves. The Master processor is identified as having processor identification (procid) equal to 0, Figure 8-5, with the rest of the processors 1 to (N_p - 1) being Slaves, where N_p is the number of processor in the network. Figure 8-5, also shows the dimensions of the network- DimX and DimY processors in the two directions respectively.

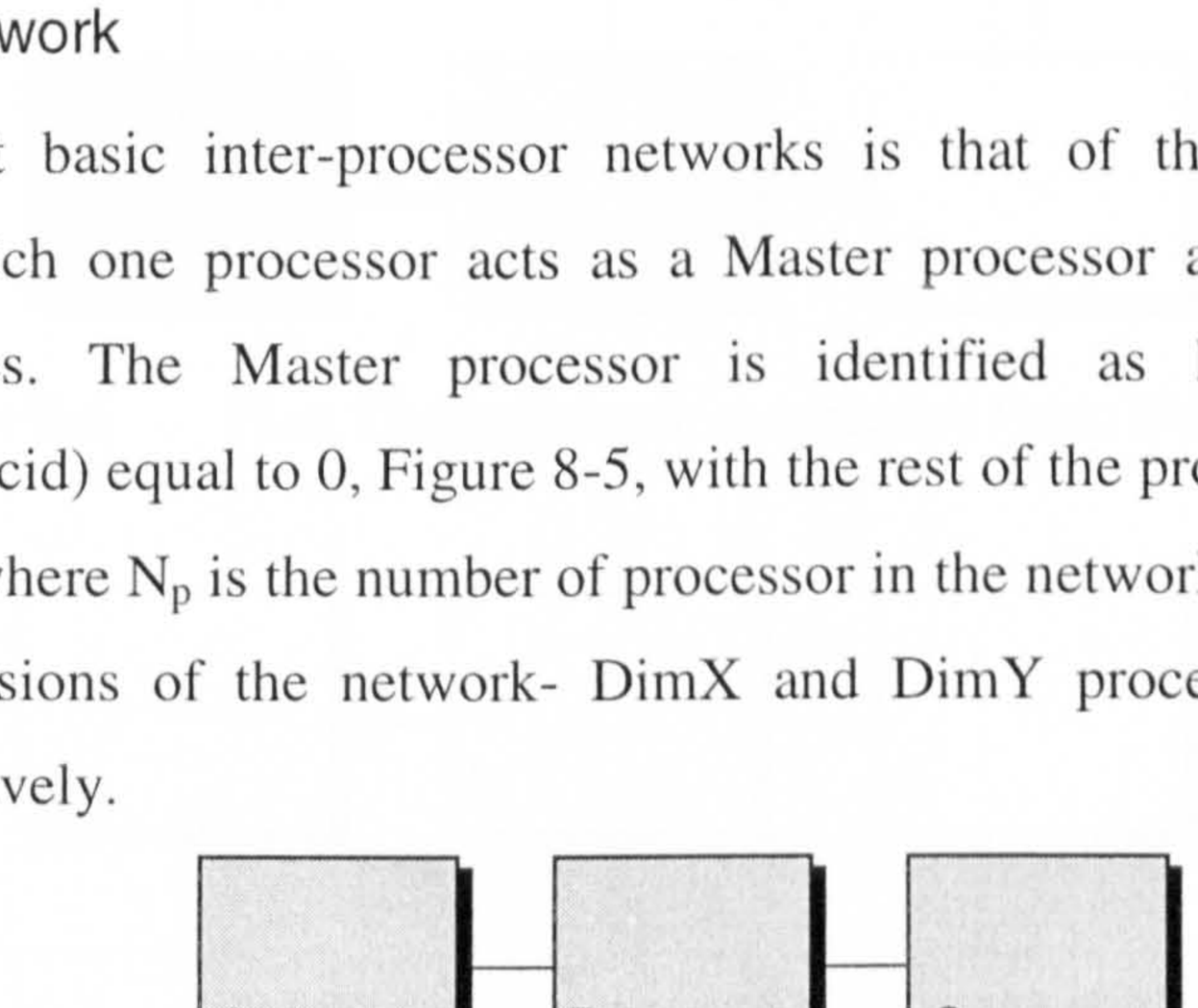


Figure 8-5 A Basic Processor Network

The Farm topology allows the Master to communicate with each slave processor in sequence. Consequently, as the Master processor communicates to processor 1, the remaining processors in the network from 2 to N_p are idle, Figure 8-6. Thus it can be said the network is obtaining its solutions from a single source. The number of idle processors reduces by one for each communication cycle. The maximum delay experienced by any individual processor can be quantified as ((DimX x DimY)-1) communication cycles. Figure 8-7 shows the second Cycle, with subsequent cycles communicating to processors 3 to N_p respectively.

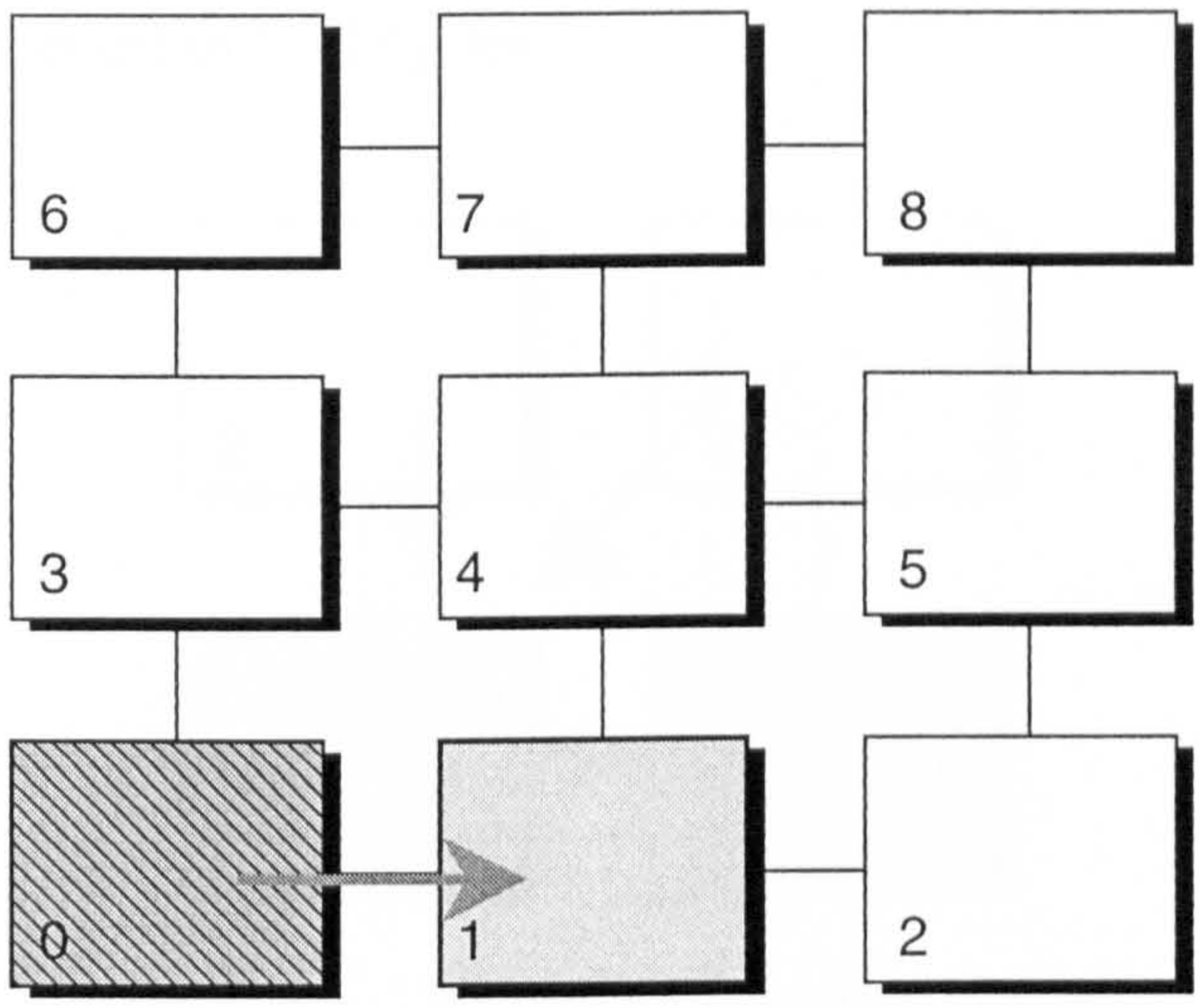


Figure 8-6 First Communication cycle for Farm Network.

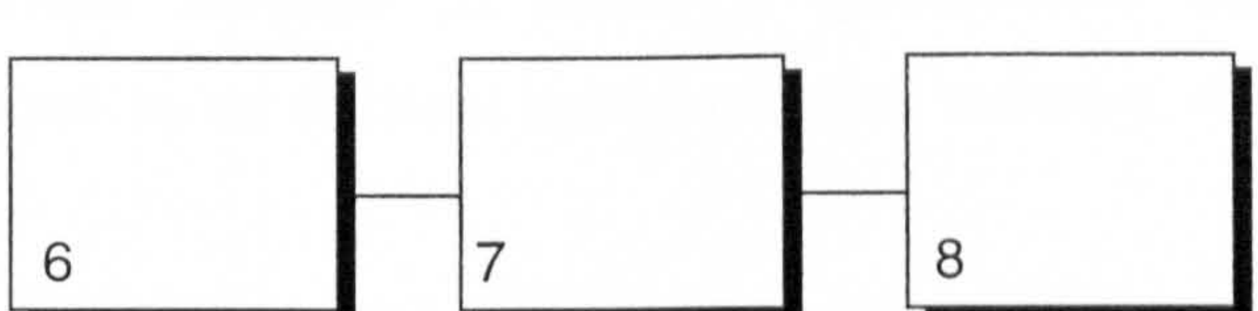


Figure 8-7 The Second Communication cycle for Farm Network.

Idle Processor

Transmitting processor

Calculating processor

Receiving processor

The Farm recovers the solutions from the network in the same way, communicating with each processor in sequence. The problem associated with this is the number of cycles required to recover and transmit all the data to and from the master processor from the network. This leads to large delays in the network due to communication bottlenecks forming when the slave processors are trying simultaneously to communicate with the master processor. Figure 7-5, shows a communication bottleneck with three slave processors simultaneously trying to communicate with the master processor. The result is that the network would recover the solution from processor 1, delaying 2 and 3. The maximum delay being experienced by any one processor would be equal to N_p-1 cycles.

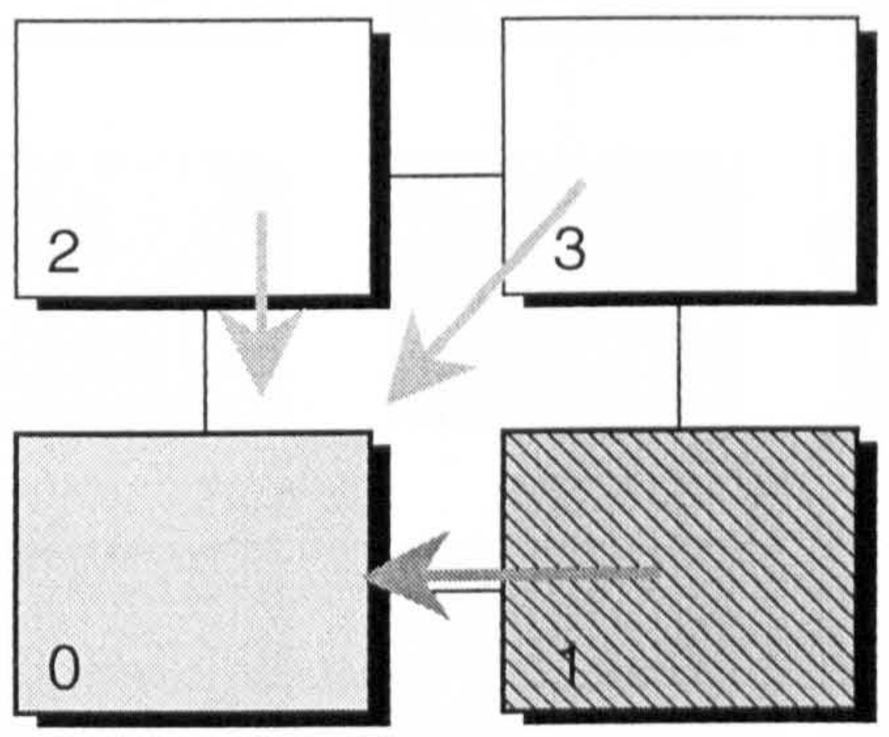


Figure 8-8 A Communication Bottleneck

Bearing the previous network in mind, a specification detailing the main characteristics desired in an efficient communication network were identified. The main points were:-

i. The network should be entirely generic and scale itself to available resources.
ii. The network should be capable of communicating from multiple roots.
iii. The network should allow data to be communicated around the network without significant difficulty.

The network topology developed during this research was termed the Finger Network due to the communication structure employed by the network.

8.2.2 The Finger Communication Network

The Finger Network was a significant improvement on the Farm network, although it also has a small bottleneck problem associated with it. The Finger network, unlike the Farm version, has a totally adaptive communication structure allowing the maximum delay experienced by any single processor to be minimised.

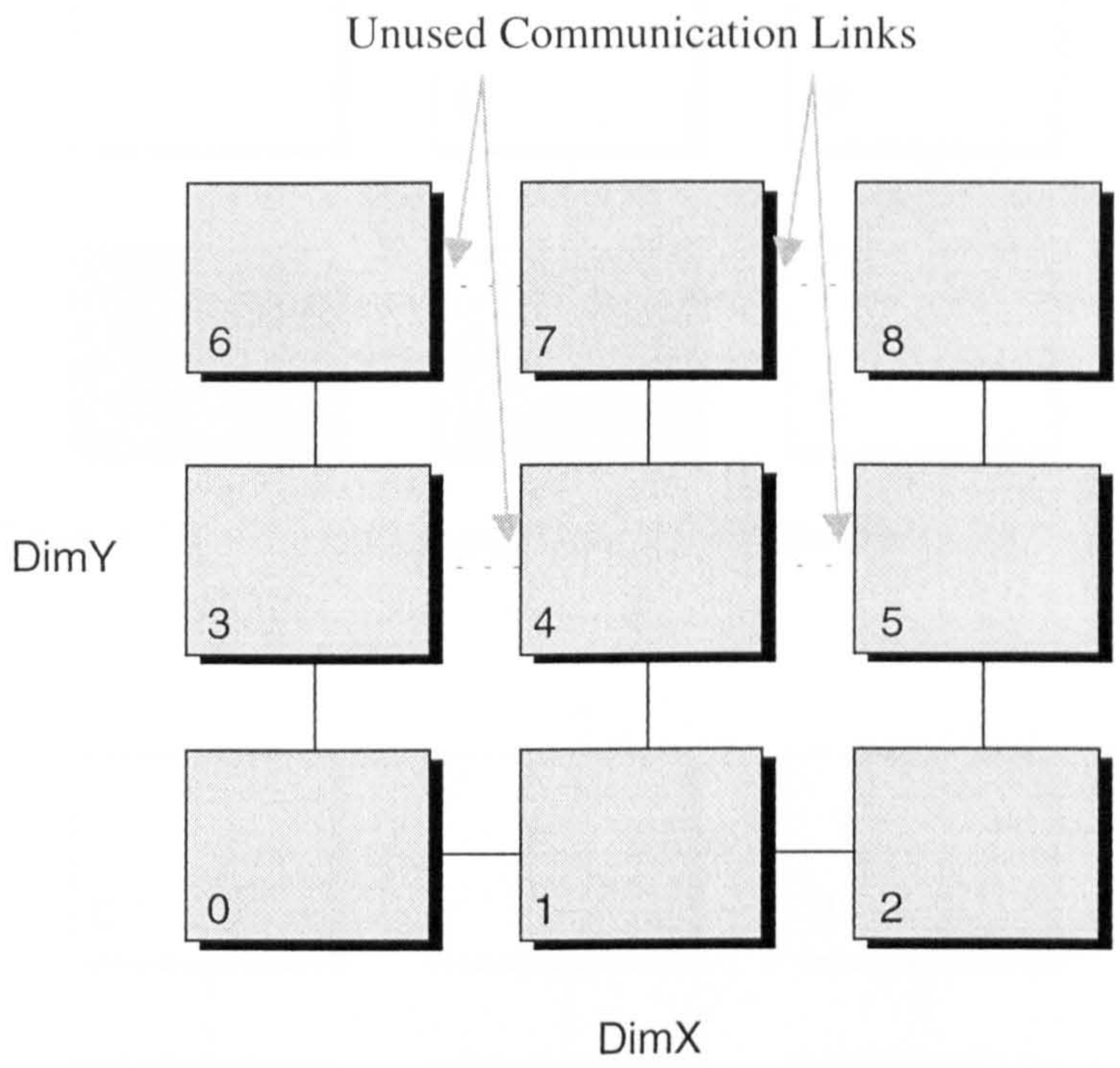


Figure 8-9 The Finger communication network showing the physical communication links

Initially the network communicates from the master to processor 1, in a similar way to the Farm network, the rest of the processors in the network being idle, Figure 8-10. However, in the second cycle, Figure 8-11, the Master processor communicates to processor DimX, while processor 1 communicates to processor 2. This methodology is repeated cycle after cycle with communication from multiple sources, the number of idle processors reducing by a maximum of DimX with each communication cycle, Figure 8-12 & Figure 8-13. The maximum possible delay experienced by a processor is (DimX + DimY -2) communication cycles, with the corresponding maximum number of sources simultaneously communicating being DimX.

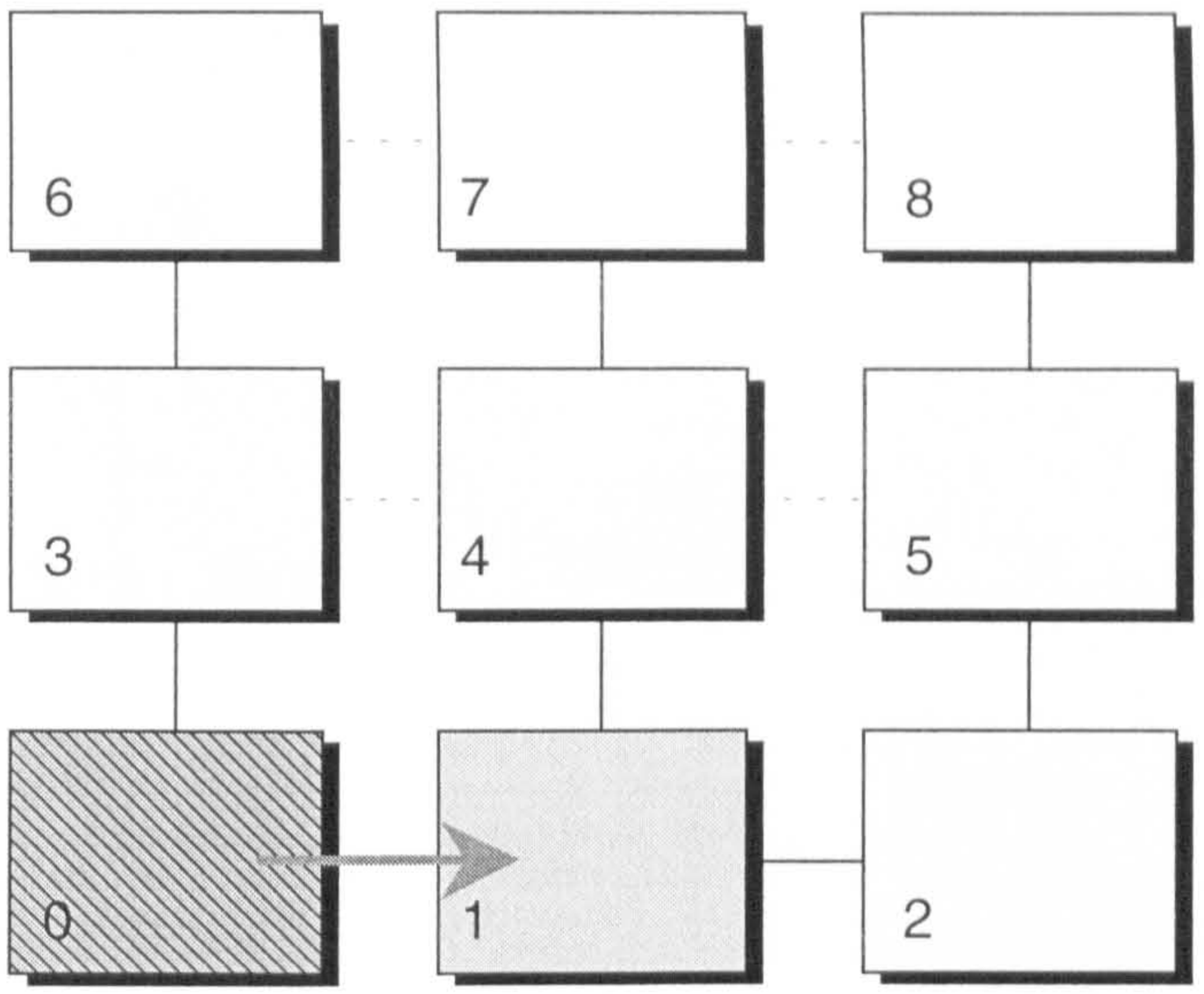


Figure 8-10 First Communication cycle for Finger Network

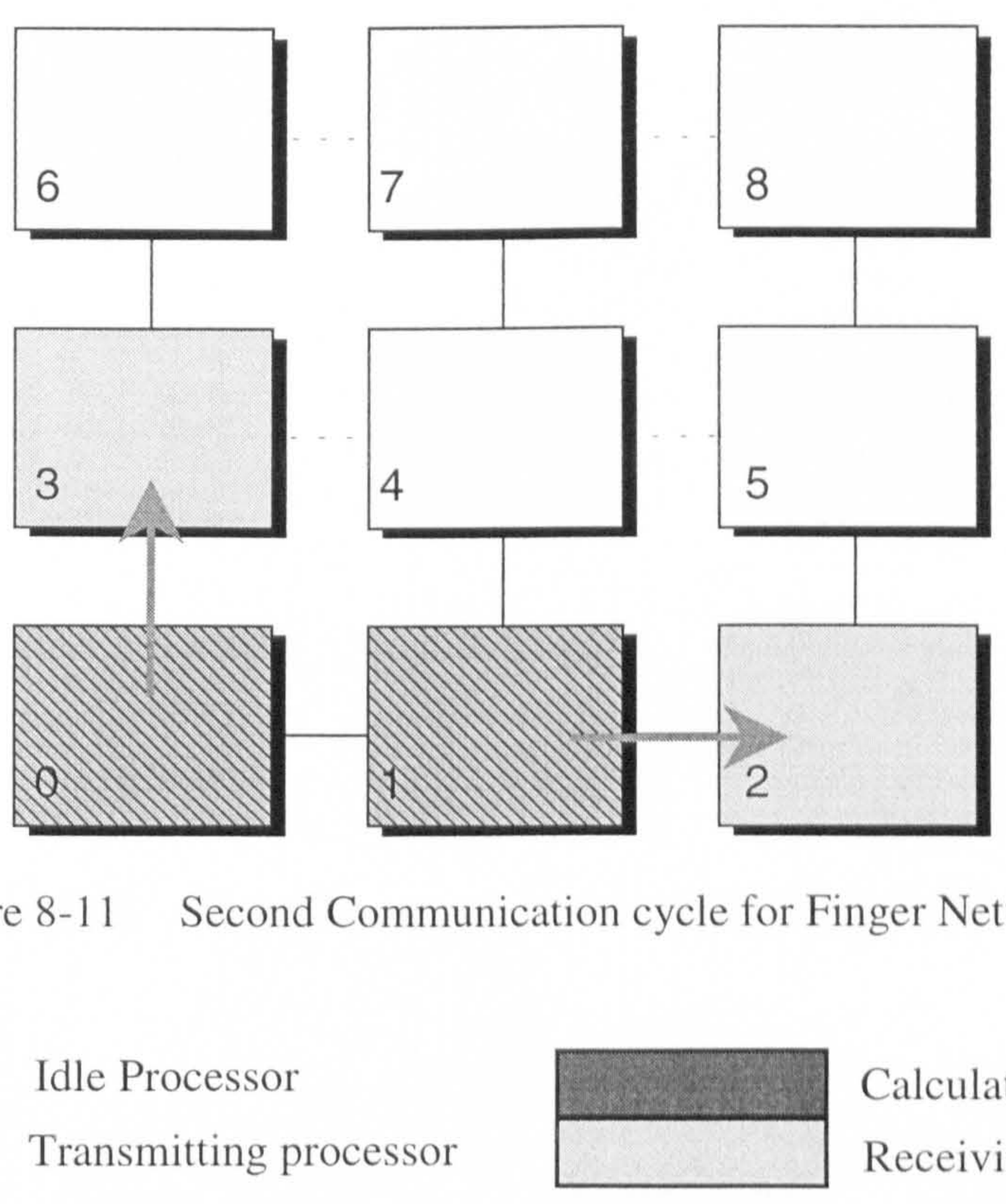
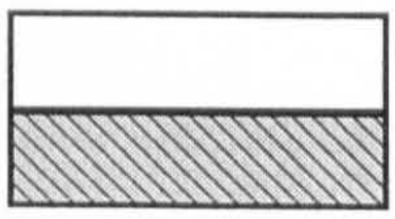


Figure 8-11 Second Communication cycle for Finger Network



Idle Processor

Transmitting processor

Calculating processor

Receiving processor

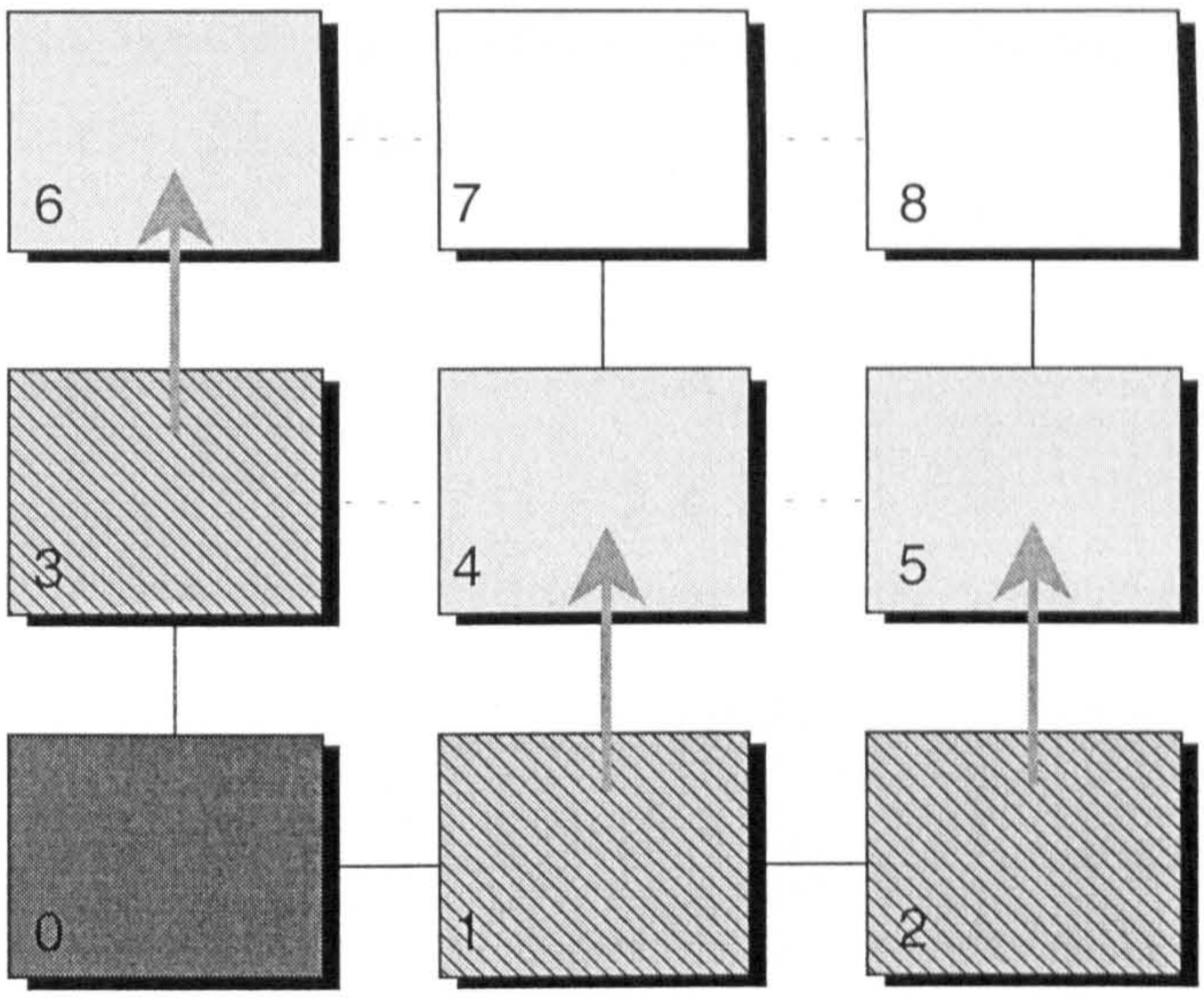


Figure 8-12 Third Communication cycle for Finger Network

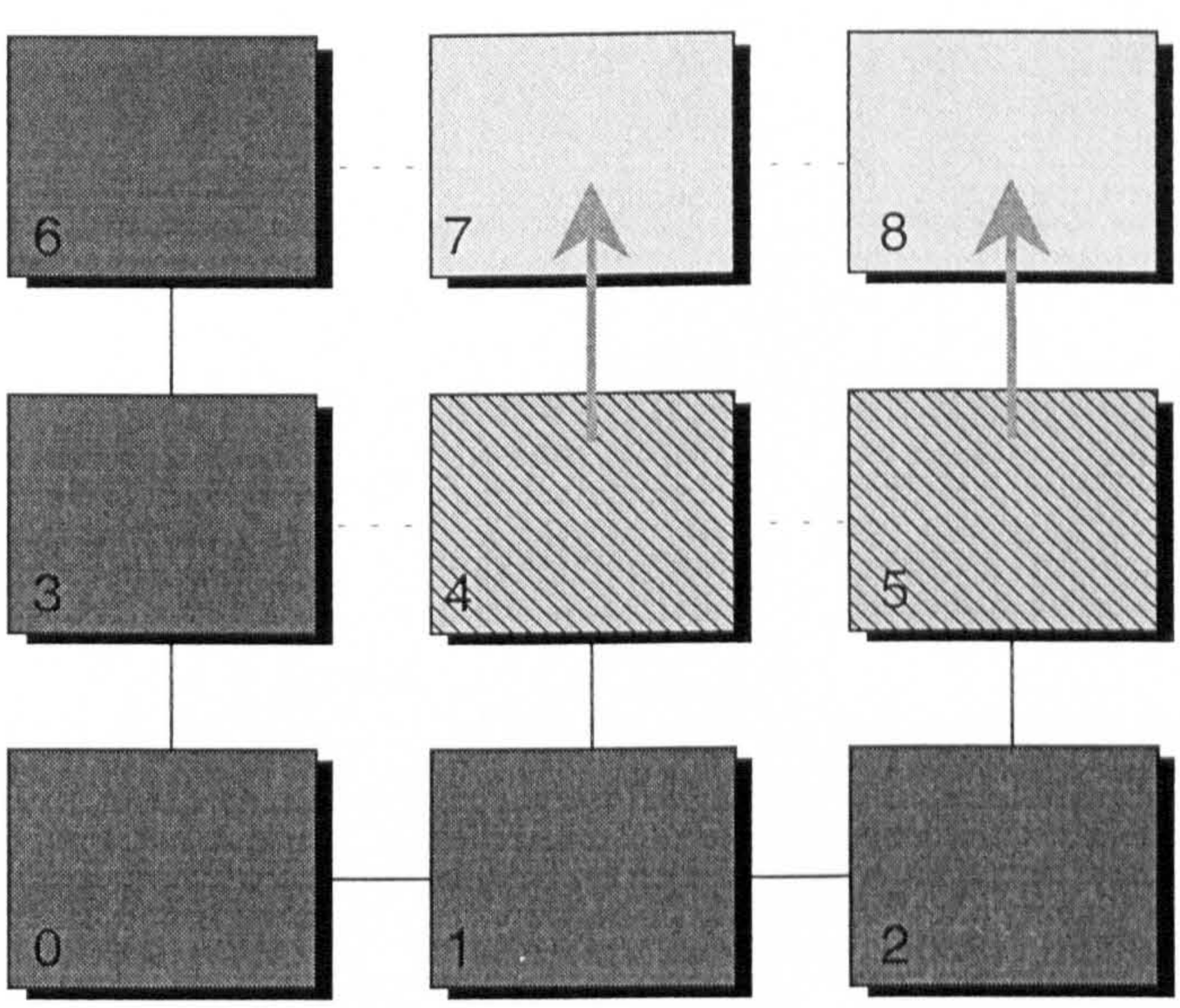
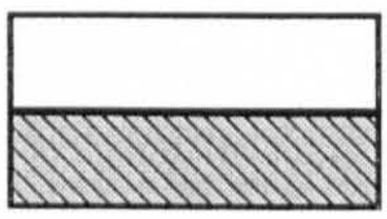
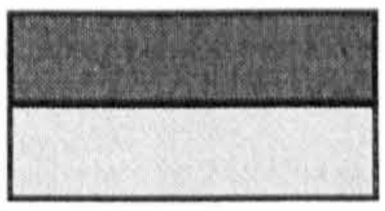


Figure 8-13 Final Communication cycle for Finger Network



Idle Processor

Transmitting processor



Calculating processor

Receiving processor

Another significant difference between the Finger and Farm communication topologies is that in the Finger topology, each processor communicates with their adjacent neighbour, allowing a significant reduction in the time required for each communication cycle, refer to Section 8.1.1.

9 Parallelisation of The Ansusp Program

The ANSUSP[34] program is a three dimensional suspension bridge dynamic analysis program, that is capable of performing six different types of analysis. These various types of analysis are solved using two sets of algorithms. The first directly solves the equations of motion by numerical integration. The second performs the eigenvalue analysis of the system to calculate the natural frequencies and normalised relative modes of the structure. For more details of the ANSUSP program refer to Chapter 4.

9.1 General

9.1.1 Time History Analysis

The three main types of Time History analysis that utilise the numerical integration algorithm are the Static, Time-History Ground Motion and Time-History Flutter forms of analyses.

The numerical integration procedure of dynamic analysis is based upon satisfying the dynamic equation of forced vibration Equation 9-1. If the structure is analysed as its position in space move (i.e. equilibrium is applied to the structure in its deflected shape) then the method is capable of analysing a suspension bridge structure during its construction stage when it is highly non-linear.

$$[M]\{\ddot{u}\}+[C]\{\dot{u}\}+[K]\{u\}=\{P\}$$

Equation 9-1

The solution was achieved using the Newmark Implicit integration scheme that requires equilibrium to be satisfied at all times. Thus for every iteration of a particular time step, the method requires the nodal displacements and forces within the structure to be known.

To facilitate this, the algorithm RESVEC was written. This algorithm calculates the forces in the five main components of the structure, namely the Cables, Hangers, Deck, Towers and Anchorages. For each time increment, RESVEC was given a set

of member end displacements, from which it would calculate the forces acting in each member and at joints. These forces were combined into a global force vector which is used by the Newmark scheme to calculate the global out-of-balance force vector for that iteration of the time step. The Newmark procedure continues to iterate until equilibrium is achieved to a satisfactory tolerance. The two schemes implemented are the Linear Acceleration Method and Fox-Goodwin Method which are both conditionally stable. For a more detailed discussion on various numerical integration methods refer to Section 2.2.3.1. Figure 9-1 shows a schematic view of the calls within RESVEC.

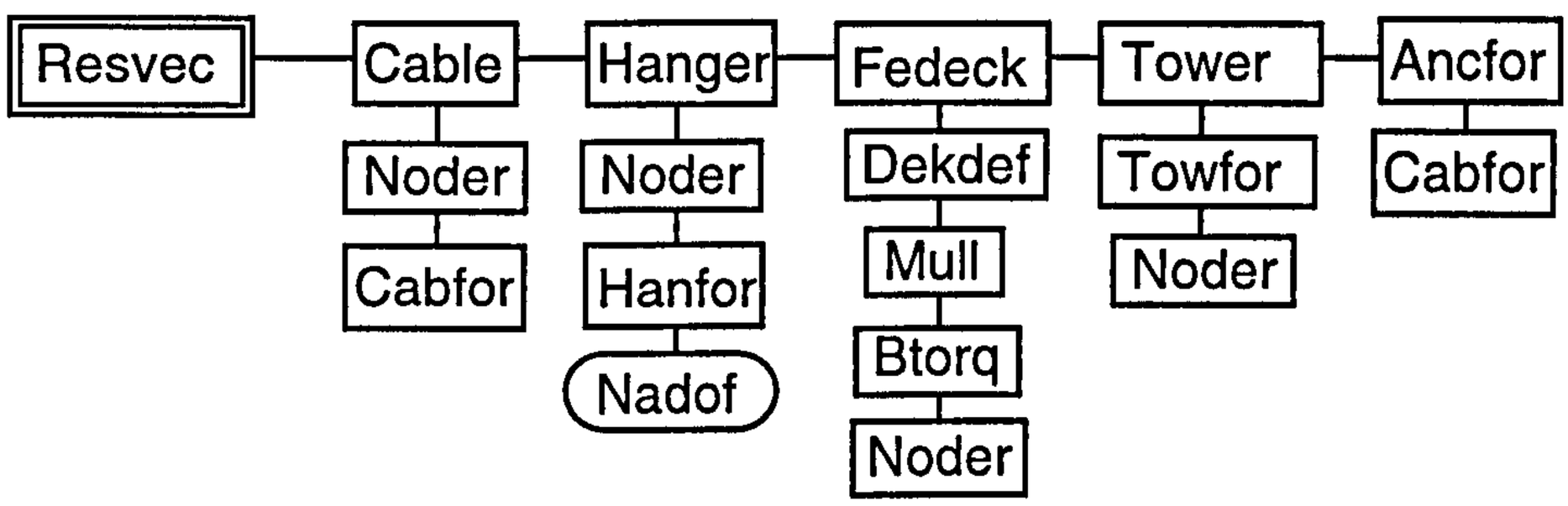


Figure 9-1 Calls within routine RESVEC

The main difference in the solution procedure for the three different forms of analysis is the external loading applied to the structure and level of artificial damping. In the particular case of Static analysis, this employs the method of dynamic relaxation. The loading is defined at the beginning of the problem and the structure needs a percentage of critical damping applied to its motion to ensure convergence to the correct static solution with the minimum computation, refer to Section 3.1.2.

In contrast both the Time-History Flutter and Time-History Ground Motion analyses have temporal external loading functions that require to be augmented at the beginning of each time step. Hence, the loading is recalculated within the algorithm at the beginning of each time increment.

9.1.2 Eigenvalue Analysis

There are two main eigenvalue algorithms within the ANSUSP package, each being used for different forms of analysis. The Simultaneous Iteration algorithm developed by Jennings[62], is used for Natural Frequency Analysis of the structure, while the QR method developed by Francis[68] is used to perform Modal Flutter Analysis.

The QR method has been extensively studied and parallelised by Dongarra *et al* [85], who employed a 'divide and conquer' method to subdivide the eigenvalue problem into smaller problems that were concurrently solved around a processor network.

Due to the extensive body of research and since the majority of problems solved with this method are of small order, no attempt will be made to parallelise this algorithm.

The Simultaneous Iteration algorithm on the other hand has not been extensively researched, the most significant work being by Agar[112], who vectorised the majority of the algorithm, with relatively modest success in terms of overall speed-up.

9.2 Parallelisation of Time History Analysis

The main goal of any parallelisation of existing serial code is to reduce the total CPU time required to complete a standard analysis. In the present case since the solution is obtained using an iterative numerical integration scheme, the greatest opportunity for improvement is to parallelise the section or sections of code that are called at every iteration. Table 9-1 shows the percentage of CPU time required by the main components of the numerical integration routines, for four different suspension bridge model idealisations.

The NEWMARK subroutine calculates the displacement and velocity vectors for each iteration of a particular time step using the results of RESVEC, until satisfactory equilibrium is achieved for that time step. The external loading is calculated at the beginning of each time step and remains constant for each iteration of that time step. RESVEC calculates the equilibrium forces within the structure ensuring that the external loading is balanced at the end of each time step, by a combination of stiffness, damping and inertial forces within the structure.

The model idealisation A-B-C, indicates that there are A & C number of deck sections modelled in the left and right side spans of the bridge respectively, while B represents the number of deck sections in the centre span.

Model Idealisation	NEWMARK	External Loading	RESVEC
3-9-3 (152 dof)	0 %	1.1 %	98.9 %
5-16-5 (251 dof)	0 %	2.0 %	98.0 %
8-27-8 (404 dof)	0 %	1.5 %	98.5 %
17-54-17 (809 dof)	0 %	1.3 %	98.7 %

Table 9-1 CPU Time required by components of Newmark Integration

The results indicate that the amount of time required for calculating the external loading and NEWMARK was negligible, thus they would not significantly reduce the total CPU usage by parallelisation. On the other hand, the RESVEC routine

accounts for the greater part of each iteration, irrespective of the problem size being attempted. Table 9-2 shows the distribution of CPU time for the main components of RESVEC, for four different idealisations.

Model Idealisation	CABLE	HANGER	FEDECK	TOWER	ANCFOR
3-9-3 (152 dof)	38 %	31 %	17 %	5 %	8 %
5-16-5 (251 dof)	39 %	35 %	18 %	3 %	5 %
8-27-8 (404 dof)	40 %	38 %	18 %	2 %	2 %
17-54-17 (809 dof)	41 %	39 %	18 %	1 %	1 %

Table 9-2 CPU Timing of Main components of RESVEC.

Before beginning to parallelise the existing serial codes, these were examined to identify any inherent parallelism that could be exploited. This inherent parallelism may exist explicitly within the structure of the various algorithms or implicitly due to the nature of the operation being performed by the algorithm.

As was shown earlier, the RESVEC routine is formed from five main components. It was found that each component was capable of running independently of the rest, thus allowing them to run concurrently within the network with the other components. Their respective contributions to the global restoring force vector are summed at the end of each iteration cycle.

Bearing this in mind and taking account of the above measured data, it was possible to determine an initial parallel topology for the network of processors that would be utilised in the subsequent parallelisation of the RESVEC subroutine. Figure 9-2, shows this preliminary network as it was constructed on the Parsytec Super Cluster, which readily allows the creation of two-dimensional networks.

The logic behind this orientation was as follows: since the subroutines CABLE and HANGER consume the greatest percentage of each iteration, they should be positioned adjacent to the Master processor. Similarly, the subroutines FEDECK, ANCFOR & TOWER respectively consume less CPU time than the former main

components and were correspondingly placed more remotely within the processor network.

Initially no great care was taken to optimise the amount of inter-processor communication or the potential for bottlenecks within the network. This could only be accurately gauged by measuring the performance of the network running specific problems, the results being used to optimise the network in a logical manner.

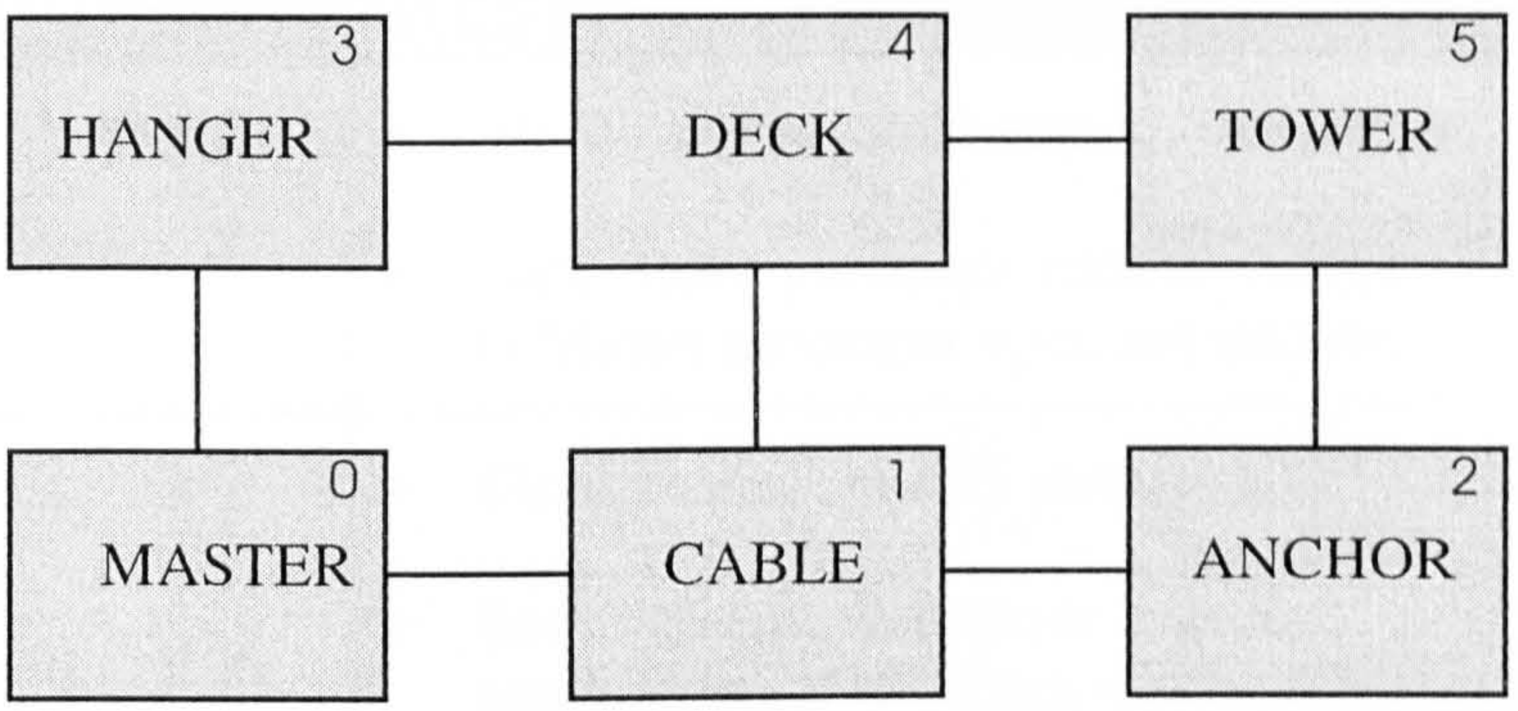


Figure 9-2 Preliminary Topology adopted within RESVEC

9.2.1 Parallelisation Strategies Developed for Numerical Integration

The following section describes the series of different parallelisation strategies attempted on the numerical integration algorithms, with the logic behind each strategy being discussed. Three main parallel strategies were implemented, the main salient points of each strategy being shown in Table 9-3.

Strategy	Processor Network	Main Features
Parallel_1	3 x 2	• All Slave processors sent all initial data. • Each Slave processor returns complete force vector to Master processor upon completion
Parallel_2	3 x 2	• Slave processors only sent essential data. • Each Slave processor returns only modified components of the force vector to Master processor.
Parallel_3	2 x 2	• Slave processors only sent essential data. • Each Slave processor only returns modified components of the force vector to Master processor.

Table 9-3 Strategies Adopted for the Parallelisation of the Numerical Integration Algorithm.

9.2.1.1 Identification of Communication Bottlenecks

Communication bottlenecks form when two or more Slave processors try to communicate simultaneously with another processor. The result is one processor communicating, while the rest are delayed for the duration of that communication.

However, there is always the potential that the formation of one bottleneck can have considerable knock-on effects within the global communication network, resulting in considerable delay to the total communication cycle. The goal in resolving this problem is not to totally remove the bottleneck problem, though this would be a considerable bonus if possible, but to distribute the delay caused by bottlenecks around the network in such a way as to minimise the total computational cycle time.

To better appreciate the level of communication within the networks and the potential for the formation of communication bottlenecks, every aspect of the calculations being performed for each of the constituent routines within the main algorithm was examined. The duration of the three main phases for each Slave processor were measured. The main phases were receiving data from the Master processor, performing the allocated calculation on the received data and finally communicating the solution back to the Master processor.

To illustrate the complexity of the task in determining the minimum computational cycle time, Figure 9-3 shows the significant difference in the computational time required by each of the main components of the algorithm as the size of the problem increases.

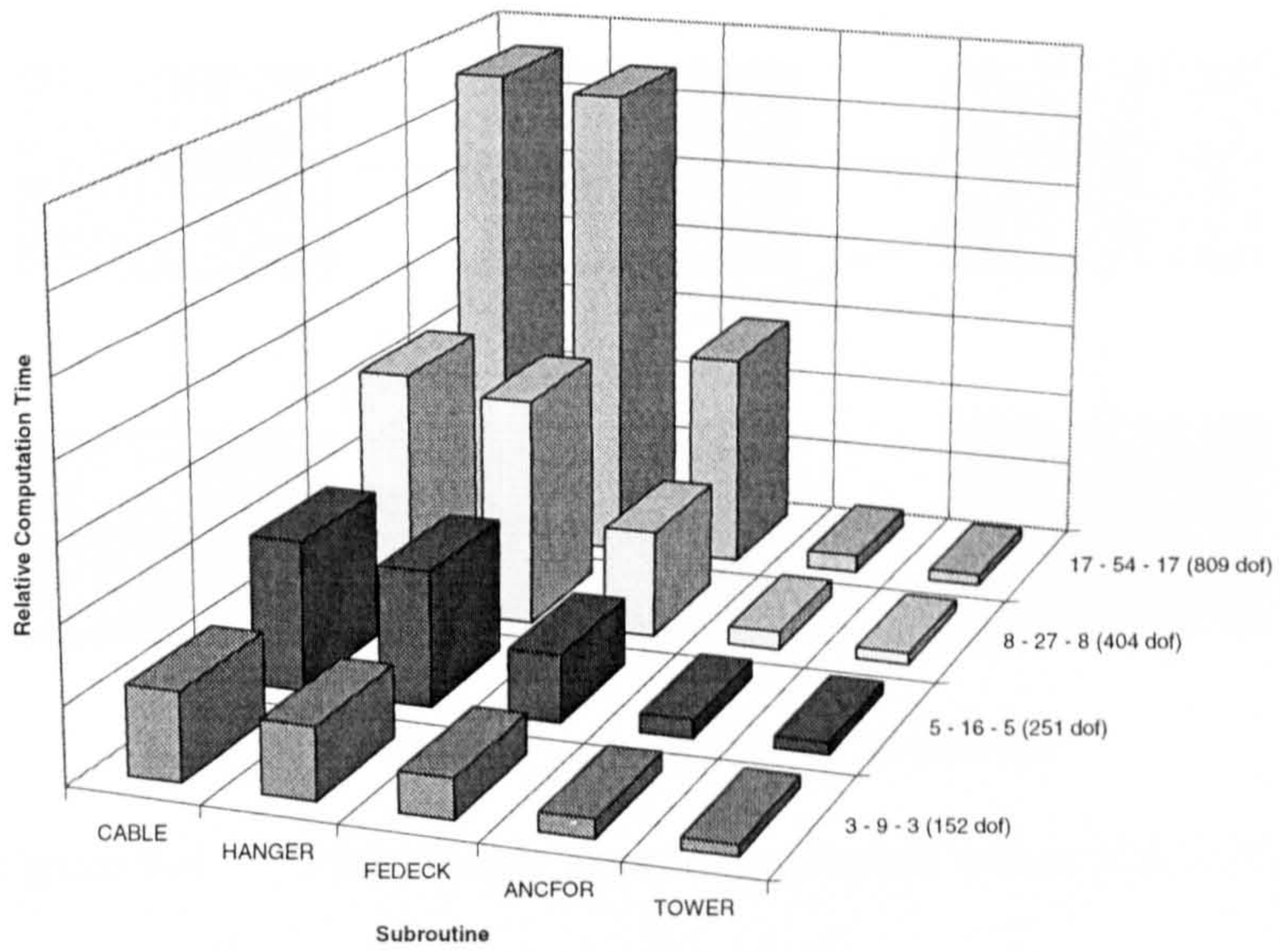
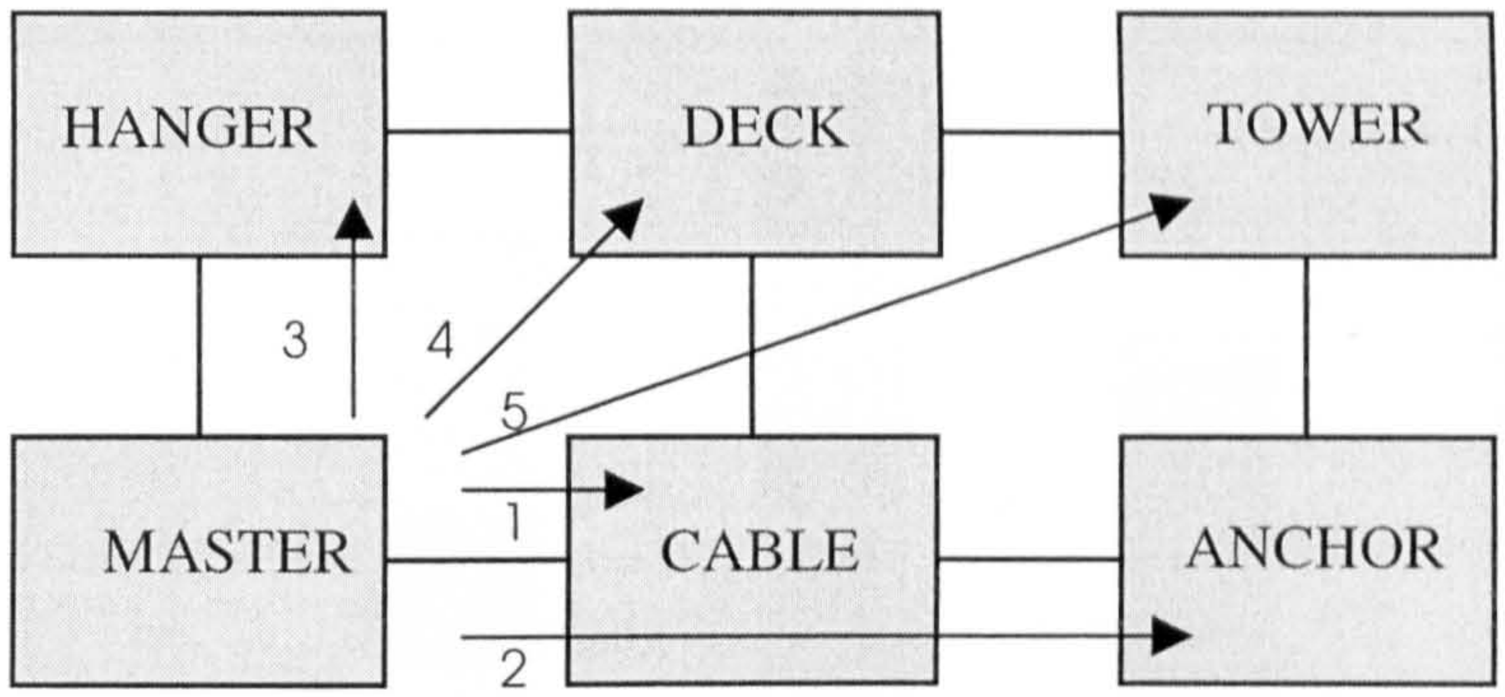
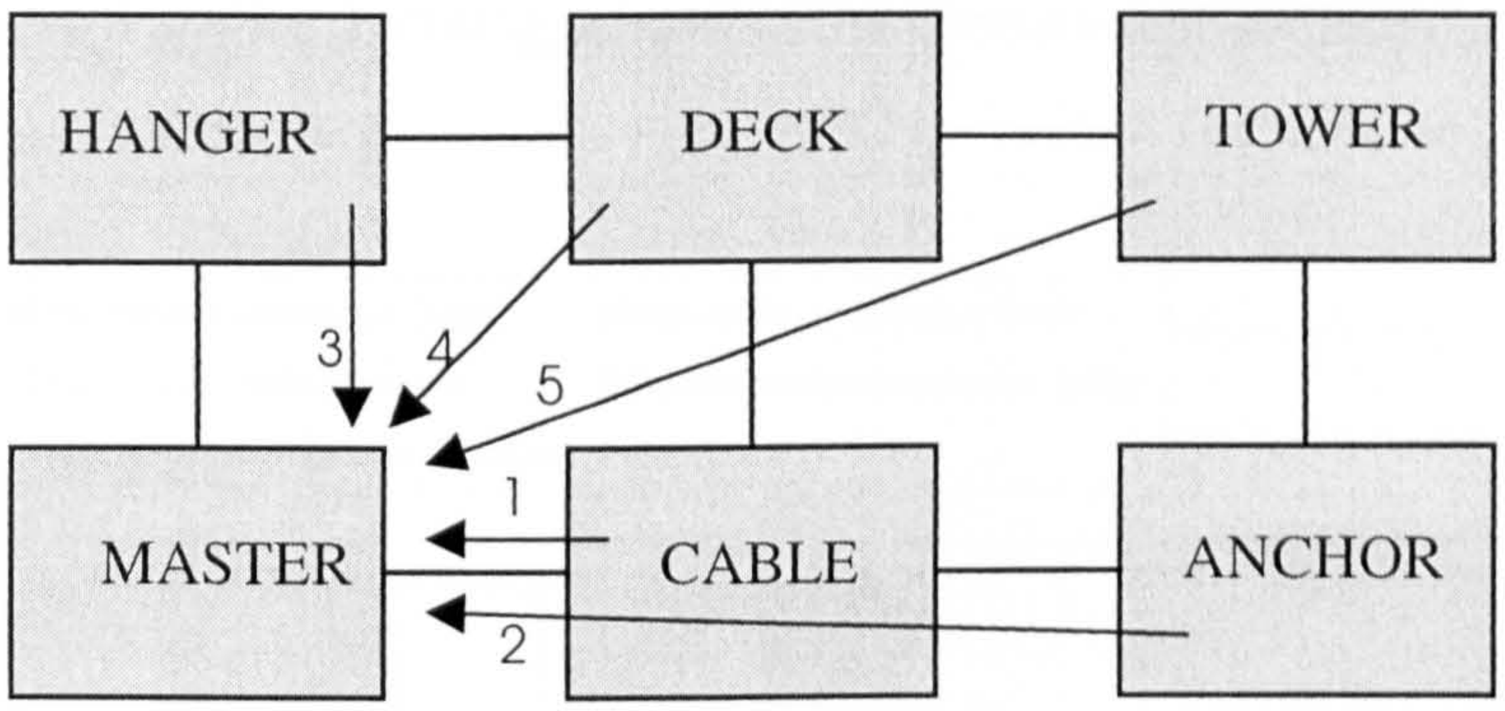


Figure 9-3 Calculation times for the main components of the numerical integration algorithm, for four different problem idealisations

Figure 9-4 shows the communication strategy used by the PARALLEL_1 implemented codes. As can be seen the Master initially communicates the input data to each Slave processors in sequence, then recovers the solutions from each processor in turn, using the Farm communication topology highlighted in Section 8.2.1



a) Master communicating to Slaves



b) Slaves communicating to Master

Figure 9-4 Preliminary Topology adopted within RESVEC

The results for the above strategy are shown in Figure 9-5 & Figure 9-6. As is evident, there are considerable communication delays within the network due to the formation of communication bottlenecks. The reason for the formation of these bottlenecks is due to the rigid and mechanical way in which the Farm topology communicates with each Slave processor in sequence according to their processor number. The consequence of this is the network delays ANCFOR while the Master waits for the results of CABLE. Also, FEDECK & TOWER are delayed while the Master waits for the communication from HANGER.

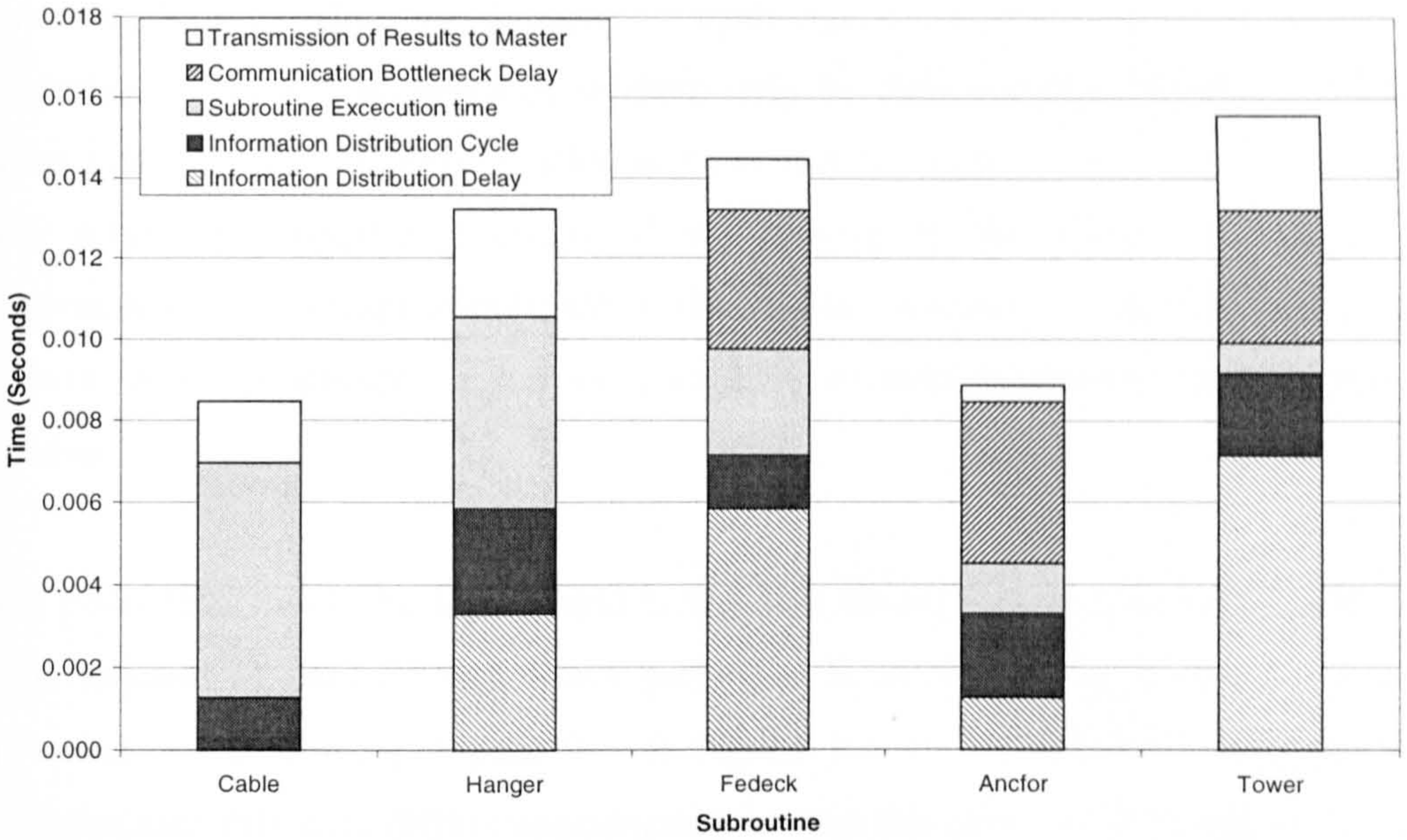


Figure 9-5 Parallel_1 strategy showing the communication delay for the 3-9-3 idealisation

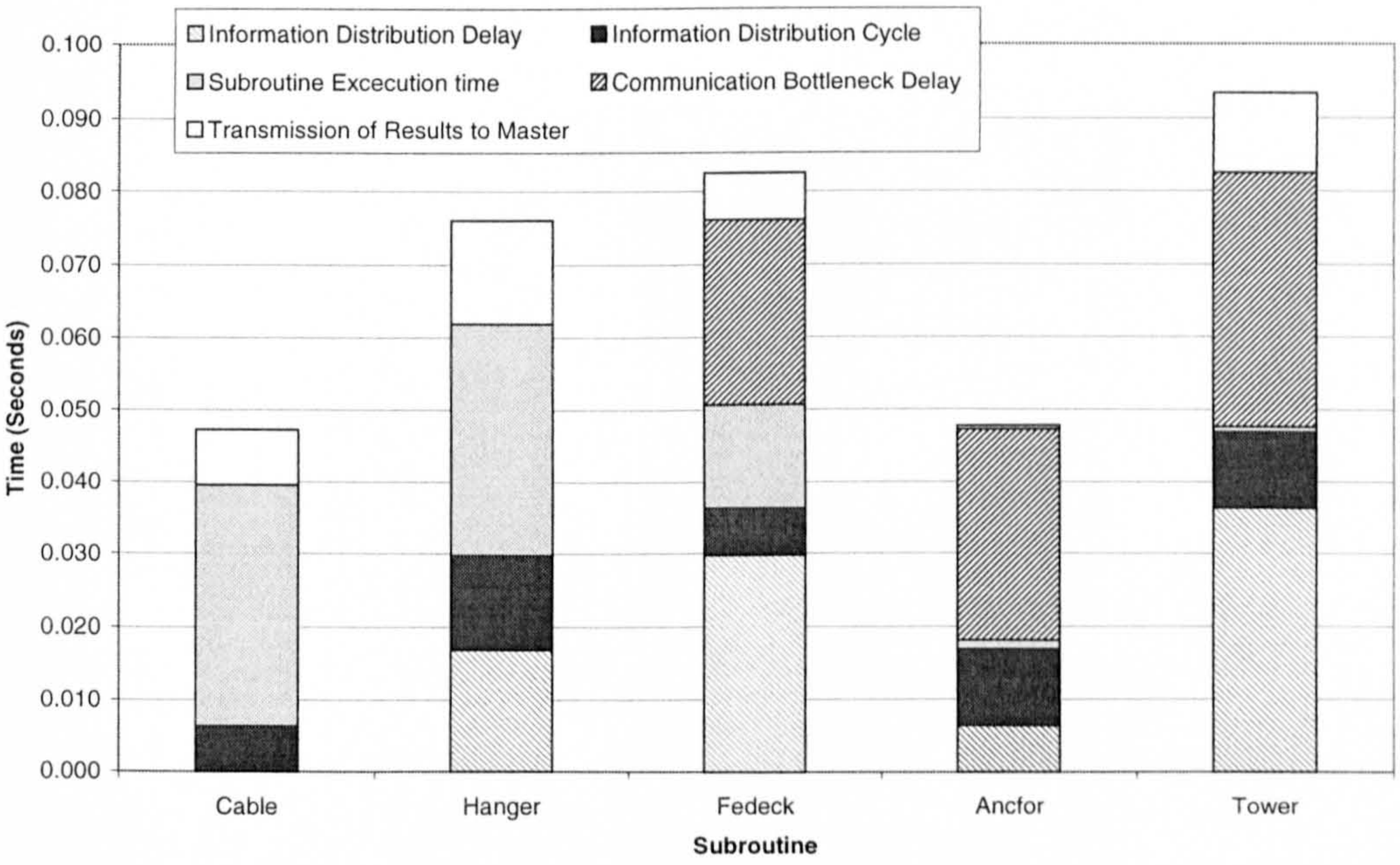


Figure 9-6 Parallel_1 strategy showing the communication delay for the 17-54-17 idealisation

This strategy allowed two main conclusions to be drawn that were applied to the subsequently refined strategies. Firstly, due to the duration of some of the inter-processor communications, the potential for formation of communication bottlenecks

was considerable. Thus the duration of each individual communication should be minimised. This was achieved by sending only the data essential for any individual Slave processor to complete its allocated calculation. In addition, each Slave would only return the modified sections of any vectors to the Master. Secondly, the communication topology should allow the Master processor to recover the results from the Slave processor that has just completed its task, irrespective of its processor number.

One point that should be highlighted is that this strategy communicates exactly the same amount of data to each Slave processor at the beginning of each iteration. However, by examining Figure 9-5 & Figure 9-6, it is clear that time taken to communicate this data differs considerably around the network. The reason for this was touched upon in Section 8.1, which presented the results of the system measurements, and concluded that the inter-processor communication rate is directly related to the Slave processors position and distance from the Master processor.

9.2.1.2 Optimisation of Communication Bottlenecks

One of the main features identified within the RESVEC routines was the amount of communication that was required to distribute properly and to recover the total solution at the end of each iteration.

The main algorithm and the constituent subroutines of RESVEC were examined to determine the minimum level of communication that would be required to perform an analysis. This yielded information about the arguments in the calling structure, their size and the possibility of classifying the numerous arguments into three types, Constant, Input or Output variables, Table 9-4.

The significance of these categories is evident from their names; the Constant arguments need only to be communicated once to each Slave processors. While the Input and Output variables require to be sent and received at the beginning and end of each iteration respectively.

COMPONENT	Number of Variables Needed	Number of Constant Variables	Number of Input Variables	Number of Output Variables
ANCHORS	8	6	1	1
CABLE	9	6	1	2
DECK	12	7	1	4
HANGER	15	10	2	3
TOWER	6	3	1	2

Table 9-4 Distribution of Arguments within the Newmark Algorithm

Table 9-4 gives an explicit indication of a clear program structure that could be exploited to minimise the communication cycle time. The communication cycle was further reduced by examining the variables to determine which pieces were required on each of the respective Slave processors, thus allowing the Master to communicate the absolute minimum amount of data.

The results in Table 9-5 indicate that each Slave processor that solves a specific component of RESVEC, neither requires all the information from the Input variables nor does it modify all the Output variables. Thus, the potential exists to optimise the topology of the algorithm and the processor network to minimise the solution.

COMPONENT	Number of Input Variables	Percentage of Input Variables Required	Number of Output Variables	Percentage of Output Variables Required
ANCHORS	1	41 %	1	31 %
CABLE	1	71 %	2	76 %
DECK	1	32 %	4	32%
HANGER	2	86 %	3	88 %
TOWER	1	10 %	2	22 %

Table 9-5 Percentage of variables required by components within RESVEC

Applying these improvements to the original computational strategy resulted in the Parallel_2 strategy. The results are shown in Figure 9-7 & Figure 9-8 for the 3-9-3 and 17-54-17 idealisations respectively. This strategy resulted in only one communication bottlenecks forming in the 3-9-3 Idealisation. However, no communication bottlenecks formed in the larger 17-54-17 idealisations as is shown in Figure 9-8.

The most significant improvement for this strategy over the Parallel_1 strategy is that the bottleneck delays within the network do not increase the total computational cycle time. Parallel_1 had large delays associated with FEDECK, ANCFOR & TOWER, all of which directly increased the total computational cycle time.

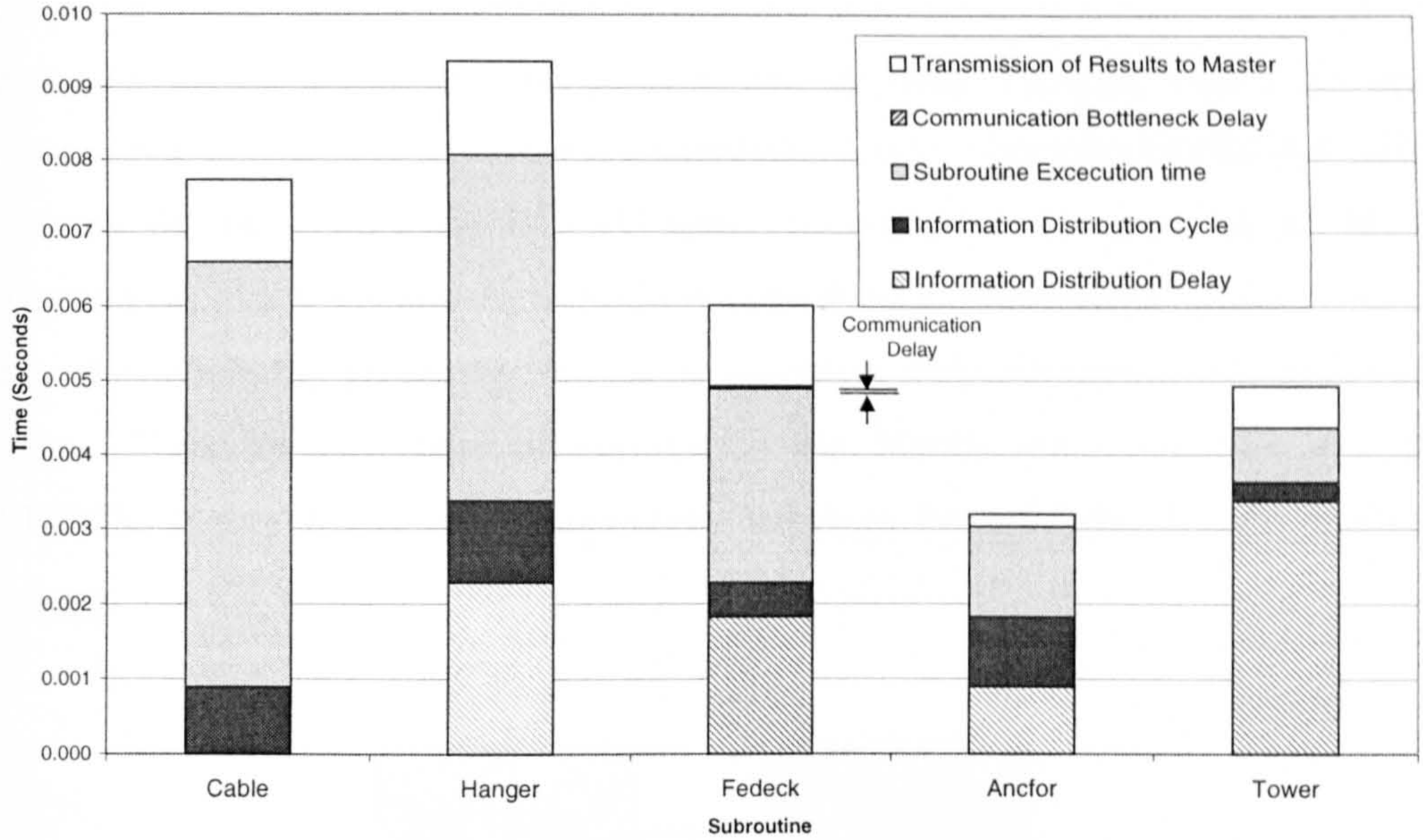


Figure 9-7 Parallel_2 strategy showing the communication delay for the 3-9-3 idealisation

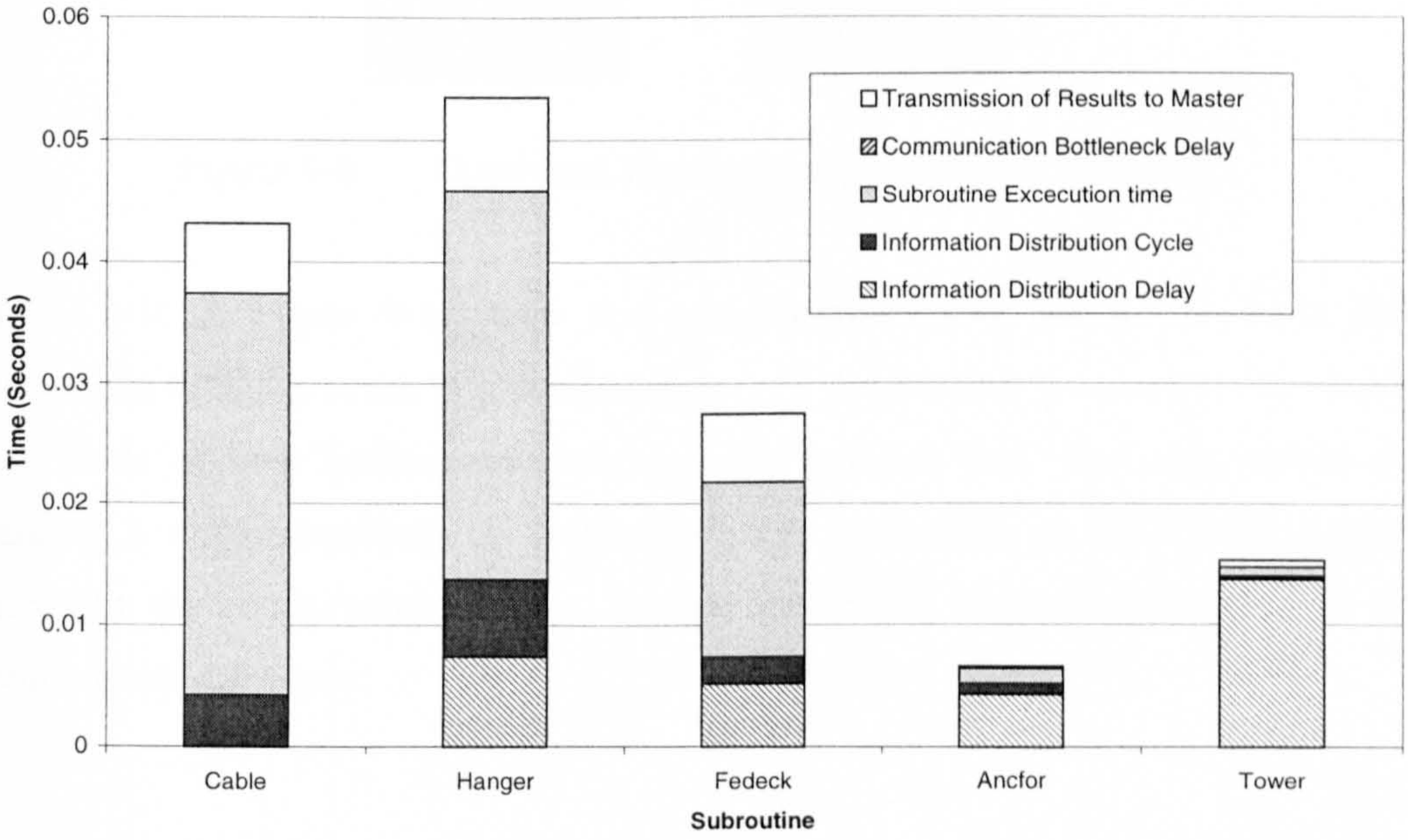


Figure 9-8 Parallel_2 strategy showing the communication delay for the 17-54-17 idealisation

These results were examined to identify any potential for further reduction of the CPU time required to perform an iteration. Since the time required to calculate and communicate the results of ANCFOR & TOWER is negligible compared to the other

components, they were combined to run on one Slave processor. The Master processor which is idle for the greater part of each iteration, would be more effectively utilised if allocated some computational task. Since HANGER & CABLE consume the largest amount of CPU time, placing either of these on the Master processor would automatically introduce considerable bottlenecks to the network, since the remaining processors would all complete their allocated tasks before the Master. Thus by a process of elimination the Master processor was allocated FEDECK. Figure 9-9, shows the optimised topology implemented for the parallel_3 strategy.

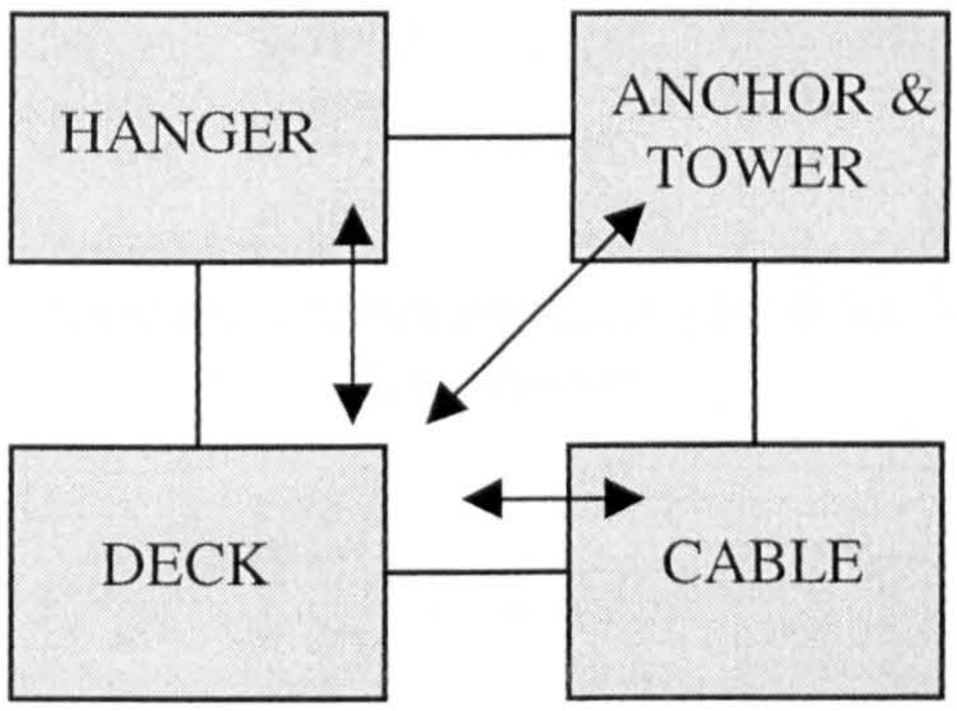


Figure 9-9 Optimised Topology adopted within RESVEC

Figure 9-10 & Figure 9-11 show that two communication bottlenecks have been introduced to the network in a similar manner to the Parallel_2 strategy, the relative magnitude of both bottlenecks reducing with problem size. However, unlike the Parallel_2 implementation, the communication bottleneck on HANGER directly increases the computational cycle, though this delay becomes negligible as the problem size increases.

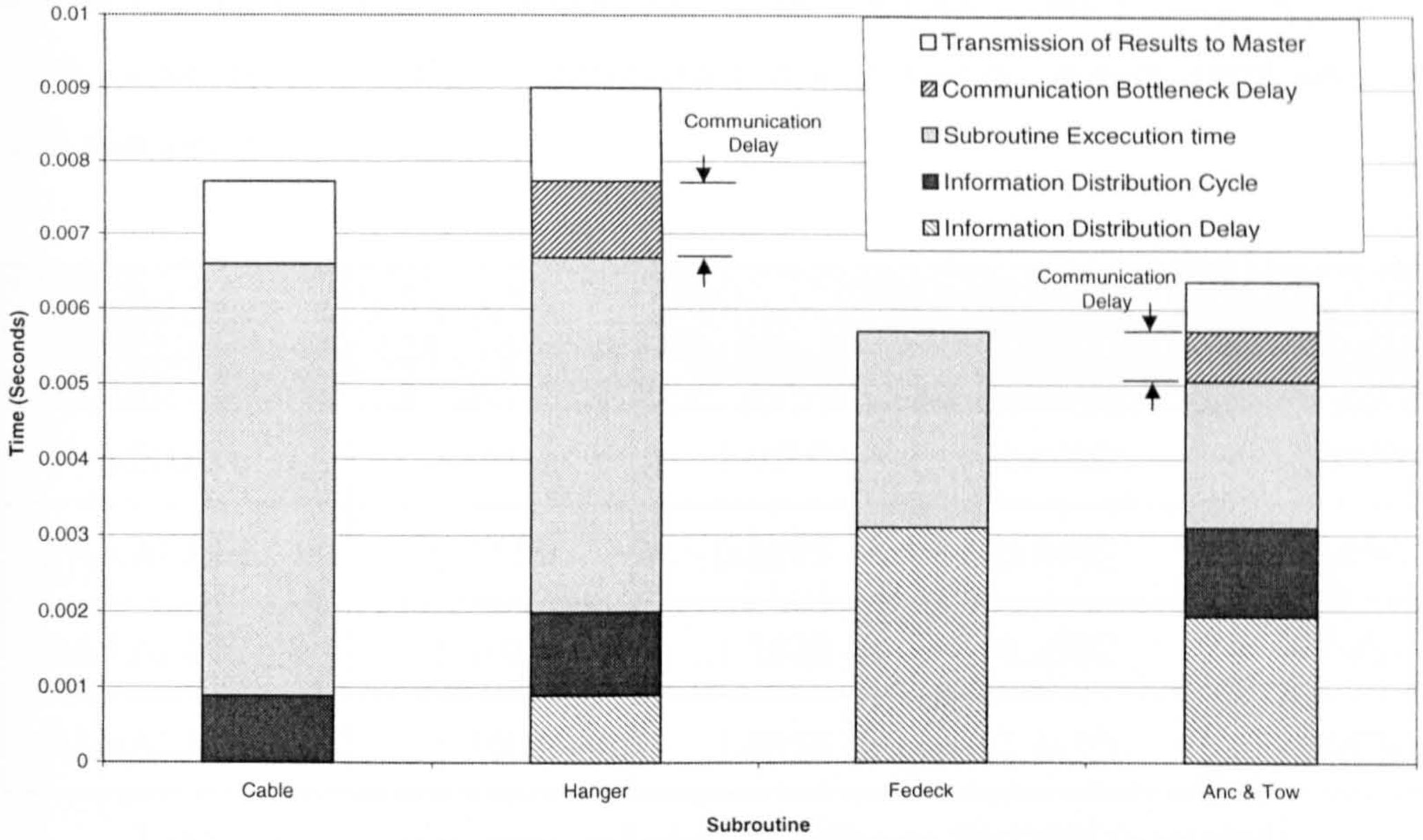


Figure 9-10 Optimised Communication network on four processors 3-9-3 Idealisation

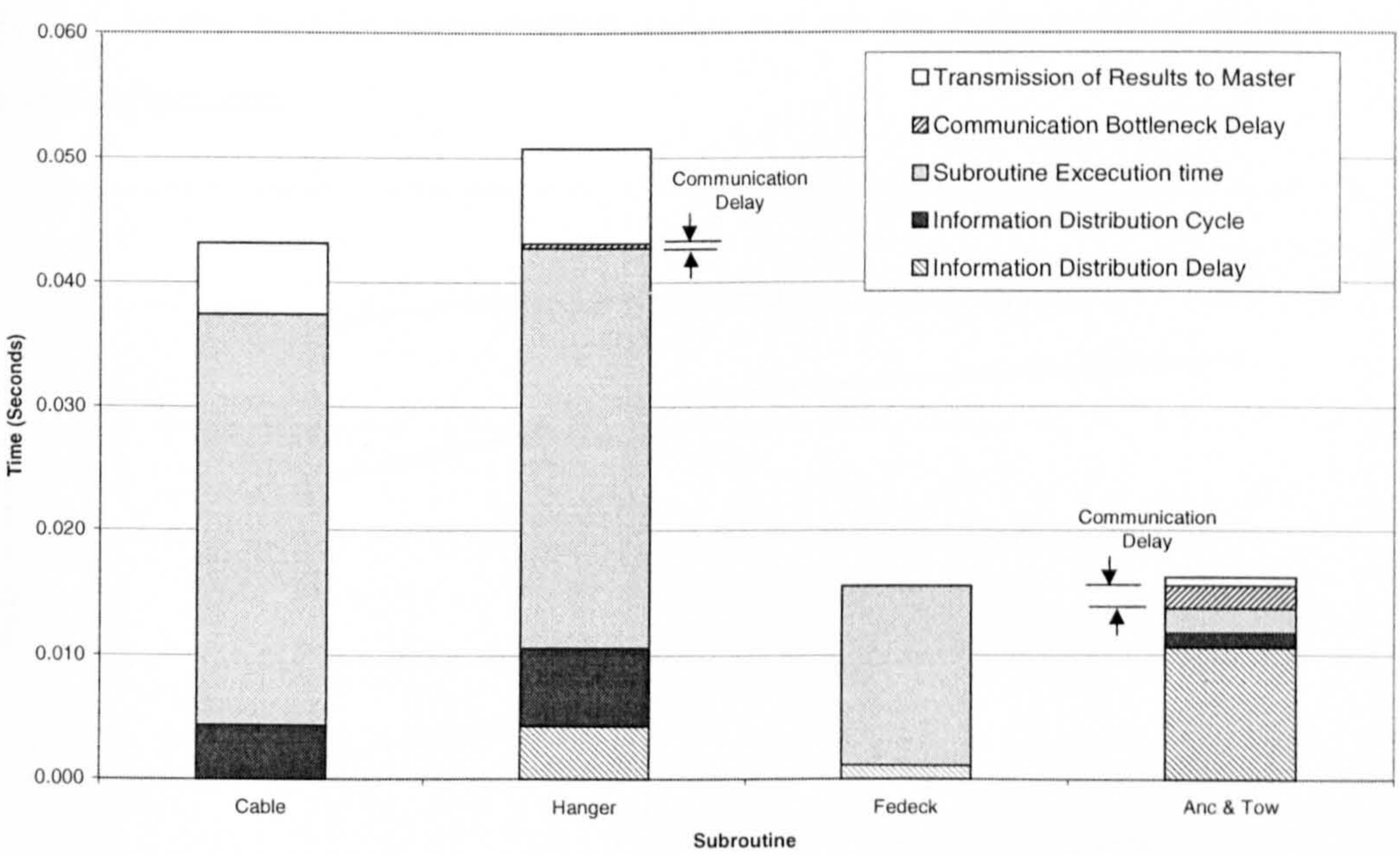


Figure 9-11 Optimised Communication network on four processors 17-54-17 Idealisation

9.2.2 Conclusions for Parallelisation of Time History Analysis

The results of the three implemented parallel codes are given in Table 9-6 and are shown in graphical form in Figure 9-12. The CPU times quoted were obtained by

running the respective implementations on the Parsytec Super Cluster Model 64, which is an electronically re-configurable multiprocessor system combining 64, T800 transputers.

RESVEC Version	3 - 9 - 3 (152 dof)	5 - 16 - 5 (251 dof)	8 - 27 - 8 (404 dof)	17 - 54 – 17 (809 dof)
SERIAL	1.0000	1.0000	1.0000	1.0000
PARALLEL_1	0.6313	0.6775	0.7407	0.8969
PARALLEL_2	1.1025	1.1520	1.2500	1.3100
PARALLEL_3	1.3602	1.4058	1.4153	1.4570

Table 9-6 Performance Index for different RESVEC strategies.

To facilitate the comparison of results a Performance Index is quoted. The Performance Index gives the ratio of CPU time required by the original serial version divided by the parallel implemented codes run on the same system, for the same model refinement.

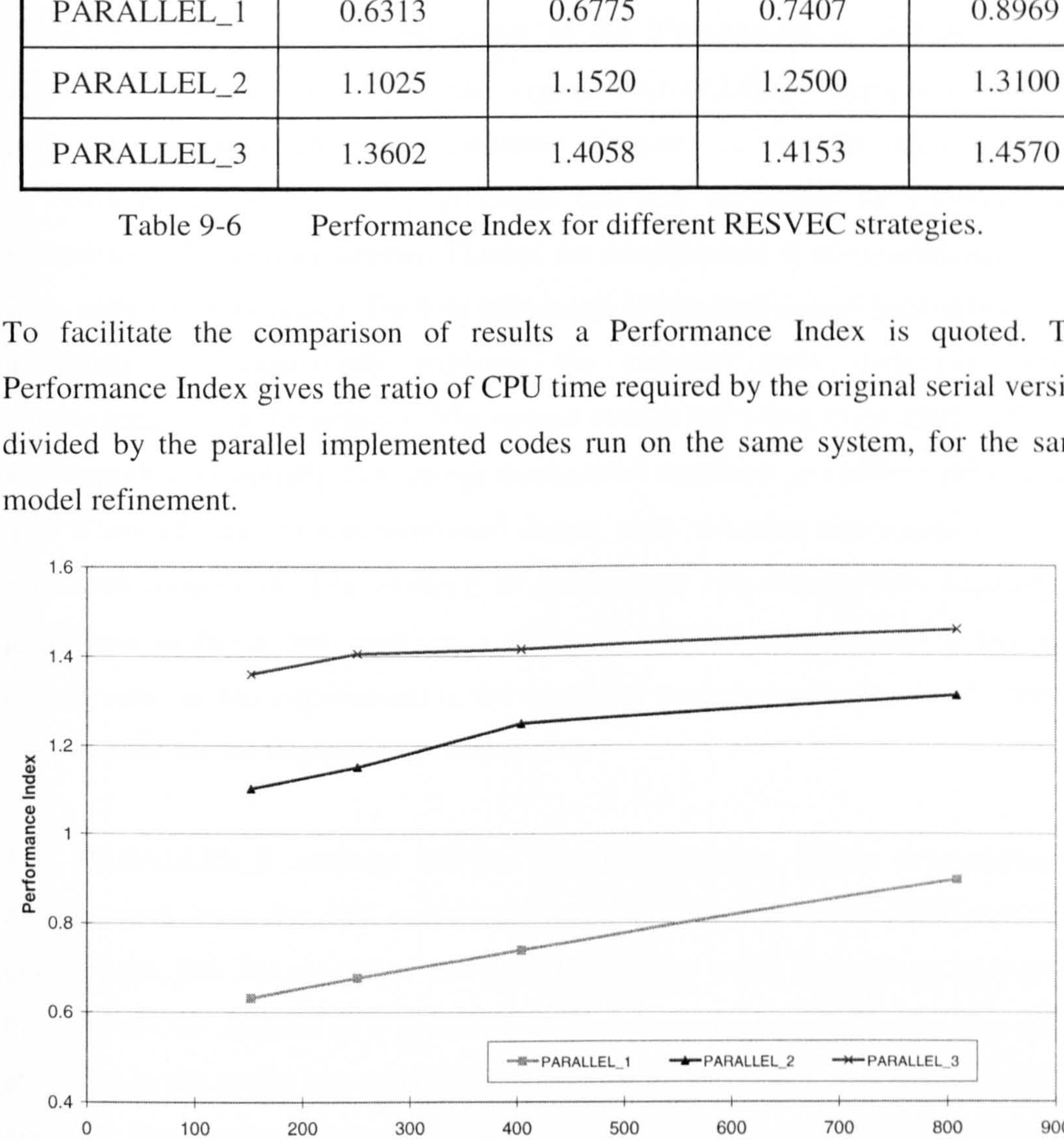


Figure 9-12 Performance Index for different RESVEC strategies.

There are several general observations that can be drawn form the above results. Firstly, the various implementations indicate that the performance indices increase with the size of problem being attempted. This improvement is due to the gradual reduction of communication bottlenecks between the main routines of the algorithm. Since each routine gradually takes longer to complete its allocated task and has its own unique amount of data to communicate, this naturally introduces idle periods within the network, allowing it to communicate in an orderly fashion.

Secondly, the refinements introduced to the PARALLEL_2 strategy show a considerable improvement from the original PARALLEL_1 implementation. The main changes associated with this implementation are, allowing the Master processor to communicate to any Slave processor that had completed its allocated task, irrespective of processor number. Thirdly, the minimisation of communication to and from each Slave processor. The first refinement clearly had a significant effect on the algorithm by considerably reducing the communication bottleneck delays experienced within the network. The second change has had a more subtle effect on the algorithm; essentially this change rescheduled the Slave processor's execution of their allocated task. As was mentioned earlier, each processor was uniquely sent the minimum amount of data required to perform its calculation. This allowed the processors to begin their tasks at a different relative point in time to the other components, as was experienced in the originally implemented codes, thus reducing the potential for the formation of bottlenecks.

The PARALLEL_3 strategy of the algorithm strategy shows a considerable improvement over the two previously attempted methods. The only significant change was, that the algorithm was coded to run on a 2 x 2 processor network as opposed to the original 3 x 2 network. The improvement is totally due to the reduction in the global communication within the network. This strategy removes the need for the distribution and recovery of data to and from FEDECK. A further reduction is also introduced by combining the communications for ANCFOR & TOWER into one large communication.

This highlights a significant point that is often overlooked when parallelising algorithms. In attempting to break algorithms down into the smallest possible

structures, programmers generally do so without due regard to the amount of additional inter-processor communication that will be required as a direct consequence. The RESVEC algorithm emphasises the fine balance between the level of subdivision within an algorithm and the communication that will be consequently required.

9.3 Parallelisation of Eigenvalue Analysis

The following section will discuss in detail the work undertaken to parallelise the Simultaneous Iteration algorithm within the ANSUSP program. The structure of the algorithm will be discussed, with particular attention being focused on the areas of greatest computational intensity, with any inherent parallel structures being highlighted for utilisation in the subsequent parallelisation.

9.3.1 Outline of Eigenvalue Analysis Algorithm

The eigenvalue extraction algorithm used within the ANSUSP program is the Simultaneous Iteration method (SI) developed by Jennings [62]. This chapter will concentrate on the programming aspects of the method and not the theoretical basis, which is presented briefly in Section 3.2.2

The structure of the Simultaneous Iteration algorithm consists of three main sections of subroutines, each with a unique task and algorithm topology.

i. The global structural stiffness matrix is decomposed using the Cholesky LL^T decomposition method, and an initial random trial vector is calculated.
ii. The algorithm then iterates (using the seven most computationally intensive subroutines of the algorithm) until the required eigenvalues and eigenvectors are extracted with satisfactory accuracy.
iii. The algorithm then back substitutes and mass normalises the final converged solution for the mode shapes.

Figure 9-13 below shows the distribution of computational effort among the three main sections of the SI algorithm measured on a serial computer system, for the four different problems sizes being considered. Table 9-7shows the range of problem characteristics that would commonly be attempted using these algorithms.

The results of Figure 9-13 allow two conclusions to be draw; firstly they indicate that the distribution of computational effort among the main sections of the algorithm are in general terms, totally non-sensitive to the problem size being attempted. Secondly, Figure 9-13 shows that the iterative section consistently accounts for over 90% of the

algorithm duration. Thus the most effective way of improving the performance of entire algorithm would be to successfully parallelise the iterative component.

Problem Identification	Number of Equations. (N)	Number of Modes required accurately †.(NRQD)
(N=116, M=11)	116	6
(N=116, M=76)	116	66
(N=773, M=15)	773	10
(N=773, M=34)	773	24

† The number of iteration vectors `M` used within the SI routine is determined by `M=NRQD+5` or `M=NRQD+10`, depending on the size of `NRQD`, the additional iteration vectors are used to speed up convergence.

Table 9-7 The various models attempted

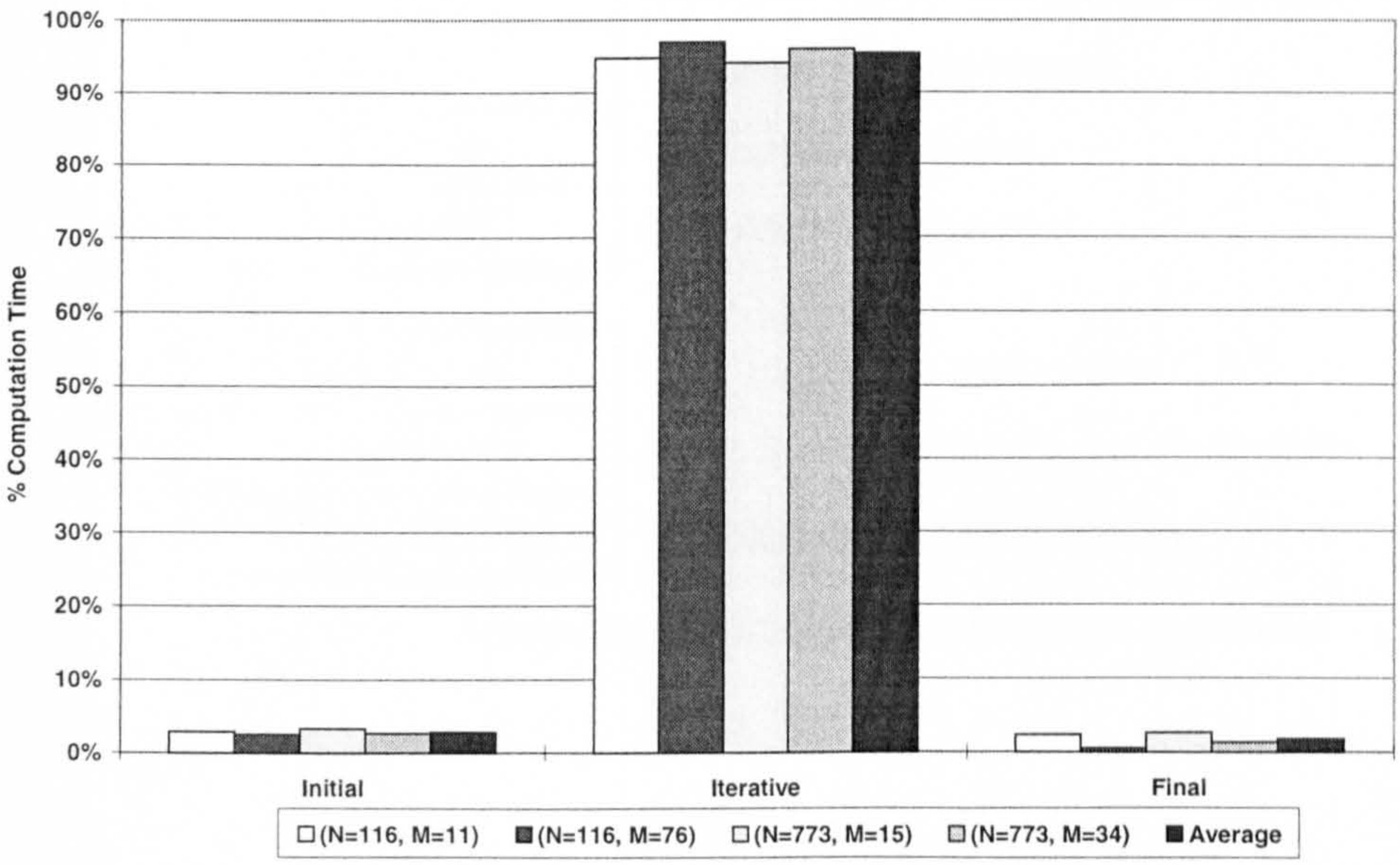


Figure 9-13 Comparative results for the main sections of significance within the Simultaneous Iteration Algorithm

9.3.1.1 Identification of significant areas within Iterative section

As was highlighted previously, the Simultaneous Iteration algorithm is constructed from three main sections of subroutines, the structure of the algorithm is shown schematically in Figure 9-14. The results from the previous section indicate that the vast majority of the computational effort (approximately 90%) for any problem is consumed within the iterative section of the algorithm. Figure 9-15(a, b, c & d) below show the distribution of computational effort among the seven constituent

subroutines of the iterative section, for the four typical problems being considered, Table 9-7. The distributions of computational effort for all the constituent routines within the iterative section are summarised in Figure 9-16.

(i)

REDUCE	Cholesky LL^T decomposition of stiffness matrix
TRIVEC4	Generate random trial vectors
ORTHOG	Orthogonalises vectors

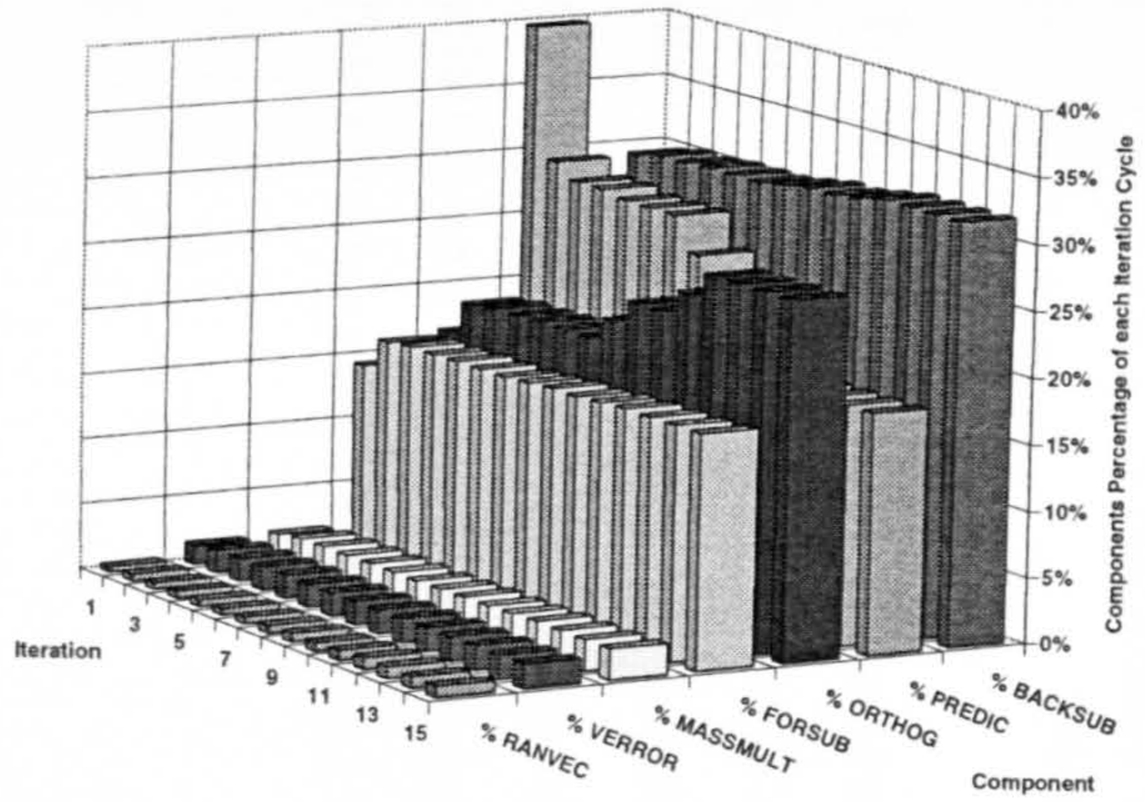
(ii)

BACKSUB	Back substitutes trial vectors
MASSMULT	Product of mass matrix and trial vector
FORSUB	Forward substitutes trial vectors
PREDIC	Generates eigenvalue prediction matrix
RANVEC	Generates random vectors
ORTHOG	Orthogonalises vectors
VERROR	Calculates vector errors

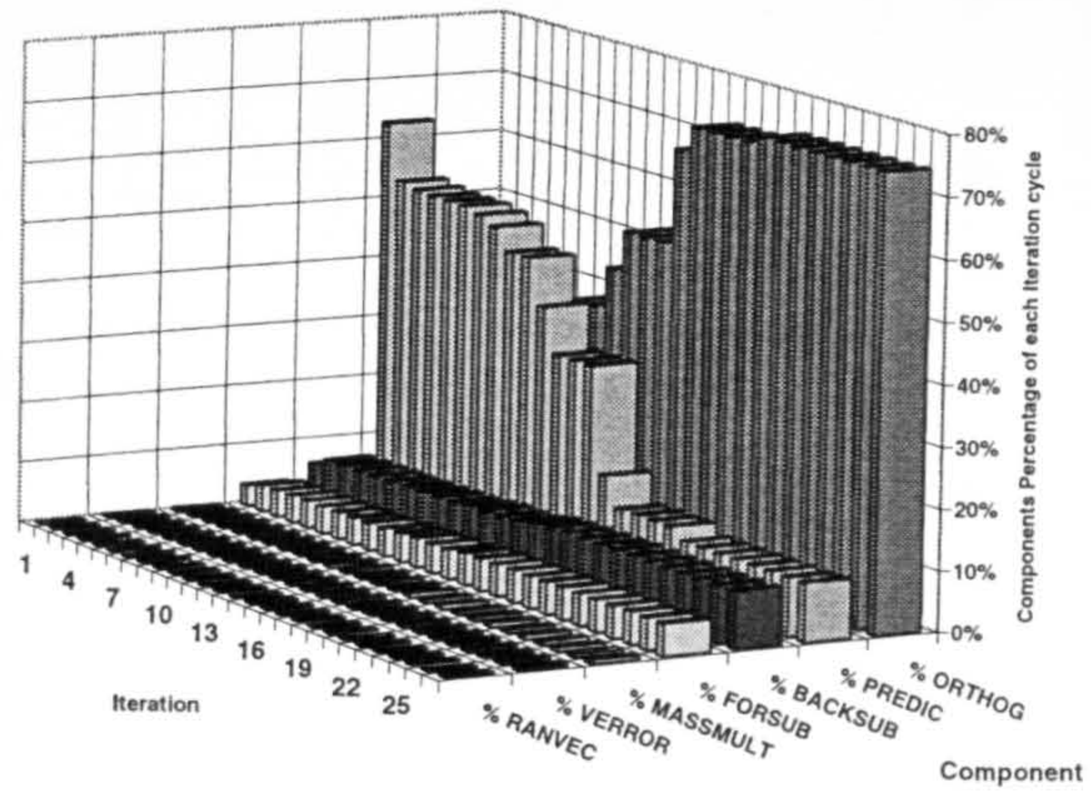
(iii)

BACKSUB	Back substitutes trial vectors
MASSMULT	Product of mass matrix and final vector
NORMALISATION	Normalisation of final vectors

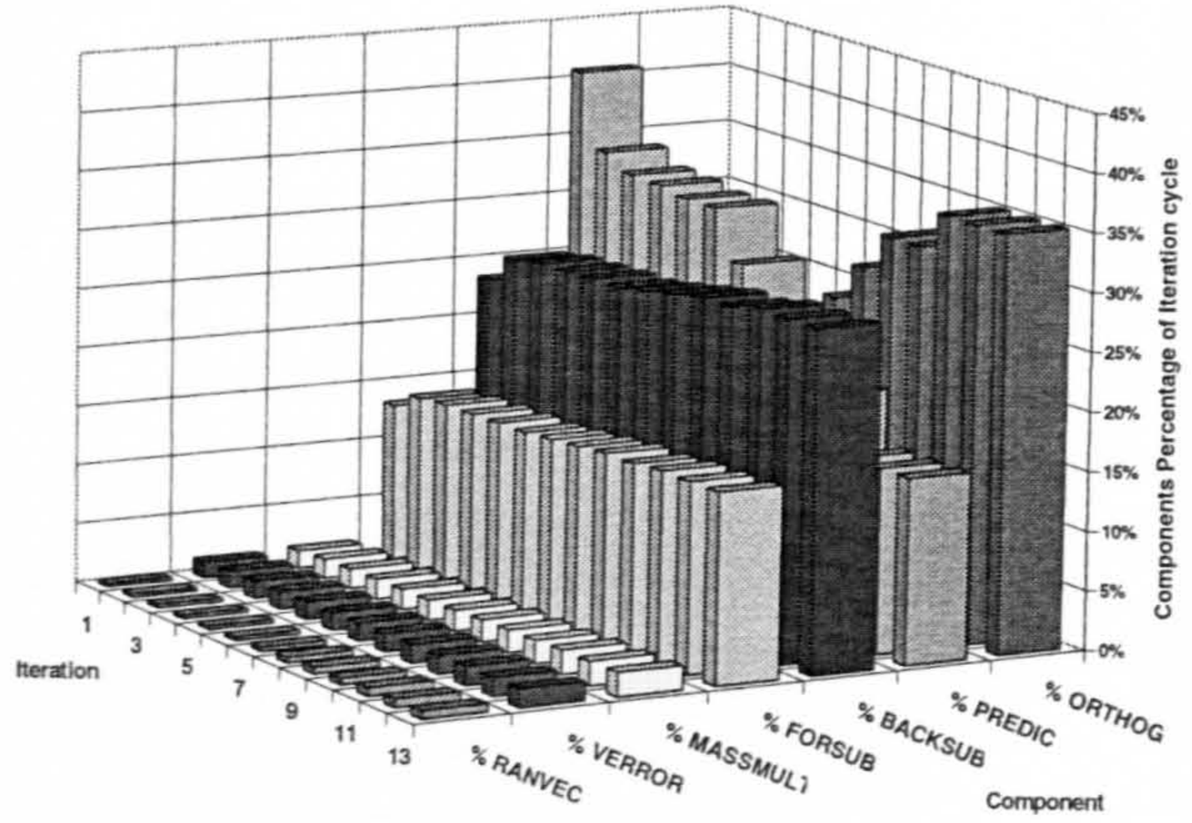
Figure 9-14 Schematic of Simultaneous Iteration Algorithm



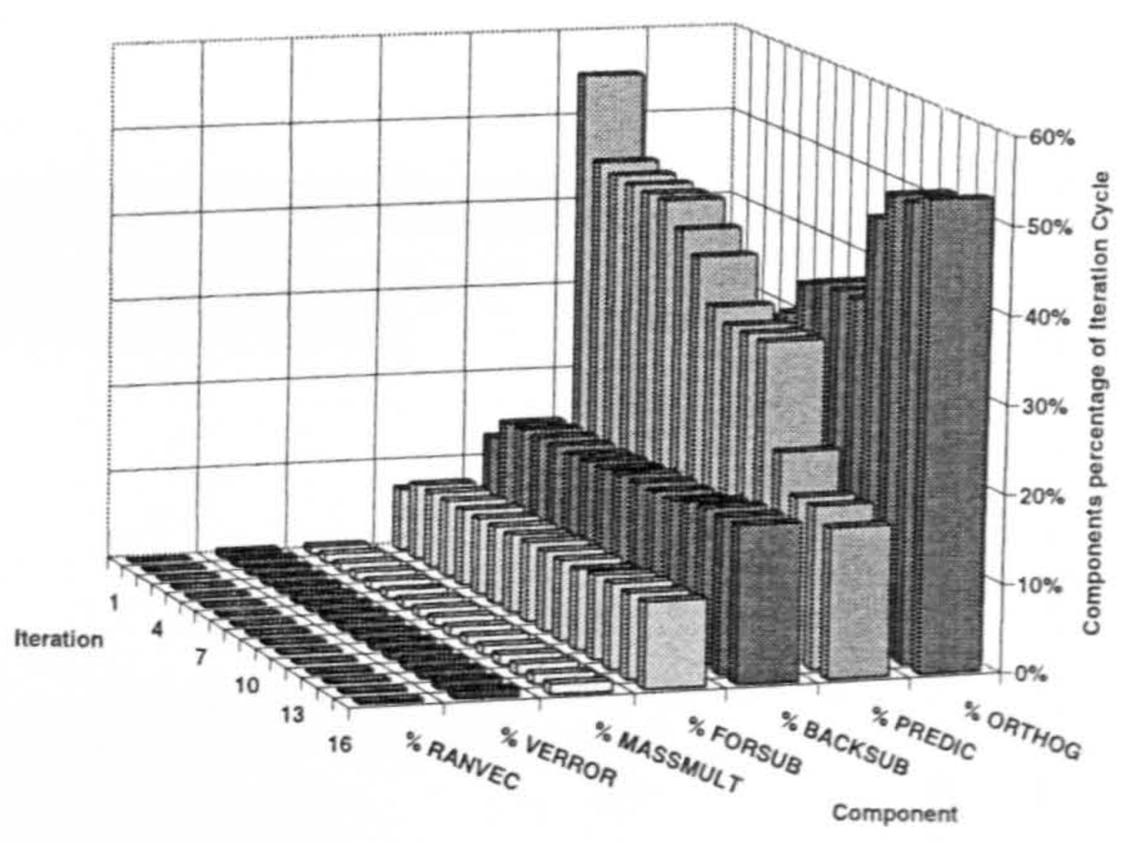
(a)



(b)



(c)



(d)

Figure 9-15 Distribution of Computational Effort within Iterative Section of Algorithm (a)(N=116, M=11), (b)(N=116, M=76), (c)(N=773, M=15), (d)(N=773, M=34) problems respectively

By examining Figure 9-15 & Figure 9-16 it is evident that the majority of the computational effort within this section of the algorithm is consumed by four main subroutines PREDIC, ORTHOG, BACKSUB and FORSUB. By examination of the results in Figure 9-15, it is also evident that the relative computational time spent on each of the two main subroutines PREDIC and ORTHOG varies as the iteration number increases for all models (i.e. the stage within the overall solution).

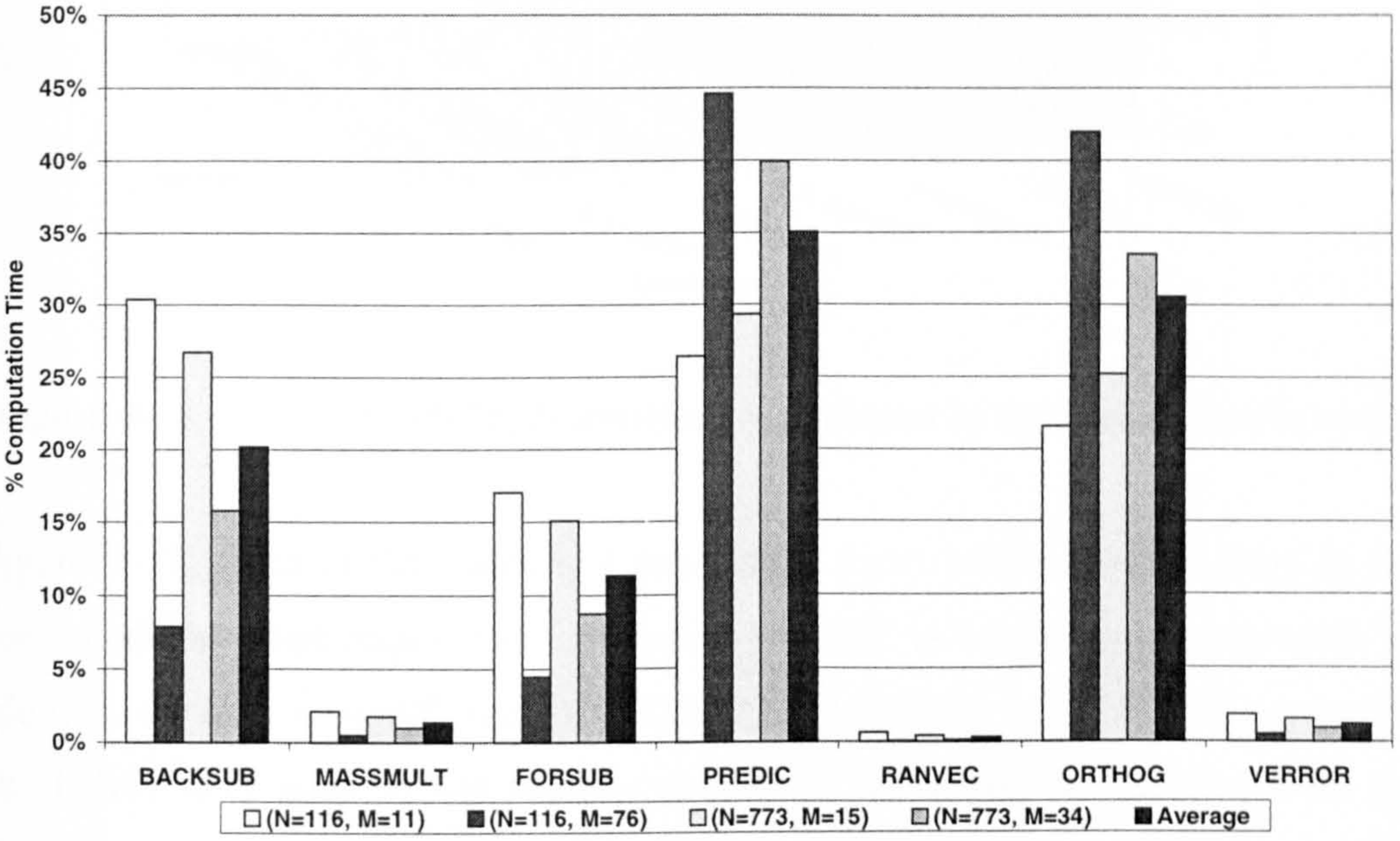


Figure 9-16 Comparative results of the main subroutines within the iterative section of the SI Algorithm.

To understand better the distribution of computational effort among the different routines, the results for the (N=116, M=76) analysis were normalised with respect to the total time required to perform the first iteration cycle. This revealed the relative significance of the main components with respect to the stage of the solution, and the relative time required performing each subsequent iterative cycle with respect to the first iteration cycle, Figure 9-17.

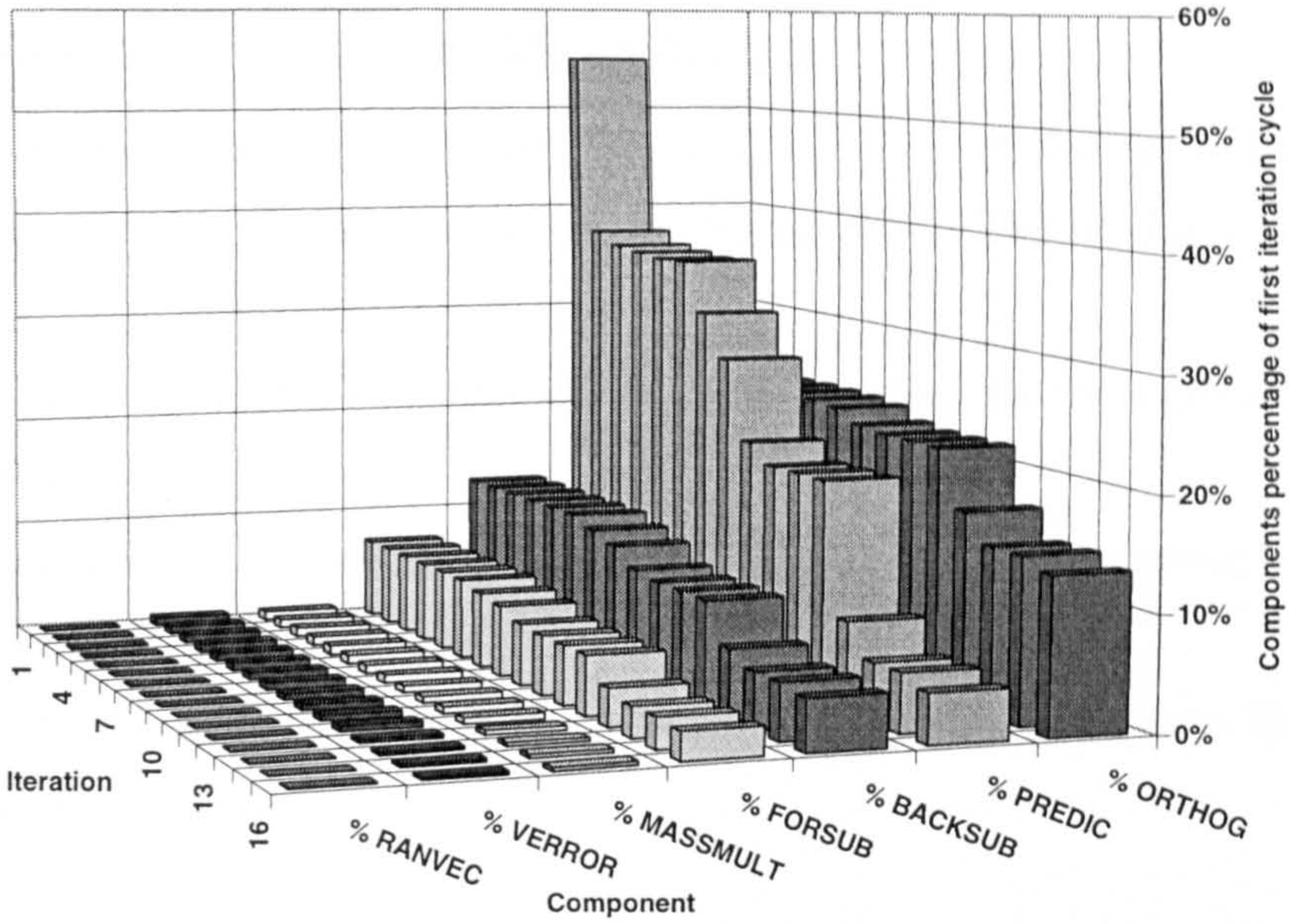


Figure 9-17 (N=116, M=76) Normalised with respect to the initial iterative cycle

Figure 9-17 indicates that there is a substantial (from ≈55% to ≈5%) drop in the computational effort required in calculating PREDIC as the solution progresses. In addition, the relative significance of ORTHOG, BACKSUB and FORSUB appears to be slightly decreasing as the solution progresses, as might be expected from the characteristics of the method.

As the algorithm extracts eigenvalues to the required accuracy, the solution stops operating on the eigenvectors associated with these eigenvalues. This in some routines leads to a significant reduction in the computational effort required for the next iteration.

It should be noted that the results presented in Figure 9-17 are specifically for the (N=116,M=76) problem. However, upon further investigation into the distribution of computational effort at different stages, for various problem sizes, it was found that the same general conclusions were correct for the range of typical problems.

Thus the routines that are most computationally intensive and require to be massively parallelised are PREDIC and ORTHOG, while the routines BACKSUB and FORSUB would benefit from being partially parallelised.

9.3.2 PREDIC - The Prediction Matrix Routine

This routine on average accounts for 35% (Figure 9-16) of the entire calculation effort required in the iterative section of the SI algorithm. The routine is structured into three clearly defined sections.

i. Calculate the eigenvalue predictions corresponding to the current iteration vectors.

ii. Calculate the interaction matrix, which is used to modify the current set of trial vectors.

iii. Sort the calculated eigenvalues and associated eigenvectors into descending order of magnitude.

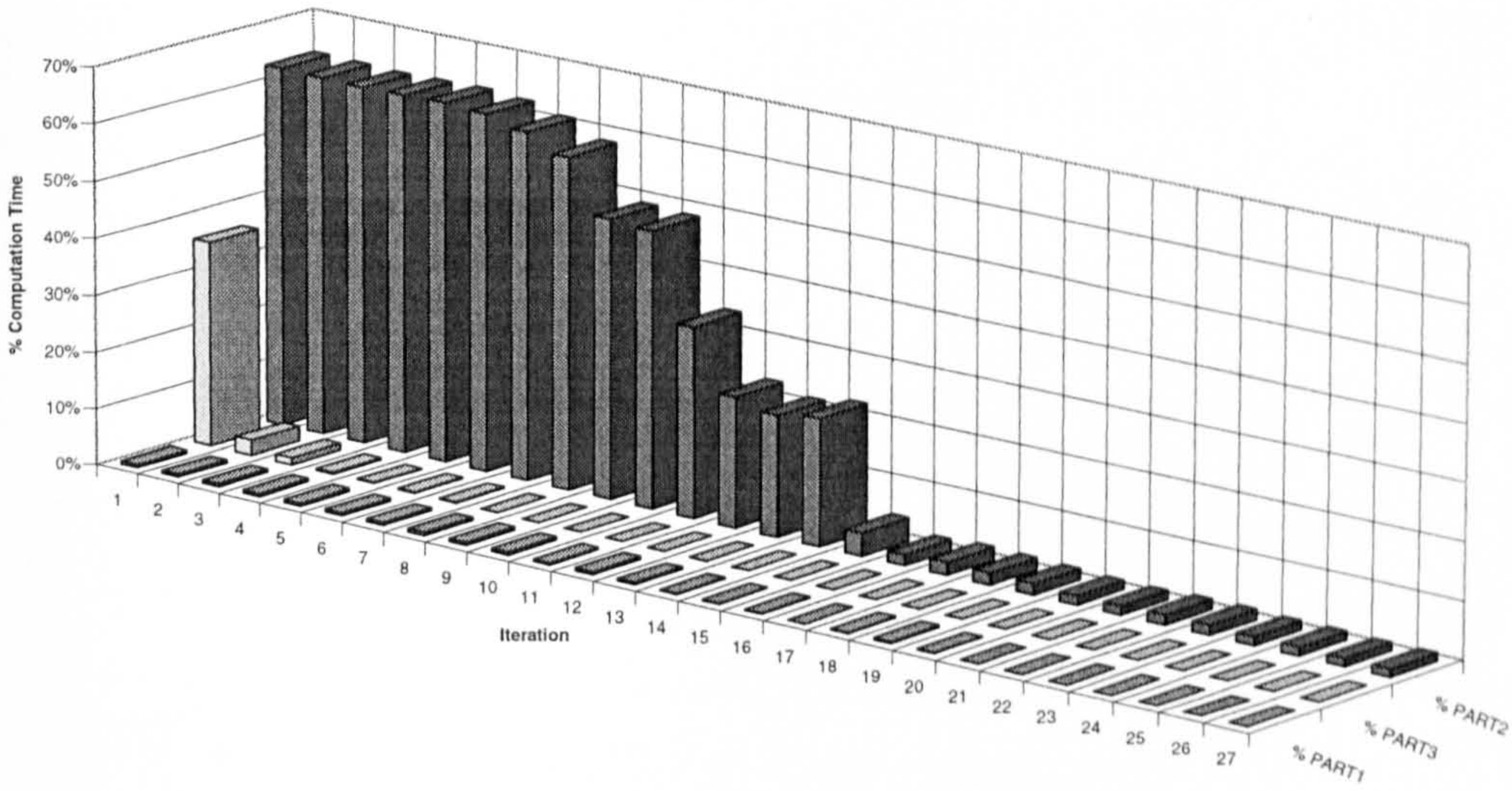


Figure 9-18 The relative significance of the three components of PREDIC, normalised with respect to the first cycle within PREDIC (N=116, M=11).

Figure 9-18 allows several conclusions to be draw:

Part I of the subroutine is relatively insignificant when compared with the other two components, thus would not benefit from being parallelised.

Part II, consumes the majority of the cycle time for each iteration. Thus the greatest potential for a significant overall improvement would be realised by parallelising this section as much a possible.

Part III, is highly significant for the first iteration, the level of relative significance reducing rapidly with iteration until it becomes negligibly small. Thus any strategy required to parallelise this section would need to be developed with particular attention being given to greater solution efficiency while within the first few iteration cycles.

9.3.2.1 Determination of Computational Effort within Part II

To understand better how the routines might be parallelised, the serial coding was examined to determine any clearly defined program structures that could be utilised in the parallelisation of the serial coding, Program 9-1.

```
C
C       PART II
C
      DO 4 I=LOCK,M
        IF(I.EQ.1) GOTO 4
        II=I-1
      DO 400 J=LOCK,II
        EL=0.0D0
        DO 3 K=1,N
    3   EL=EL+U(K,I)*V(K,J)
        EL=-2.0D0*EL
        Q=DBD(I)-DBD(J)
        EL=ALIN(EL,Q,DBD(I),INT)
        CALL VECSUM(V,W,I,J,-EL,N,M )
        CALL VECSUM(V,W,J,I,EL,N,M)
  400   CONTINUE
    4 CONTINUE
C
```

Program 9-1

Where

N	The number of active degrees of freedom.
M	The number of iteration trial vectors being used.
U(N,M)	Array which holds trial vectors.
V(N,M)	Array which holds the new product AU.
W(N,M)	Array which contains updated trial vectors.
DBD(M)	Vector which contains the current eigenvalue predictions.
LOCK	Lower limit of active trial vectors (i.e. inactive vectors no longer included in some operations).
INT	Iteration cycle number.

By examining the structure of PART II, it was clear that the calculation of the updated trial vectors, W is achieved using nested DO loops. The outer DO loop runs from the number of vectors calculated at the end of the previous cycle to the maximum number of trial vectors required (i.e. LOCK to M).

The inner DO loop cycles from the number of vectors calculated at the end of the previous cycle to the current value minus one of the outer DO loop. Each cycle of the inner DO loop involves two calls to the external routine VECSUM, Program 9-2.

This routine modifies a vector of W by adding the product of vector V multiplied by a scalar constant CONST.

```
      SUBROUTINE VECSUM(V,W,K,KK,CONST,N,M)
C
      DO 1 I=1,N
    1 W(I,K)=W(I,K)+CONST*V(I,KK)
      RETURN
      END
```

Program 9-2

To determine the computational effort required in calculating this routine a numerical example was employed. Consider the routine calculating on the first iteration, with no vectors having been accurately extracted (i.e. LOCK=1).

Outer I	II	Inner J	EL U(K,I)·V(K,J)	W	Vectors Modified
1	---	---	---	---	---
2	1	1	$\hat{U}(2) \bullet \hat{V}(1)$	$\hat{W}(2) = \hat{W}(2) + \hat{V}(1) \bullet -EL$ $\hat{W}(1) = \hat{W}(1) + \hat{V}(2) \bullet EL$	1..2
3	2	1	$\hat{U}(3) \bullet \hat{V}(1)$	$\hat{W}(3) = \hat{W}(3) + \hat{V}(1) \bullet -EL$ $\hat{W}(1) = \hat{W}(1) + \hat{V}(3) \bullet EL$	
3	2	2	$\hat{U}(3) \bullet \hat{V}(2)$	$\hat{W}(3) = \hat{W}(3) + \hat{V}(2) \bullet -EL$ $\hat{W}(2) = \hat{W}(2) + \hat{V}(3) \bullet EL$	1..3
4	3	1	$\hat{U}(4) \bullet \hat{V}(1)$	$\hat{W}(4) = \hat{W}(4) + \hat{V}(1) \bullet -EL$ $\hat{W}(1) = \hat{W}(1) + \hat{V}(4) \bullet EL$	
4	3	2	$\hat{U}(4) \bullet \hat{V}(2)$	$\hat{W}(4) = \hat{W}(4) + \hat{V}(2) \bullet -EL$ $\hat{W}(2) = \hat{W}(2) + \hat{V}(4) \bullet EL$	
4	3	3	$\hat{U}(4) \bullet \hat{V}(3)$	$\hat{W}(4) = \hat{W}(4) + \hat{V}(3) \bullet -EL$ $\hat{W}(3) = \hat{W}(3) + \hat{V}(4) \bullet EL$	1..4
etc.					

Note:- $\hat{U}$ & $\hat{V}$ represents a column vector of the respective array.

Table 9-8 Example of calculation in PREDIC Part II.

As is highlighted in the above example, there is a considerable amount of calculation effort contained in this program section. For any number *N* in the outer DO loop of the section, there is a corresponding *N-1* cycles of the internal DO loop which

modifies the values of all the vectors of W up to and including the *N* th vector of the interaction matrix, as determined in the outer DO loop.

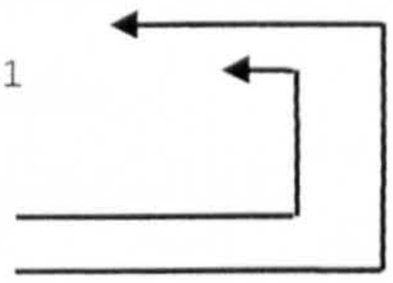
To determine logically the optimum method of parallelisation of these codes, it was found beneficial to develop a rational basis on which the work involved in completing the calculation could be quantified and subsequently divided among the network.

If we consider what is actually being achieved in this section of coding, the subroutine could be simplified to the following section of code, Program 9-3, namely an outer DO loop that has an inner DO loop.

```
C
C       PART II
C
      DO 4 I=LOCK,M
      DO 400 J=LOCK,I-1



  400 CONTINUE
    4 CONTINUE
```



Program 9-3

As was shown in Table 9-8, the amount of computation involved in each of the cycles of the inner DO loop is constant, determined only by the number of degrees of freedom in the model, N. Thus making this a reliable benchmark from which to calculate the computational effort required for the entire subroutine. Hence for future discussions, each of the inner DO loop cycles will be described as a Calculation Cycle.

Table 9-9 shows the number of cycles of the outer and inner DO loops required to calculate any general sized problem using this routine.

Outer Loop Cycle	Number of Inner loop Cycles	
LOCK	$[(\text{LOCK}-1)-\text{LOCK}+1]$	0
M	$[(\text{M}-1)-\text{LOCK}+1]$	$(\text{M}-\text{LOCK})$

Table 9-9 Number of DO loop cycles for a general sized problem.

The total number of Calculation Cycles is the sum of inner DO loop cycles from 0 to $(M-LOCK)$, which is given by:

$$\text{Calculation Cycles} = \sum_{i=0}^{(M-LOCK)} i$$

Equation 9-2

This is an arithmetic series, Equation 9-3, which can be expressed in two forms, Equation 9-3.

$$S_n = \frac{n}{2}(a+L) \qquad \text{or} \qquad S_n = \frac{n}{2}\left(2a+(n-1)d\right)$$

Equation 9-3

Where

S_n	Arithmetic sum of *n* terms.
a	First member of the series.
L	Last member of the series.
n	Number of members in series.
d	The arithmetic difference between members in the series.

Therefore, the total work or number of calculation cycles in the above routine can be expressed as shown in Equation 9-4.

$$S_n = \frac{((M-LOCK)+1)}{2}(M-LOCK)$$

Equation 9-4

To illustrate the implication of this expression for the total amount of work involved in any typical problem, an example will be considered. Table 9-10 below shows the distribution of Calculation Cycles for the (N=773, M=34) problem. Figure 9-19 shows the results in graphical form.

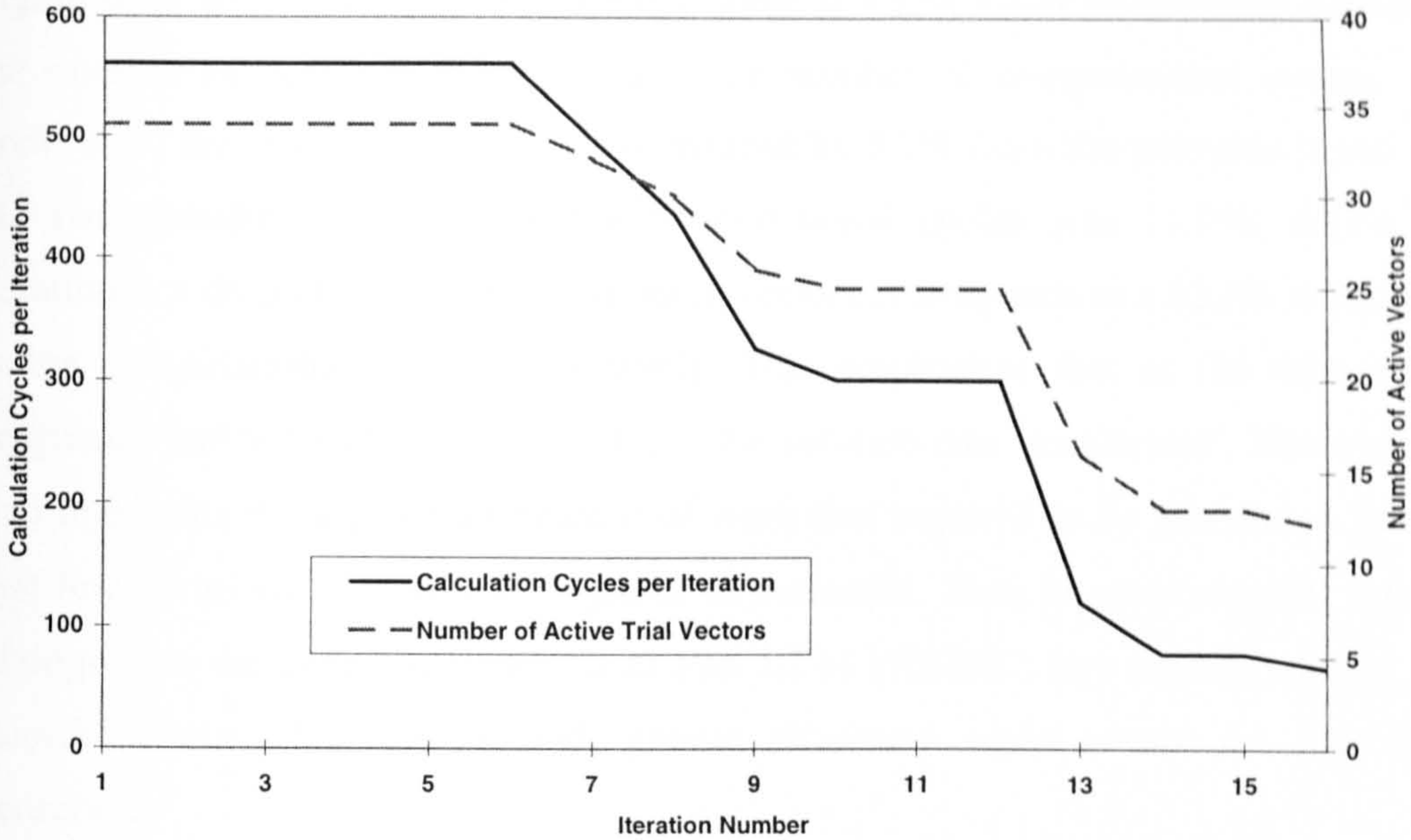


Figure 9-19 The calculation history for the (N=773, M=34) problem

Iteration Number	LOCK	Number of Active Vectors	Number of Calculation Cycles, S_n
1	1	34	561
2	1	34	561
3	1	34	561
4	1	34	561
5	1	34	561
6	1	34	561
7	3	32	496
8	5	30	435
9	9	26	325
10	10	25	300
11	10	25	300
12	10	25	300
13	19	16	120
14	22	13	78
15	22	13	78
16	23	12	66

Table 9-10 The calculation effort involved in the (N=773, M=34) problem.

Figure 9-19 and Table 9-10 indicate that there is a non-linear relationship between the number of active trial vectors and the number of computational cycles. At iteration 7, the number of vectors was reduced by 5.9% from the previous iteration, the corresponding reduction in the computational cycles was 11.9%. Again in iteration 8, a drop of 6.3% in the number of vectors corresponds to a 12.3% reduction in the computational cycles respectively. This emphasises that as the subroutine progresses and begins to solve for vectors, the solution rate 'accelerates'. However, it also highlights the significant amount of work that requires to be performed in the first few iterations until vectors begin to be extracted. Thus in some respects this is analogous to the situation presented in Part III of PREDIC; any strategy should be biased to solve the problem with greater efficiency while within the first few iterations.

The results of Figure 9-19, Table 9-8 & Table 9-10 also suggest that there are two rational ways of potentially reducing the total calculation time, by dividing the total work, either by,

i. An even distribution of Calculation Cycles.
ii. An even distribution of trial vectors.

However, any strategy for parallelisation of this subroutine would have to take account of the relative communication rates of the distributed computer system being used. In particular the relative speed of the processors in performing on-board calculation compared with the inter-processor communication rates, see Section 8.1.

9.3.2.2 Parallelisation of Serial Codes

The results presented in this section pertain exclusively to the parallelisation of the PREDIC Part II subroutine. The remaining subroutines of the eigensolution algorithm were run as serial codes.

The results presented relate to Table 9-11, which show the various different combinations of work distribution strategies and communication topologies implemented.

Work Distribution	Receive Network	Distribution Network	Program
Even	Farm	Farm	ParaNat7
Even	Farm	Finger	ParaNat20(8)
Polynomial	Finger	Finger	ParaNat21

Table 9-11 Distribution Methods and Communication Networks implemented for Part II, PREDIC.

As was discussed earlier, the main criterion for determining the optimum method by which parallelisation of the serial codes should be achieved relates to the strategies used to divide the total number of calculation per iteration. Should the division be along the lines of the total number of trial vectors being divided evenly among the network, or should the distribution be achieved by dividing the total calculation effort evenly among the network?. The following section discusses the two possible solutions in detail.

9.3.2.2.1 Even Vector Work Distribution

This distribution method as is suggested in the title is implemented to divide the total number of active trial vectors evenly amongst the processor network. The even work distribution of active trial vectors among the processor network is given by:

$$\text{Number of trial Vectors per Processor} = \frac{\text{Number of Active Trial Vectors}}{N_p}$$

Equation 9-5

where the number of active trial vectors is equal to (M - Lock) + 1.

The above expression (Equation 9-4) was programmed into a section of logic that was implemented in a generic parallel code thus allowing the program to internally optimise or "self-scale" itself to the total amount of system resources allocated to it.

The main section of the PREDIC Part II routine is computationally exceptionally demanding for the first few iteration cycles with the number of active vectors reducing in a non-linear manner with iteration number. Hence, as the algorithm progresses, the likelihood of obtaining an even distribution of trial vectors with no remainder becomes increasingly more unreasonable, prompting the question of what should be done with any remaining unallocated vectors.

The solution to this particular problem has already been addressed in Section 8.1.1 Consider what is actually being achieved when PREDIC is parallelised. Each processor in the network is given a unique distribution of the total amount of work, achieved by specifying the limits of the active trial vectors that will be operated upon by that processor, as shown below.

$$\text{DO for I} = \quad \text{Limit}_{\text{lower}} \quad \text{TO} \quad \text{Limit}_{\text{upper}}$$

The upper and lower limits are unique for each processor, such that the total calculation will be performed among the network. Once each processor completes its allocated task, it communicates its partial solution back to the Master processor, which sums all the partial solutions to obtain the total solution for that iteration.

In Table 9-8, the most distinguishing feature of PREDIC Part II is that as a processor operates on the vectors between $\text{Limit}_{\text{lower}}$ and $\text{Limit}_{\text{upper}}$, the processor has actually modified the partial solution from LOCK, which is the lower limit of the current number of converged vectors up to $\text{Limit}_{\text{upper}}$. This single feature was used to develop the subsequent logic of all the work distribution strategies. Since all the vectors between the limits of LOCK and M are to be distributed, it was logical to ensure that the greatest amount of communication should be done over the shortest distance possible, to keep the total duration of communication to a minimum.

In addition, to reduce the amount of communication further, the Master processor that sums up the partial solutions should be given a distribution of the work, that would if distributed to another processor significantly reduces the efficiency of the computational network. The work was distributed in such a way that the upper limit of the trial vectors was operated upon by the Master processor, with the lower limit being operated upon at the greatest distance from the Master processor, as shown in Figure 9-20.

Also, any remaining unallocated trial vectors, would be operated upon on the master processor to reduce the amount of communication that would be required to transmit the extra partial solution back through any particular communication network.

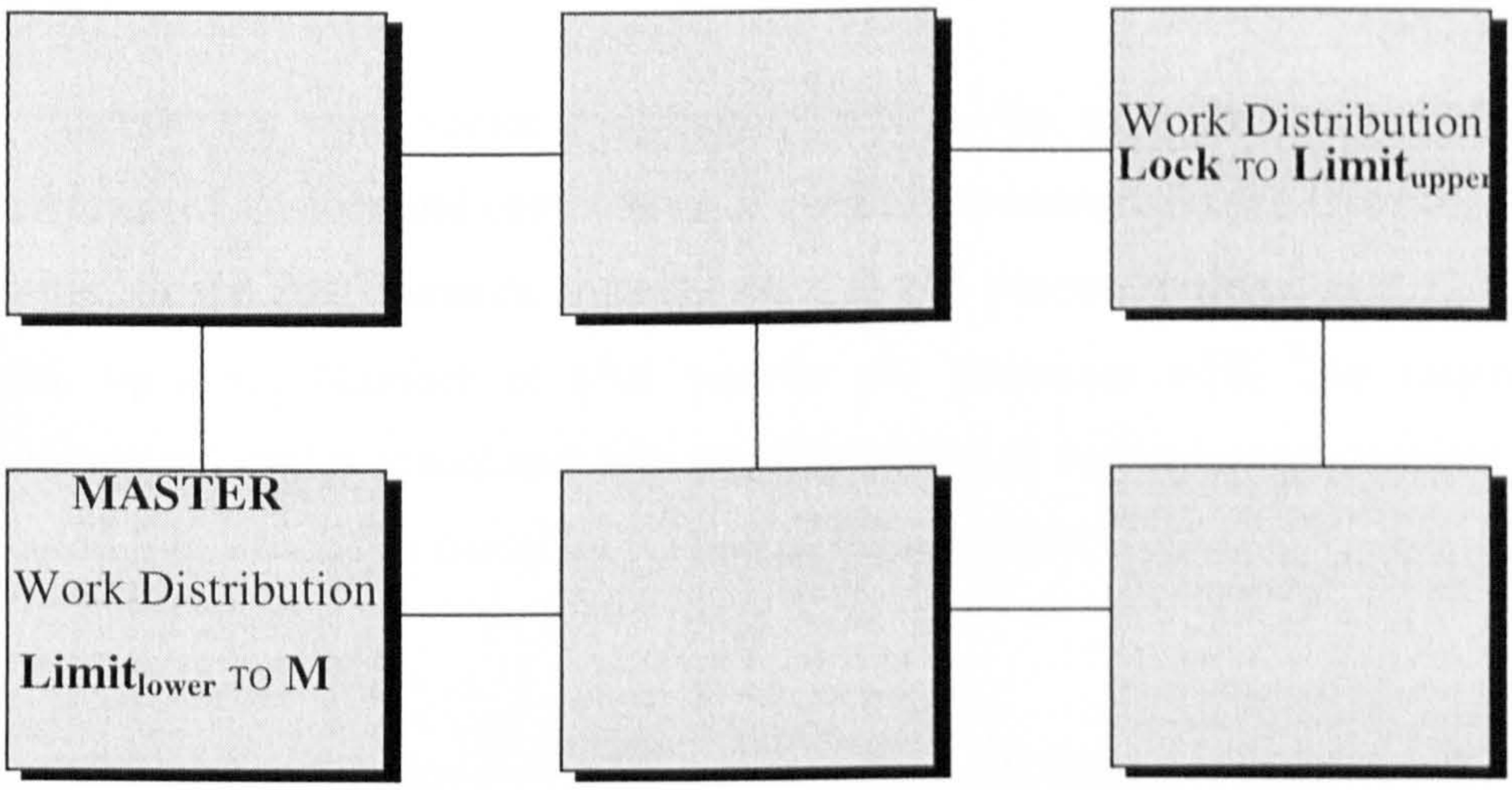


Figure 9-20 The allocation of work distributions to main processors

Applying these criteria, the final sections of the implemented logic in the generic algorithm was obtained and coded as shown below, Program 9-4.

```
CCC
C       Calculate Division of Work
CCC
      number=(m-lock)+1
      division=(number/nproc)
      work(1)=m
      work(1+nproc)=lock-1

      DO 100 i=1,nworker
      work(1+i)=((nproc-i)*division)+(lock-1)
  100 CONTINUE
```

Program 9-4

These codes were employed in the subroutine as shown in the Program 9-5 below, which is a section of the PREDIC Part I subroutine.

```
      DO 2 I= work(procid+2)+1,work(procid+1)
      EL=0.0D0
      DO 1 K=1,N
    1 EL=EL+U(K,I)*V(K,I)
      DBD(I)=EL
    2 CONTINUE
```

Program 9-5

To illustrate the even vector distribution method, the example below shows the distribution of vectors and computational cycles associated with the (N=116, M=76) analysis. In the first iteration, running on a 3 x 2 processor array. (i.e. LOCK=1, M=76, nproc=6, Number of trial vectors per processor =12) The number of computational cycles associated with the distribution of vectors was calculated using Equation 9-6, which is a modified version of Equation 9-3.

$$\text{Calculation Cycle} = \frac{\left(\left(\text{Limit}_{\text{upper}} - \text{Limit}_{\text{lower}}\right)+1\right)}{2}\left(\text{Limit}_{\text{upper}} + \text{Limit}_{\text{lower}} - 2\right)$$

Equation 9-6

Processor	$\text{Limit}_{\text{lower}}$	$\text{Limit}_{\text{upper}}$	Number of Vectors	Calculation Cycles
0	61	76	16	1080
1	49	60	12	642
2	37	48	12	498
3	25	36	12	354
4	13	24	12	210
5	1	12	12	66

Table 9-12 Example of Even vector distribution (N=116, M=76).

As is evident from Table 9-12, although the active trial vectors are evenly distributed among the network, the distribution of the computational cycles is exceptionally biased. The implications of this are discussed in detail in Section 9.3.2.2.3 where the problems associated with communicating the partial solutions to the master processor will also be addressed.

9.3.2.2.2 Polynomial Work Distribution Method

In the previous section, the computation associated with a constant division of active trial vectors can differ significantly depending on the upper limit of the work distribution allocated to any one processor. The next logical step in an attempt to resolve this significant bias in the work distribution is to divide the total number of potential computational cycles evenly amongst the processor network.

This was achieved by modifying the expression for the arithmetic sum, Equation 9-3, to calculate the limits for the vectors to be distributed resulting in an even distribution of total work. Again the upper limit of the trial vectors was allocated to the master processor, following the logic developed for the even vector distribution method. Any remainder was allocated to the Master processor.

The sum of n terms; S_n; of the total work in the PREDIC routine was calculated using the following expression,

$$S_n = \frac{n}{2}(2a+(n-1)d-2)$$

Equation 9-7

This was re-arranged into the form required to calculate the number of trial vectors that should be associated with each processor to give the required division of total work. The work distribution was calculated from the M th vector backwards, thus the arithmetic difference between consecutive vectors, d was equal to -1. The re-arranged expression (Equation 9-8) is expressed in terms of active trial vectors and S_{nD}, where S_{nD} is an even distribution of the total computational effort and is calculated by dividing Equation 9-4 by the number of processors N_p.

$$n^2 - n\cdot(2a-1)+2\cdot S_{nD} = 0$$

Equation 9-8

Solving this for the number of active vectors n gives,

$$n = \left(a - \frac{1}{2}\right) - \frac{1}{2}\sqrt{\left[(2a-1)^2 - 8 \cdot S_{mD}\right]}$$

Equation 9-9

This expression allows the limits of the allocated trial vectors to be calculated in a sequential manner, building on one processor's calculated limit to determine the next processor's limit in sequence, Equation 9-10.

$$Work(I+1)=Work(I)-n$$

Equation 9-10

The generic algorithm coding was as follows, Program 9-6

```
CCC     Calculate Division of Work
        number=(m-lock)+1
        divi=(number/nproc)
C
        work(1)=m
        work(1+nproc)=lock-1
C
        cycles=(number/2)*(number-1)
        division=real(cycles/nproc)
C
        DO 215 I=1,nworkers
        share= ((work(i)-0.5)-(0.5*(sqrt((((2*work(i))-1)**2)
       * -(8*divis)))))
        work(i+1)=work(i)-nint(share)
  215 continue
CCC
```

Program 9-6

The following examples show the results of the polynomial distribution method for the (N=116, M=76) problem. Table 9-13 shows the result for a 3 x 2 network, Table 9-14 for the same problem running on a 3 x 4 network. Both Tables give the results for the first iteration cycle; Lock=1.

Procid	$Limit_{lower}$	$Limit_{upper}$	Number of Vectors	Calculation Cycles
0	70	76	7	504
1	63	69	7	455
2	55	62	8	460
3	45	54	10	485
4	32	44	13	481
5	1	31	31	465

Table 9-13 (N=116, M=76) running on a 3 x 2 array.

Procid	$Limit_{lower}$	$Limit_{upper}$	Number of Vectors	Calculation Cycles
0	74	76	3	222
1	71	73	3	213
2	67	70	4	270
3	63	66	4	254
4	59	62	4	238
5	55	58	4	222
6	50	54	5	255
7	45	49	5	230
8	39	44	6	243
9	32	38	7	238
10	23	31	9	234
11	1	22	22	231

Table 9-14 (N=116, M=76) running on a 3 x 4 array.

As is clear from the above Tables, the distribution of the work amongst the processors is much more even while the number of active trial vectors allocated to each processor is relatively non-uniformly distributed. This also has implications as to how the partial solutions should optimally be recovered and distributed around the network. The problems associated with this will be discussed in the following section.

9.3.2.2.3 Communication Network Topology

To aid the interpretation of the choice of communication network, a few criteria must be satisfied. The criteria associated with PREDIC part II are as follows.

1. The network must be able to collect and sum partial solutions from all processors in the network while minimising the communication time and avoiding bottlenecks.
2. The network must distribute the total solution to all processors in the network as quickly as possible to allow the network to progress to the next subroutine in the eigensolution algorithm.

The main communication demands of this routine are to collect partial solutions and distribute the total solution. The effects of each communication method are first discussed separately before cumulative effects are considered.

As was shown in Table 9-11, two communication topologies were employed in three combinations. Table 9-15 shows the Performance Indices (PI) for the various combinations of work distribution strategies and communication topology, for the (N=116, M=76) problem running on a number of different processor networks.

Processor Array	N_p	Even-Farm-Farm	Even-Farm-Finger	Polynomial-Finger-Finger
		ParaNat7	ParaNat20(8)	ParaNat21
SERIAL	1	1.00	1.00	1.00
2 x 2	4	1.98	2.07	3.64
3 x 2	6	2.03	2.27	4.63
3 x 3	9	2.08	2.60	5.56
4 x 3	12	1.95	2.70	5.77
4 x 4	16	1.37	1.90	6.00
4 x 8	32	---	---	4.05

Table 9-15 Performance Index for the various combinations of parallelisation strategies and communication topologies (N=116, M=76)

Table 9-15 gives an indication of the effect the distribution communication has upon the calculation time. In particular, if the results of ParaNat20 and ParaNat7 are compared, the only difference in the implemented algorithm is the distribution network being changed from the Farm to Finger network, respectively. The results indicate that there is a 5 - 39 % increase in the PI associated with this single change across the various sizes of networks. This is due to the existence of a large bottleneck in the Farm communication network.

The way the Farm network operates is to communicate the entire solution to each processor in sequence. Consequently, as the master processor communicates to processor 1, the remaining processors in the network from 2 to N_p are idle. Thus it can be said that the network is obtaining its solutions from a single source. The

number of idle processors is reduced by one for each communication cycle. The maximum delay experienced by any individual processor can be quantified as ((DimX x DimY)-1) communication cycles.

The Finger network is a significant improvement on the Farm network, although it also has a small bottleneck problem associated with it. The Finger network (unlike the Farm network) has a totally adaptive communication structure allowing the maximum delay experienced by any single processor to be minimised.

Initially the Finger network communicates from the master to processor 1, in a similar way to the Farm network, the rest of the processors in the network being idle. However, in the second cycle, the Master processor communicates to processor DimX, while processor 1 communicates to processor 2. This methodology is repeated cycle after cycle with communication from multiple sources, the number of idle processors reducing by a maximum of DimX with each communication cycle. The maximum possible delay experienced by a processor is (DimX + DimY -2) communication cycles, with the corresponding maximum number of sources simultaneously communicating being DimX.

Another significant difference between the Finger and Farm communication topologies is that in the Finger scheme each processor communicates only with their adjacent neighbour, allowing a significant reduction in the time required for each communication cycle, (see Section 8.1.1).

Another major difference is the collection of the results; there are two variables that control the collection of the partial solution. They are, firstly the work distribution being employed and secondly the communication network used to recover the results.

A separate study was made to examine the effects of the communication network upon the calculation time of the algorithm. The Finger network yielded a 21% - 52% increases in the Performance Index, for the (N=116, M=76) problem, when compared with the Farm network. This significant increase was due to the "displacement" of the potential bottlenecks within in the network.

This "displacement" of the potential bottlenecks within the network, can be easily explained. As noted in Section 8.2.2, the Finger network allows the collection and summing up of partial solutions to occur from multiple sources concurrently. The Farm network however only allows the communication of processor's partial solutions to the Master processor, while the rest of the network waits to communicate, thus causing global delays in the network.

Thus, the Finger network's ability to adapt and communicate concurrently allows the "displacement" of the communication bottlenecks within the network. Since the Master only receives results from two processors, the associated bottleneck is reduced to a negligible level due to the careful scheduling of the communication.

Finally, the effect the collection communication topology has on the solution will be discussed; however this will be done including the effect of the different work distribution strategies. The comparison will be made using the solutions for ParaNat20 which uses the Even-Farm combination and ParaNat21 which uses the Polynomial-Finger combination of work distribution and communication network respectively, (refer to Table 9-11).

This allows the comparison of a method that permits concurrent communication and summation of partial solutions, with one that only allows a single communication and summation to take place in any one cycle. The increase in the Performance Index for this combination of changes from ParaNat20 to ParaNat21 ranges from 75–216%, which represents a significant gain over solely changing the communication network used to collect the results. This increase is due in part to the role of the master processor (procid 0). In the Even vector distribution method, the Master processor has to complete significantly more calculation cycles that any other processor. This means that the rest of the processors in the network will have completed their allocated calculations before the master processor. Thus, they will be idle waiting to transmit their partial solutions and receive the total solution from the master processor.

Conversely the Polynomial distribution method leads to a more even distribution of calculation cycles, thus the processors in the network communicate their partial

solutions much more quickly allowing the master to communicate the total solution in a much reduced time.

The results of the various versions of the PREDIC Part II subroutine are shown in graphical form in Figure 9-21. The results are also shown in term of the network efficiency, Figure 9-22. This was obtained by dividing the Performance Index by the number of processors in the network, N_p.

Figure 9-21 shows initially that as the number of processors included in the network increases, the PI increases. This however peaks and then begins to rapidly drop. This significant drop in PI is due to the individual computational tasks being sub-divided to an extent that the communication bottlenecks within the system exceed the potential speed-ups within the network. Figure 9-22, emphasises this point by showing that as the number of processors in a network increases, the average utilisation of each individual processor with in the network drops considerably. Thus suggesting that it does not make good economic sense to buy large order computers to solve medium sized problems.

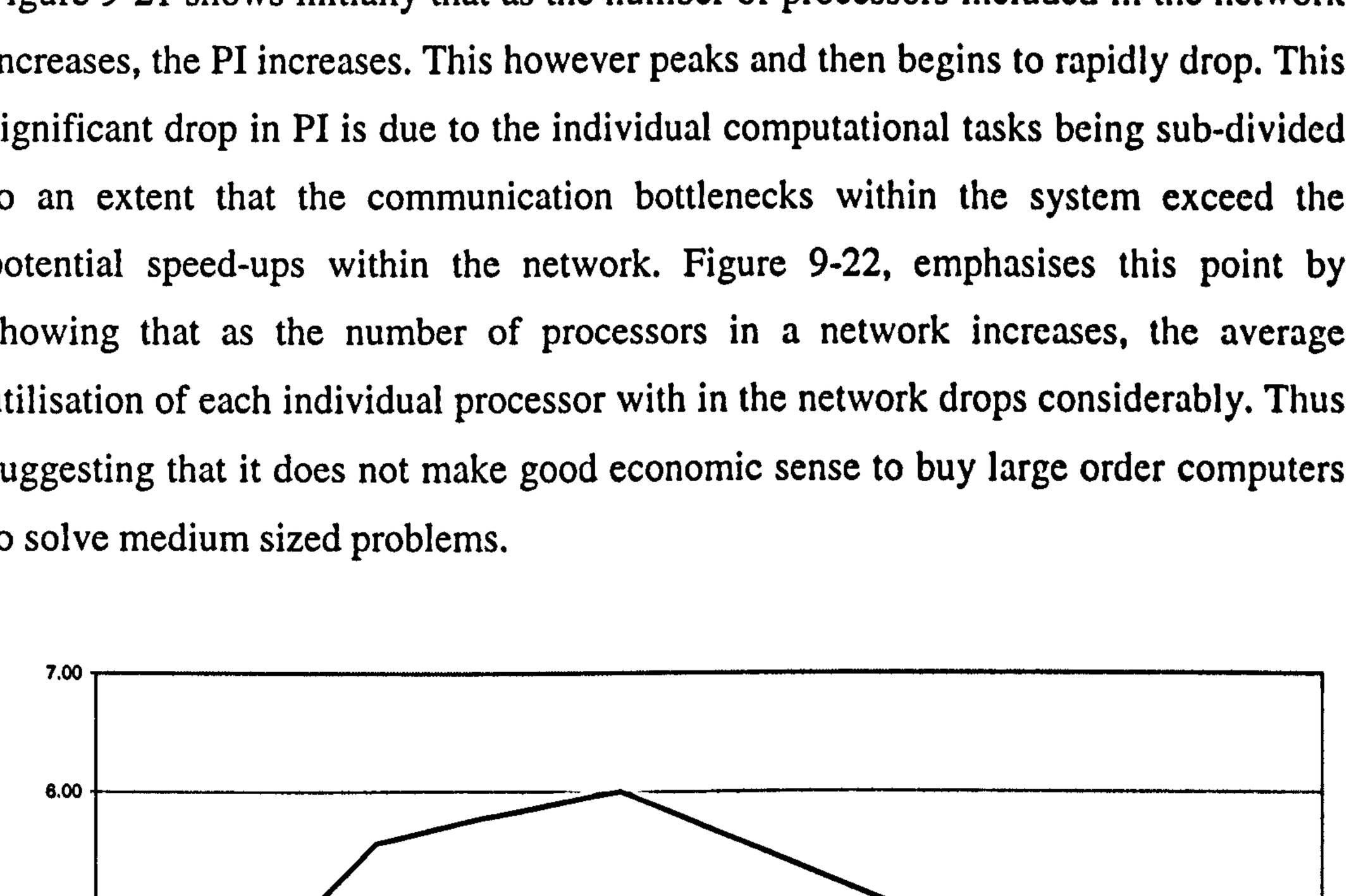


Figure 9-21 The Performance Index of the Parallelised PREDIC Part II routine.

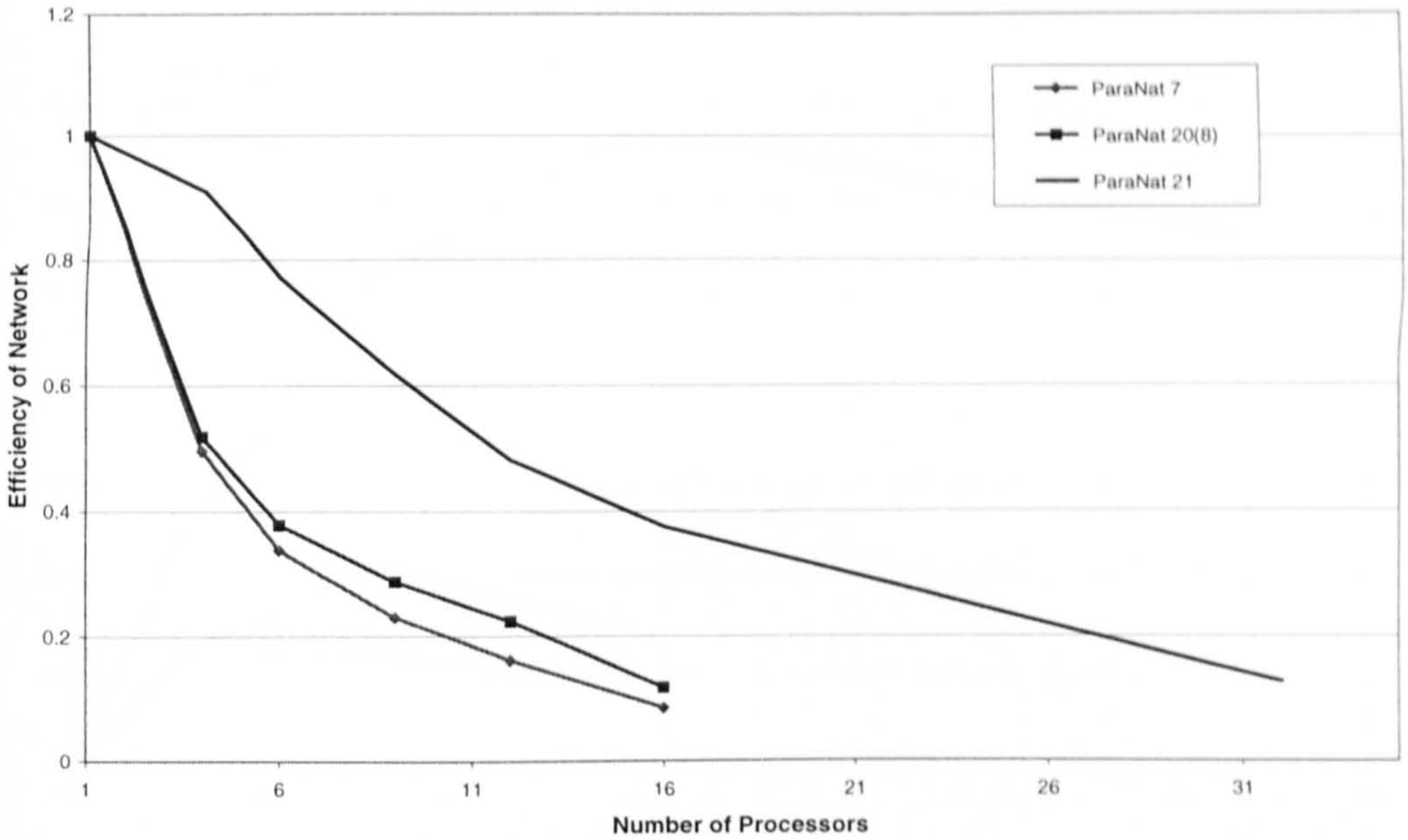


Figure 9-22 The relative efficiency of the parallelised PREDIC Part II routine.

9.3.2.2.4 Parallelisation of PREDIC Part I

This component of the PREDIC routine was insignificant compared to Part II. However due to the simplicity of the calculation it performed, the code was modified as indicated in Table 9-16.

Work Distribution	Receive Network	Distribution Network	Program
Even	Farm	Farm	ParaNat 5
Even	Farm	Finger	ParaNat 6
Polynomial	Farm	Finger	ParaNat 21

Table 9-16 Distribution Methods and Communication topologies implemented in PREDIC Part 1.

The results of these various modifications are shown in Figure 9-23. The logical arguments for the modifications follow on from the discussion of Part II above.

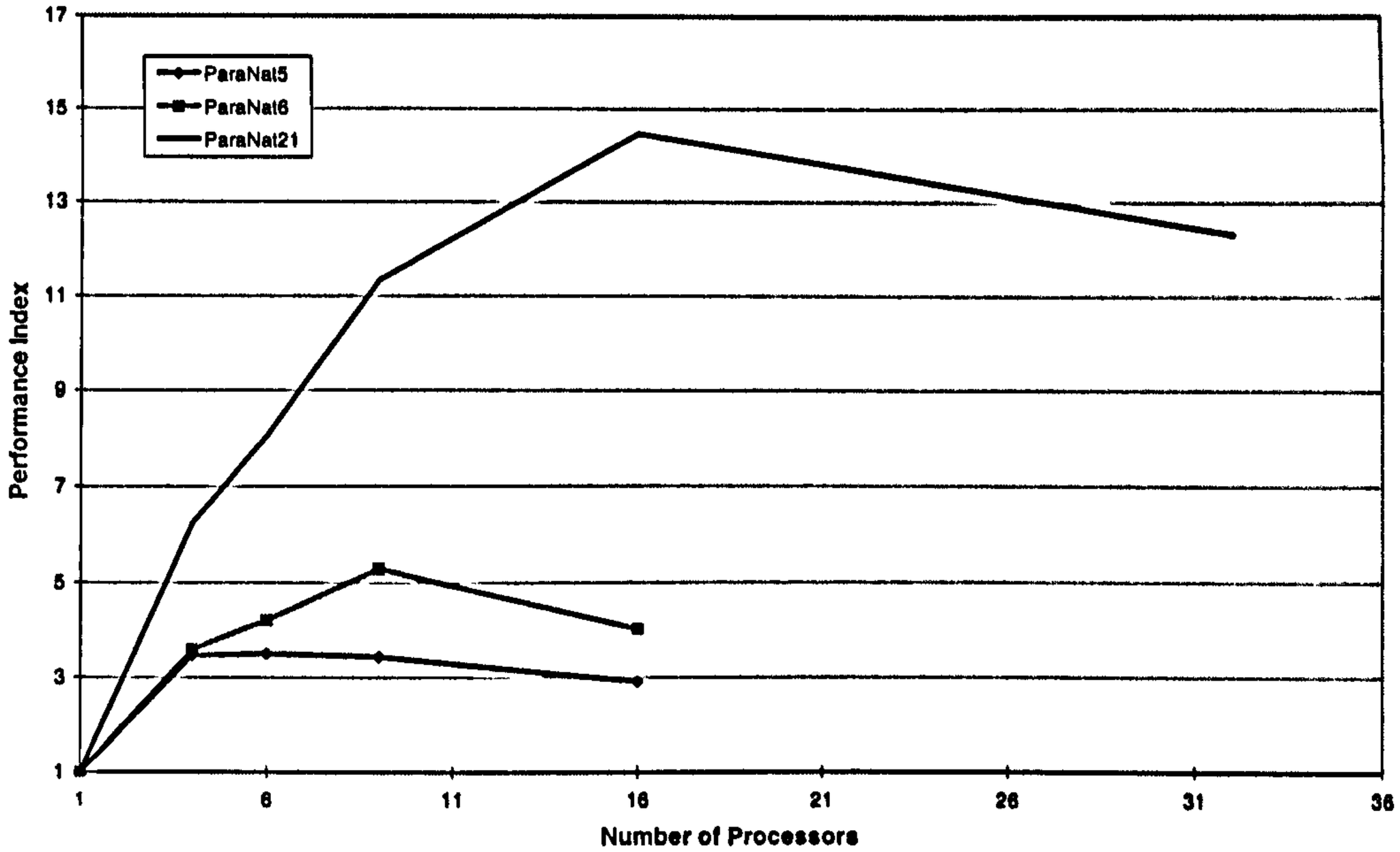


Figure 9-23 The Performance Index of the Parallelised PREDIC Part I routine.

9.3.2.2.5 Parallelisation of REDIC Part III

This section of the PREDIC routine could not be easily parallelised due to the nature of the calculation it performs. In sorting the sequence of the vectors into descending order, the routine requires that all the results be known on any one processor to allow the sorting to take place. Thus, it was decided to allow this routine to run in a concurrent manner on each processor with no inter-processor communication. Secondly, the serial version of the routine was examined and re-written to be more efficient than before.

The original serial codes sorted the eigenvalues in a consecutive manner comparing one eigenvalue with its neighbours, if a larger eigenvalues was found their positions within the solution vectors would be switched along with their associated eigenvectors. The improvement made was to only switch the eigenvalues, keeping a vector containing information about the final locations of each eigenvalues, once they were finally sorted into descending order, the associated eigenvectors were moved only once, to their final position within the solution vector. The resulting codes being 34% faster than the original.

9.3.3 ORTHOG - The Vector Orthogonalisation Routine

This routine accounts for approximately 30% of the entire calculation effort required in the iterative section of the SI algorithm, Figure 9-16. This routine, unlike the PREDIC subroutine, had no clear structures that could be sub-divided although, the routine did have a clearly defined logical structure and was essentially an iterative process. The iterative component of the subroutine naturally lent itself to the parallelisation process.

ORTHOG required an exceptionally rigid communication and distribution strategy, due to the computational demand that had to be catered for as a result of the way the calculation was performed. To calculate an ortho-normalised trial vector, all the data from the previous trial vectors has to be available on the processor performing the ortho-normalisation.

To understand better how the routines could be parallelised, the serial coding was examined to determine any clearly defined program structures that could be utilised in the parallelisation of the coding, Program 9-7.

```
      SUBROUTINE ORTHOG(V,N,M,LOCK)
            EXTERNAL VECSUM
      C
            DO 1 I=LOCK,M
      C
            DO 3 J=1,I
            EL=0.0D0
      C
            DO4 K=1,N
          4 EL=EL+V(K,J)*V(K,I)
      C
            IF(I-J)3,5,3
          3 CALL VECSUM(V,V,I,J,-EL,N,M)
      C
          5 D=1.0D0/DSQRT(EL)
            DO 1 K=1,N
          1 V(K,I)=D*V(K,I)
            RETURN
            END
```

Program 9-7

Where

N	The number of active degrees of freedom.
V(N,M)	Trial vectors to be ortho-normalised.
M	The number of iteration trial vectors being used.

LOCK Lower limit of active trial vectors (i.e. inactive vectors no longer included in some operations).

Basically the ortho-normalisation is performed in two parts. Firstly the trial vector V(i) is orthogonalised using vectors V(j),(where j= 1 to i-1). Secondly the trial vectors is normalised with respect to itself.

The two standard expressions that are used to ortho-normalise a set of vectors are

$$V(i)^T \cdot V(j) = 0 \qquad i \neq j$$
$$V(i)^T \cdot V(j) = 1 \qquad i = j$$

Equation 9-11

It is evident from the code fragment above, that to correctly orthogonalise a vector V(i) the solutions for all the vectors up to V(i-1) must be available locally to the processor orthogonalising the current vector.

The implication of having to communicate all the individual orthogonalised vectors to the rest of the network would place an overwhelming demand upon the communication network. It became evident that having to possess the solutions of all the previously calculated vectors on all processors that require the solution would be very demanding. An alternative strategy was obtained, that would still allow the majority of calculation to be performed on-board local processors, while minimising the communication being performed. The solution adopted was termed the Cyclic-Distribution Method because of the manner in which the problem was solved.

Each processor in the network is allocated a single vector to orthogonalise or normalise depending on its processor number. Once each processor has performed its allocated calculation it communicates its partial solution to the Master processor which stores all the newly calculated vectors from all the Slave processors. When all the newly calculated vectors are recovered from the network, the Master processor then communicates the last N_p vectors to all the slave processors. The Slaves now locally contain the complete solution for *Computational Cycles* x N_p vectors,

allowing more calculation to be performed at a local level for the next computational cycle.

The method is illustrated by a worked example of the method, again for consistency the (N=116, M=76) problem will be used to develop the example, running on a 3 x 2 processor network, N_p=6, M=76, Lock=1.

The method developed has several distinguishing features contained within its coding, and like the parallelised PREDIC subroutine is fully generic or "self-scaling" to the system resources it is allocated.

The problem is solved in an iterative manner as follows from I= Lock to M.

Determine which computational cycle, *mult,* is currently being solved, Equation 9-12. The subroutine then allocates the next N_p vectors to be orthogonalised to the processor network, distributing them in accordance with an algebraic expression dependent on their individual processor numbers, procid, Equation 9-13.

$$mult = \frac{I}{N_p}$$

Equation 9-12

$$\text{procid} = (\mathbf{mod}(I - 1), \text{nproc})$$

Equation 9-13

The exception to this occurs when *mult* =0, in which case a special piece of logic is invoked, which instructs the Master processor only to ortho-normalise the first N_p vectors.

The newly orthogonalised N_p vectors are then distributed around the entire network at the end of each computational cycle, for use in all subsequent cycles. It is worth noting that at the beginning of a new computational cycle, all the processors in the network have the orthogonalised vectors from 1 to ((*mult*-1) x N_p).

The subroutine then calculates the upper limit of the previous computational cycle, *limit*. This upper limit is used by the Slave processors around the network to orthogonalise their currently allocated vector, using the previously orthogonalised vectors from 1 to *limit*. These locally calculated solutions are only partially orthogonalised and are denoted as $V(i)^*$, and are communicated directly to the Master processor using the Farm topology, which completes the ortho-normalisation of the vector, from *limit*+1 to i, refer to Table 9-17.

Vector	Allocated to processor p=(**mod**(I-1), nproc) †	Vectors operated upon Locally	Vectors operated upon on Master processor
1	0	1..1	-
2	0	1..2	-
3	0	1..3	-
4	0	1..4	-
5	0	1..5	-
6	0	1..6	-
7	0	1..7	-
8	1	1..6	7..8
9	2	1..6	7..9
10	3	1..6	7..10
11	4	1..6	7..11
12	5	1..6	7..12
13	0	1..13	-
14	1	1..12	13..14
15	2	1..12	13..15
16	3	1..12	13..16
17	4	1..12	13..17
18	5	1..12	13..18
19	etc.		

† This expression is disregarded if *mult* < 1

Table 9-17 The calculation strategy of the ORTHOG subroutine

The Master processor has a special role to perform in the ortho-normalisation procedure. This processor is allocated the vector ((*mult* x N_p)+1) at the beginning of each computational cycle, which is completely ortho-normalised and stored on the Master along with the previously calculated vectors from previous computational cycles. Thus once the Master processor has calculated vector ((*mult* x N_p)+1), it is the only processor in the network that has the entire set of ortho-normalisation vectors from 1 to ((*mult* x N_p)+1). Hence the other processors in the network can only orthogonalise their allocated vector up to vector ((*mult*-1) x N_p) or *limit*.

Once the locally calculated partially orthogonalised solution $V(i)^{\bullet}$ is obtained on the allocated processor, this partial solution is communicated to the Master processor that completes the ortho-normalisation of the vector using the newly stored ortho-normalised vectors. Hence, the Master calculates from *limit*+1 to I, for all the (N_p-1) vectors that are partially calculated on the Slave processors.

At the end of each computational cycle, the Master processor uses the Finger communication network to distribute the last N_p ortho-normalised vectors to the network of Slave processors.

There are two important points to be noted for the Cyclic Distribution Method for the development of the solution.

Firstly, the method used for the distribution of the N_p solutions was the Finger communication network, which was shown in PREDIC to have significant advantages over the Farm communication networks, mainly due to its ability to 'displace' data bottlenecks.

Secondly, as the subroutine progresses through several computational cycles in the particular example, the amount of locally stored data increases. After the first computational cycle, each Slave processor contained 6 (N_p) ortho-normalised vectors, which are used to partially orthogonalise the next allocated vector, in subsequent cycles each processor would locally store 12, 18, 24 vectors, etc. This means that for this increase in the amount of locally available data, a large proportion

of the calculation effort required to ortho-normalise a vector can be performed locally, freeing the Master processor to serve the rest of the network when required.

Table 9-18 below shows the performance indices for two different implementations of the ORTHOG subroutine. Both use the Farm topology to receive data and the Finger topology to distribute data; the only difference between the two methods is that in ORTHOG2 the Master processor is allocated the last vector in each computational cycle (i.e. 6, 12, 18 etc.), while in ORTHOG3 the Master processor is allocated the first vector in each computational cycle (i.e. 1, 7, 13 etc.)

Processor Array	nproc	Orthog 2	Orthog 3
SERIAL	1	1.00	1.00
2 x 2	4	1.93	3.25
3 x 2	6	2.46	3.61
3 x 3	9	2.84	3.74
4 x 3	12	2.86	3.48
4 x 4	16	2.95	3.30
4 x 8	32	2.19	2.27

Table 9-18 Speed-up factors for various combinations of parallelisation methods

By examining Table 9-18 and Figure 9-24, it is quite clear that the different methods used for the allocation of vectors have a considerable effect on the speed of the subroutine. The associated network efficiencies are given in Figure 9-25.

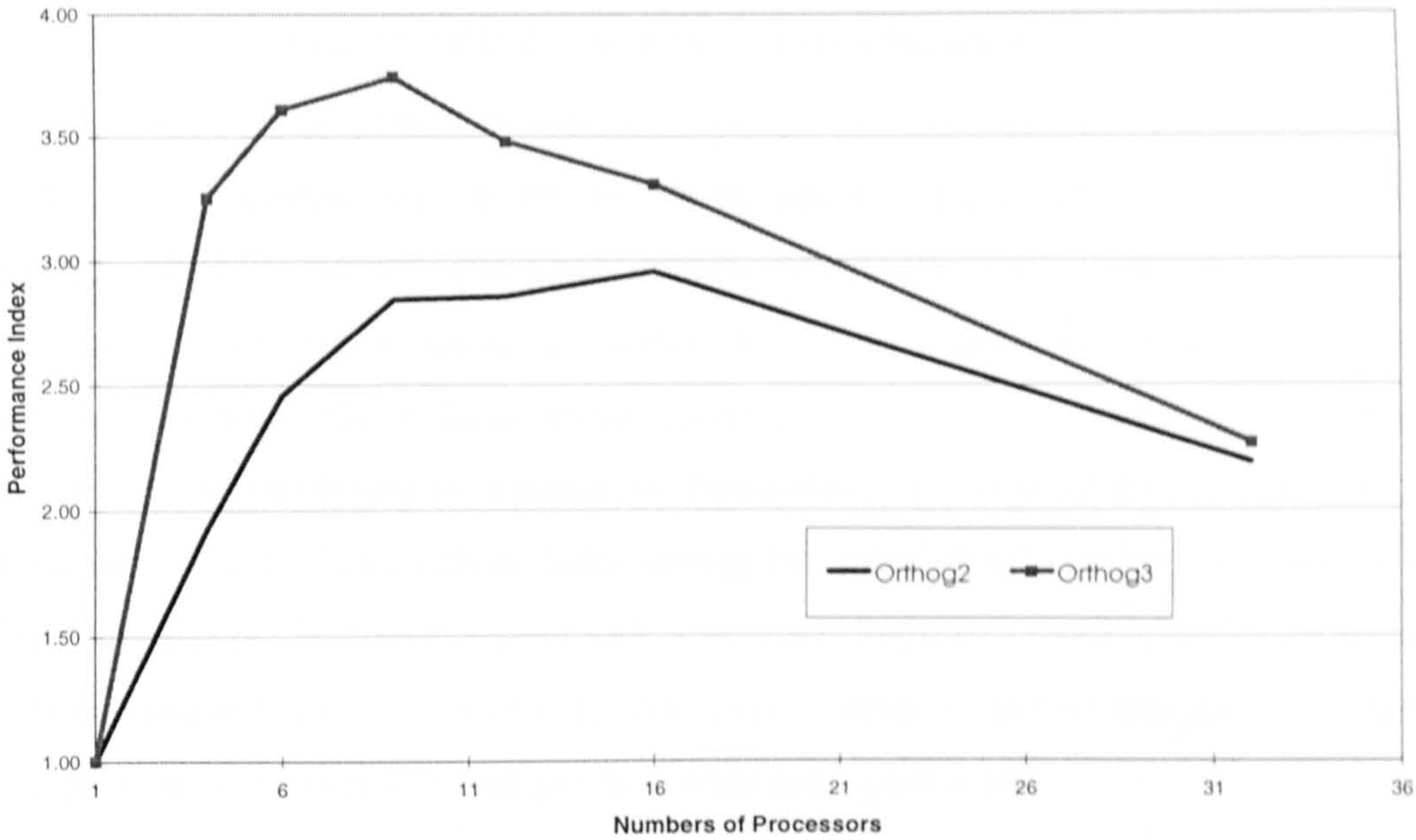


Figure 9-24 The Performance Index for the parallelised ORTHOG routines.

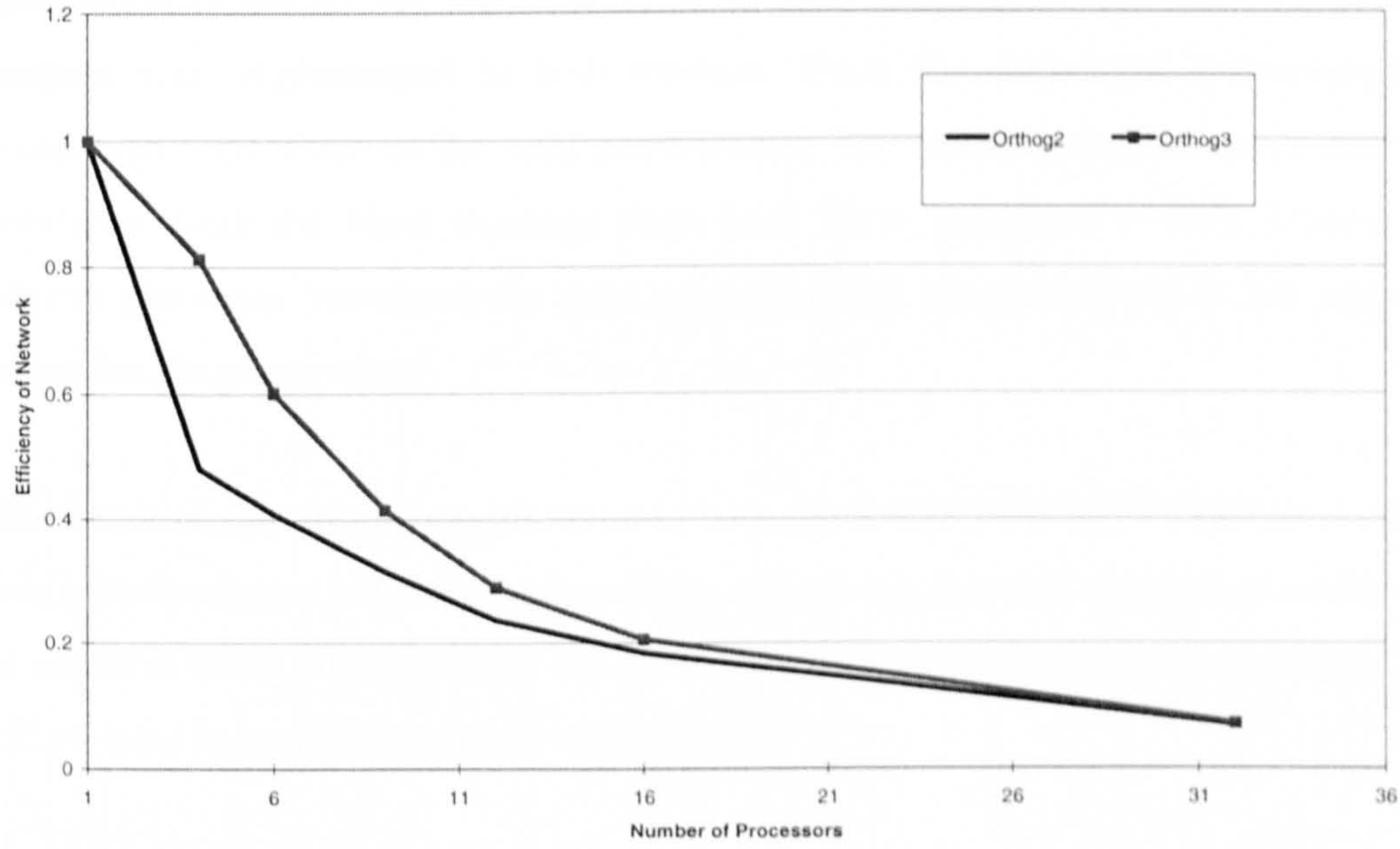


Figure 9-25 The Relative Efficiencies of the parallelised ORTHOG routines compared with the Serial Version.

9.3.4 BACKSUB & FORSUB - Auxiliary Matrix Routine

The BACKSUB and FORSUB routines typically account for approximately 20% and 12% of the iterative section of the SI algorithm respectively, Figure 9-16. The function of these routines is to perform back-substitution and forward-substitution of the current trial eigenvectors, to effect the pre-multiplication process (Refer to Jennings[63..65]). Since both these routines consist of a series of operations performed independently on a particular eigenvector, it was possible to simply divide the total amount of calculation tasks among the network in a manner similar to the PREDIC routine. Unlike the previously discussed PREDIC & ORTHOG subroutines both the routines performed exactly the same number of operations per eigenvector irrespective of the eigenvectors position within the global set.

Similar to the PREDIC routine, the simplest way to divide the total amount of calculation effort among the network is an even division of the total work. This method was implemented in both routines. Once the individual processors had completed their share of the total computation, the Master processor recovered the solutions using the Farm topology from each Slave processor in turn. Finally the Master processor broadcast the total solution to all the processors in the network using the Finger topology.

The result of this simple modification to both the BACKSUB and FORSUB routines was a Performance Index for both routines of between 2 and 3, depending on the size of problem being attempted and the size of the processor network, thus justifying the effort spent in parallelising these subroutines.

9.3.5 Conclusions for Parallelisation of Eigenvalue Analysis

There are several general conclusions that can be draw from the work in parallelising the Simultaneous Iteration eigenvalue algorithm. Firstly, it has been shown that by studying the specific structure of various algorithms, considerable reductions in computer analysis time can be achieved.

Secondly the development and implementation of efficient communication topologies that utilise and generically 'self scale' themselves can on their own introduce a considerable improvement in the computational efficiency of the algorithm. However, if combined with a relatively efficient method of distributing the tasks around the processor network, the potential gains can be considerably larger.

Thirdly, as has been demonstrated in the parallelisation of the BACKSUB and FORSUB subroutines, which perform simple independent calculation using a clearly defined algorithm structure, there can be a considerably improvement by even the most basic parallelisation of the serial codes.

Finally, and most significantly, all the above parallelised routines exhibit a speed-up of scale. This means that as the problem to be solved becomes larger, the potential improvements in the network performance increase. This increase is non-linear and for several of the routines the limit of the improvement has as yet not been reached, even when solving a model with a large number of degrees of freedom.

The main objective of parallelising the various subroutines within the simultaneous iteration algorithm was to attempt to minimise the time required for each iteration through the algorithm and fundamentally, to reduce the total time required in solving a specific problem. Figure 9-27 & Figure 9-26 show the results for the explicitly parallelised subroutines and all the subroutines within the algorithm respectively. Note, that in Figure 9-26 the performance indices for the three non-parallelised subroutines are greater than unity, this is due to the optimisation option being used when compiling the serial codes. By examining Figure 9-27, it is possible to determine the effect the parallelisation had upon the total algorithm. The

performance index for the total cycle stays above 3 for the majority of analysis, and never drops below 2. Hence, it would be reasonable to say that for this particular problem, the total solution was obtained in less than half the time, or the parallelisation resulted in an improvement greater that 100%.

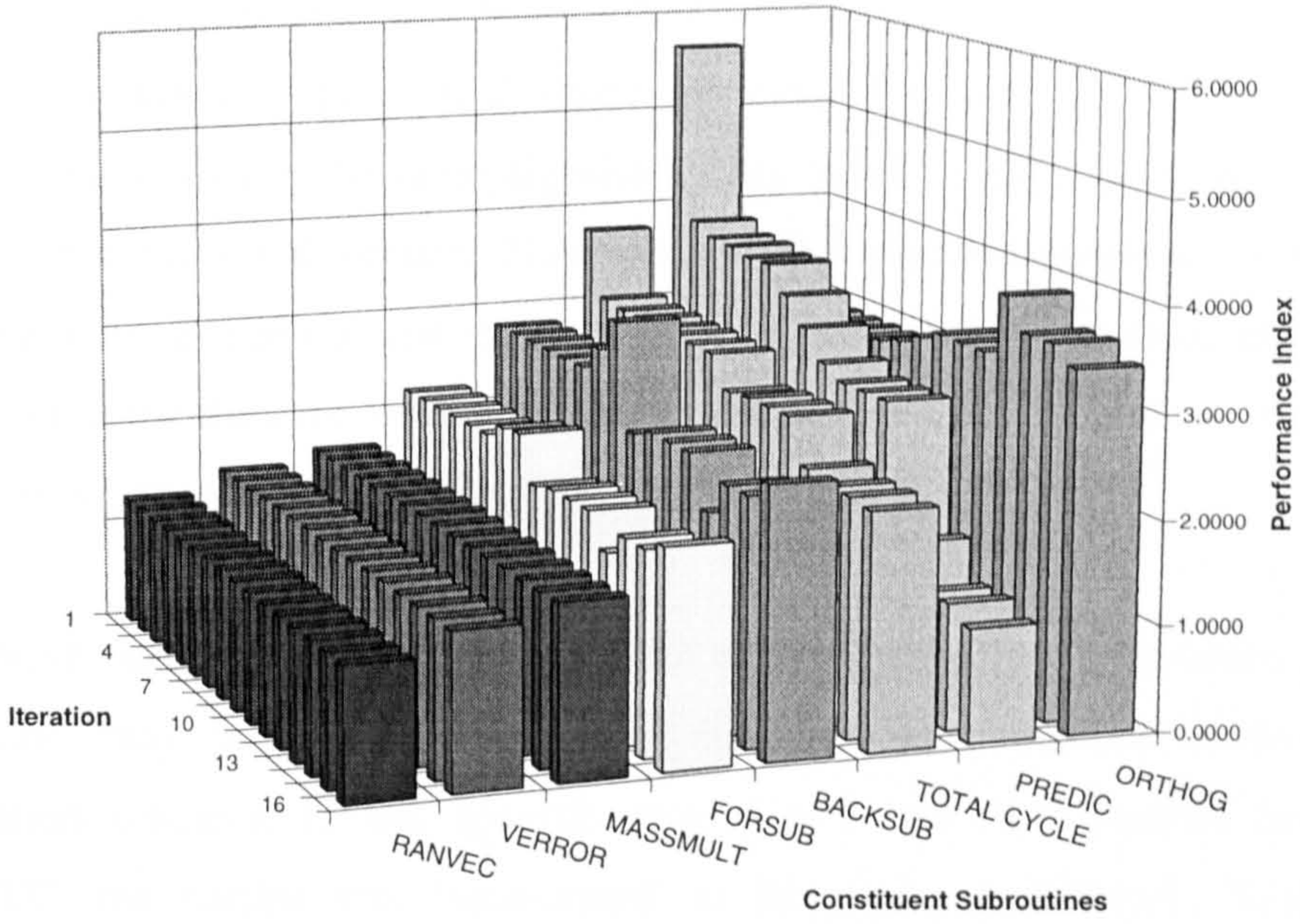


Figure 9-26 The Performance Indices for the Parallelised Simultaneous Iteration Algorithm (N=116, M=76) on a 3 x 3 processor network.

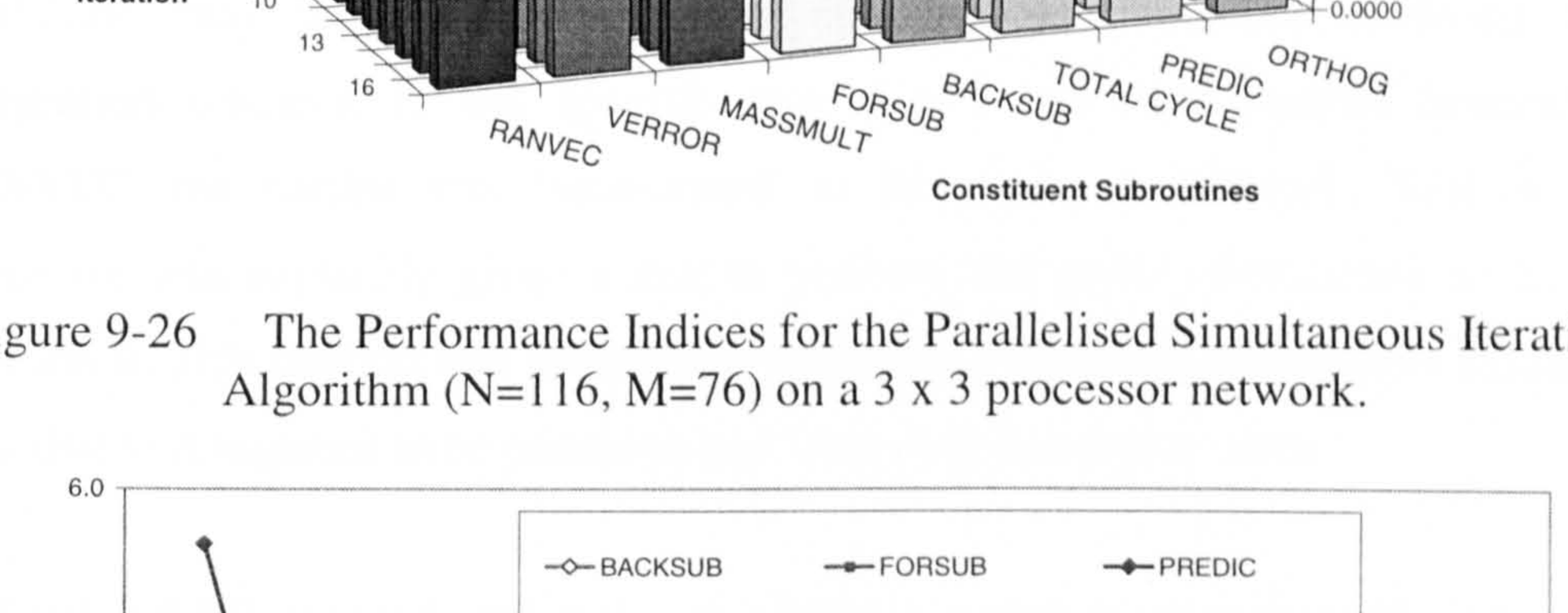


Figure 9-27 The Performance Index for the Parallelised Subroutines within the Simultaneous Iteration Algorithm (N=116, M=76) on a 3 x 3 processor network.

9.4 General Conclusions on Parallelisation

The results obtained for the parallelisation of both the time history algorithm and eigenvalue algorithm indicate that there is potential for considerable reduction in the time required in performing a specific analysis.

The results for the time history algorithm showed an improvement of around 28 % in the time required to perform a single iteration. This would suggest that if the parallelised codes ran the same algorithm, they would complete the analysis 28 % quicker than the serial version. However, if the Newmark integration scheme was replaced with a method that required less iterations per time step, the potential reduction in the duration of an analysis would have a lower bound of 28 % and an upper bound possibly significantly larger.

The above conclusion, while valid for the specific algorithms implemented within ANSUSP, may not be generally applicable to other programs involving other integration schemes. In this specific case, due to the clear parallel structure of RESVEC, the routine was 'hard-coded' to the processor network. That is each processor was explicitly given a task to perform and given instructions on how to perform it. This method was found to be necessary due to the considerable amount of data that was required to be passed to and from each Slave processor.

In Section 8.1.2, it was found that each processor could calculate considerably faster that the processor network could communicate the same amount of data. The implication of this being that the algorithm structure that would show the greatest improvement, is the one that requires to communicate a small amount of data to perform a long calculation. This is the antithesis of the structure of the time history algorithm parallelised within ANSUSP.

The Simultaneous Iteration algorithm on the other hand has exactly the desired structure. The relative magnitude of the communication within the network was almost negligible compared with the duration of the calculations. Hence, this algorithm showed considerable improvements, the average reduction in analysis time being 66 % with a peak of around 75 %. These considerable reductions were partially due to the generic nature of the implemented algorithm, allowing it to internally

decide how to solve the current iteration within the analysis. This flexible self-scaling structure allowed the algorithm to optimise every iteration of the problem, resulting in the smallest possible analysis duration.

10 References

[1] Walshe D.E.J., *Wind-excited Oscillations of Structures.*, National Physical Laboratory, Department of Trade and Industry, London, 1972.

[2] Burden A.R.," Modern Japanese Suspension Bridge Design.", *Proceedings of the Institution of Civil Engineers, Part 1*, Vol. 90, Feb 1991, pp.157-177.

[3] Upstone T.J. and Burden A.R.," Modern Japanese Suspension Bridge - Discussion.", *Proceedings of the Institution of Civil Engineers, Part 1-Design and Construction*, Vol. 90, October 1991, pp.1099-1100.

[4] Poulsen N.K., Damsgaard A. and Reinhold T.A.," Determination of Flutter Derivatives for the Great Belt Bridge.", *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 41, No. 1-3, 1992, pp.153-164.

[5] Miyata T. and Yamaguchi M.," Aerodynamics of wind effects on the Akashi Kaikyo Bridge. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1993, Vol.48, No.2-3, pp.287-315

[6] Pecora M., Lecce L., Marulo F. and Coiro D.P." Aeroelastic Behaviour of Long Span Bridges with 'Multibox' Type Deck Sections."

[7] Scruton C., " An Introduction to Wind Effects on Structures.", Engineering Design Guides, Oxford University Press, 1980, p.79

[8] The Department of Transport., *Design Rules for Aerodynamic Effects on Bridges.*, HMSO, London, January 1993.

[9] Dowell, E.H., Curtiss, H.C. Jr., Scanlan, R.H. and Sisto, F., *A Modern Course in Aeroelasticity.*, 2nd Revised and Enlarged Edition, Kluwer Academic Publishers, Dordrecht, Netherlands, 1989, 559 p.

[10] Theodorsen T., *General Theory of Aerodynamic Instability and the Mechanism of Flutter*, NACA Report No. 496, 1935.

[11] Hubner O.," The Newton Method for Solving the Theodorsen Integral-Equation.", *Journal of Computational and Applied Mathematics*, Vol. 14, No. 1-2, 1986, pp.19-30.

[12] Bera R.K.," A New Look at Theodorsen Method in Aerofoil Theory.", *International Journal for Numerical Methods in Fluids,* Vol. 9, No. 3, pp.251-262.

[13] Eversman W. and Tewari A.," Modified Exponential Series Approximation for the Theodorsen Function.", *Journal of Aircraft*, Vol. 28, No. 9, pp.553-557.

[14] Hassig H.J.," An Approximate True Damping Solution of the flutter equation by iteration.", *Journal of Aircraft*, Vol. 8, No. 11, November 1971, pp.885-889.

[15] Bleich F., "Dynamic Instability of Truss-Stiffened Suspension Bridges Under Wind Action.", *Proceedings ASCE*, Vol 75, No 8, 1948.

[16] Scruton C.," An Experimental Inverstigation of the Aerodynamic Stability of Suspension Bridges with Special Reference to the Proposed Severn Bridge.", *Proceedings of the Institution of Civil Engineers*, Vol. 1, No. 2, March 1952, pp.189-222.

[17] Smith I.P., "The Aeroelastic Stability of the Severn Suspension Bridge", National Physical Laboratory, Aerodynamics Division, May 1964, 20 p

[18] Walshe D.E.," A Résumé of the Aerodynamic Investigations for the Forth Road and The Severn Bridges.", *Proceedings of the Institution of Civil Engineers*,Vol. 42, No. 1-3, 1992, pp. 87-108. paper 7001

[19] Selberg A., "Oscillation and Aerodynamic Stability of Suspension Bridges.", Acta Polytechnica scandinavica, Civil Engineering and Building Construction Series No. 13
[20] Sabzevari, A.M. and Scanlan R.H.," Aerodynamic Instability of Suspension Bridges.", *Journal of the Engineering Mechanics Division - ASCE*, EM2, Vol. 94, April 1968, pp.489-519.
[21] Sabzevari, A.M. and Scanlan R.H.," Aerodynamic Investigations of Box Girder Bridges.", *Journal of the Structural Division - ASCE*, ST7, Vol. 95, July 1969, pp.1517-1529.
[22] Scanlan R.H. and Tomko J.J.," Airfoil and Bridge Deck Flutter Derivatives.", *Journal of the Engineering Mechanics Division - ASCE*, EM6, Vol. 97, December 1971, pp.1717-1737.
[23] Simiu, E. and Scanlan R.H., *Wind Effects on Structures: An introduction to Wind Engineering*., John Wiley & Sons, New York, 1978, 458 p.
[24] Scanlan R.H.," Role of Indical Functions in Buffeting Analysis of Bridges.", *Journal of Structural Engineering - ASCE*, No. 7, Vol. 110, July 1984, pp.1433-1446.
[25] Scanlan R.H. and Jones N.P.," A Minimum Design Methodology for Evaluating Bridge Flutter and Buffeting Response.", *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 36, No. 1-3, 1990, pp.1341-1353.
[26] Scanlan R.H.," Problematics in Formulation of Wind-Force Models for Bridge Decks.", *Journal of the Engineering Mechanics - ASCE*, No. 7, Vol. 119, July 1993, pp.1353-1375.
[27] Sarkar P.P., Jones N.P. and Scanlan R.H.," Identification of Aeroelastic Parameters of Flexible Bridges.", *Journal of the Engineering Mechanics - ASCE*, No. 8, Vol. 120, August 1994, pp.1718-1742.
[28] Irwin D.A., "Full Aeroelastic Model Tests.", *Proceedings of the International Symposium of Aerodynamics of Large Bridges*, A.A.Balkema, Rotterdam 1992, pp 125-135.
[29] Hjort-Hansen E., "Sectional Model Tests.", *Proceedings of the International Symposium of Aerodynamics of Large Bridges*, A.A.Balkema, Rotterdam 1992, pp 113-124.
[30] Cook R.D., Malkus D.S. and Plesha M.E., *Concepts and Applications of Finite Element Analysis*, Third Edition, John Wiley & Sons, New York, 1989, 630p.
[31] Agar T.J.A., *ANSUSP Users Guide*., Draft Manual, Mott, Hay & Anderson, London, 1980
[32] Beith J.G., *ANSUSP Version 2.0 Users Guide*., Draft Manual, University of Glasgow, 1995.
[33] Beith J.G., *ANSUSP Version 3.0 Users Guide*., Manual of PC & UNIX Versions, University of Glasgow, 1997.
[34] Beith J.G., *PAR-ANSUSP Version 1.0 Users Guide*., Manual for Parallel Version, University of Glasgow, 1997.
[35] Agar T.J.A.," The Analysis of Aerodynamic flutter of Suspension Bridges.", *Computers and Structures*, Vol. 30, No. 3, 1988, pp.593-600.
[36] Chen Z.Q. and Agar T.J.A.," Geometric Nonlinear Analysis of Flexible Spatial Beam Structures.", *Computers and Structures*, Vol. 49, No. 6, 1993, pp.1083-1094.
[37] Agar T.J.A.," Dynamic Instability of Suspension Bridges.", *Computers and Structures*, Vol. 41, No. 6, 1991, pp.1321-1328.

[38] Agar T.J.A.," Aerodynamic Flutter Analysis of Suspension Bridges by a Modal Technique.", *Engineering Structures*, Vol. 11, April 1989, pp.75-82.
[39] Beith J.G. and Agar T.J.A.," Long Span Bridge Dynamics Modal Eigen Response and Mass Participation.", Proceedings Civil-Comp 95, Volume - *Developments In Computational Techniques for Structural Engineering*, Civil-Comp Press, Edinburgh, August 1995, pp.77-83.
[40] Beith J.G. and Agar T.J.A.," Visualisation of Suspension Bridge Instability under Aerodynamic Forces.", Proceedings Civil-Comp 95, Volume - *Developments In Computational Techniques for Civil Engineering*, Civil-Comp Press, Edinburgh, August 1995, pp.299-303.
[41] Beith J.G. and Agar T.J.A.," Stability Analysis of Suspension Bridge Structures.", Proceedings 4th ACME-UK Conference, *Computational Mechanics in UK*, Glasgow, January 1996, pp.107-110.
[42] Beith J.G. and Agar T.J.A.," Modal Participation of Suspension Bridge Structures.", Proceedings 2nd NAFEMS International Conference, *Structural Dynamics Modelling*, Cumbria, July 1996, pp.141-152.
[43] Beith J.G. and Agar T.J.A.," A Practical Engineering Method for the Flutter Analysis of Long Span Bridges.", To be presented at the 8th U.S. National Conference on Wind Engineering, Baltimore, Maryland, 5-7th June 1997.
[44] Beith J.G. and Agar T.J.A.," Rapid Interpretation of Aerodynamic Response for Long Span Structures.", To be presented at the 8th U.S. National Conference on Wind Engineering, Baltimore, Maryland, 5-7th June 1997.
[45] Iwegbue I.E. and Brotton D.M.," A Numerical Integration method for computing the Flutter Speeds of Suspension Bridges Erection Conditions.", *Proceedings of The Institution of Civil Engineers, Part 2*, 1977, Vol 63, December, pp.785-802.
[46] Iwegbue I.E.," Dynamic Behaviour of Structural Frameworks.", PhD Thesis, UMIST, July 1976.
[47] Otter J.R.H., Cassell A.C. and Hobbs R.E.," Dynamic Relaxation - Discussion.", *Proceedings of The Institution of Civil Engineers*, pp.723-750.
[48] Day A.S.," Introduction to Dynamic Relaxation.", *Engineer*, 1965, 219, 29 January, pp.219-221.
[49] Bell A.J., Brotton D.M., " A Numerical Integration Method for the determination of flutter speeds.", *International Journal of Mechanical Science*, Vol. 15, March 1973, pp.473-483.
[50] Dumanoglu A.A. and Severn R.T.," Seismic Response of Modern Suspension Bridges to Asynchronous vertical ground motion.", *Proceedings of the Institution of Civil Engineers, Part 2*, Vol. 83, December 1987, pp.701-730.
[51] Dumanoglu A.A. and Severn R.T.," Seismic Response of Modern Suspension Bridges to Asynchronous Longitudinal and Lateral ground motion.", *Proceedings of the Institution of Civil Engineers, Part 2*, Vol. 87, March 1989, pp.73-86.
[52] Brownjohn J.M.W., Dumanoglu A.A., Severn R.T. and Taylor C.A.," Ambient Vibration Measurements of the Humber Suspension Bridge and comparison with calculated characteristics.", *Proceedings of the Institution of Civil Engineers, Part 2*, Vol. 83, September 1987, pp.561-600.
[53] Brownjohn J.M.W., Dumanoglu A.A and Severn R.T.," Ambient Vibration Survey of the Fatih Sultan Mehmet (2nd Bosporus) Suspension Bridge.", *Earthquake Engineering and Structural Dynamics*, Vol. 21, No. 10, 1992, pp.907-924.

[54] Brownjohn J.M.W., Dumanoglu A.A and Severn R.T.," Seismic Analysis of the Fatih Sultan Mehmet (2nd Bosporus) Suspension Bridge.", *Earthquake Engineering and Structural Dynamics*, Vol. 21, No. 10, 1992, pp.881-906.
[55] Brownjohn J.M.W., Dumanoglu A.A., Severn R.T. and Blakeborough A.," Ambient Vibration Survey of the Bosporus Suspension Bridge.", *Earthquake Engineering and Structural Dynamics*, Vol. 18, 1989, pp.263-283.
[56] Dumanoglu A.A and Severn R.T.," Stochastic Response of Suspension Bridges to Earthquake Forces.", *Earthquake Engineering and Structural Dynamics*, Vol. 19, 1990, pp.133-152.
[57] Abdelghaffar A.M. and Rubin L.I.," Torsional Earthquake Response of Suspension Bridges. ", *Journal of the Engineering Mechanics Division-ASCE*, 1984, Vol. 110, No. 10, pp. 1467-1484
[58] Wilson J.C. and Tan B.S.," Bridge Abutments - Formulation of Simple-Model for Earthquake Response Analysis.", *Journal of Engineering Mechanics-ASCE*, 1990, Vol.116, No.8, pp.1828-1837
[59] Wilson J.C. and Tan B.S.," Bridge Abutments - Assessing Their Influence on Earthquake Response of Meloland Road Overpass. ", *Journal of Engineering Mechanics-ASCE*, 1990, Vol.116, No.8, pp.1838-1856
[60] Abdelghaffar A.M., Rood J.D., " Simplified Earthquake Analysis of Suspension Bridge Towers. ", *Journal of the Engineering Mechanics Division-ASCE*, 1982, Vol.108, No.2, pp.291-308
[61] Abdelghaffar A.M., Rubin L.I., " Suspension Bridge Response to Multiple-Support Excitations. ", *Journal of the Engineering Mechanics Division-ASCE*, 1982, Vol.108, No.2, pp.419-435
[62] Jennings A. and McKeown J.J., *Matrix Computation 2nd Edition.*, John Wiley & Sons, New York, 1992, 427 p.
[63] Corr R.B., *Specification of SIMK1 (Structural Version).*, Internal Report, Department of Civil Engineering, Queens University Belfast, April 1971.
[64] Jennings A., *The Development and Application of Simultaneous Iteration for Eigenvalue Problems.*, Internal Report, Department of Civil Engineering, Queens University Belfast, 1971.
[65] Corr R.B. and Jennings A.," A Simultaneous Iteration Algorithm for Symmetric Eigenvalue Problems.", *International Journal for Numerical Methods in Engineering*, Vol. 10, 1976, pp.647-663.
[66] Wilkinson J.H., *The Algebraic Eigenvalue Problem*, Oxford University Press, 1965, 662p.
[67] Golub G.H. & Van Loan C.F., *Matrix Computations*, The Johns Hopkins University Press, 1983, 474p.
[68] Francis J.G.F., "The QR transform, Part I and II.", *Computer Journal*, Vol 4, 1961, pp265-271 and 332-345.
[69] Abo-Hamd M. and Utku S.," Analytical Study of Suspension Bridge Flutter.", *Journal of the Engineering Mechanics Division - ASCE*, EM3, June 1978, pp.537-550.
[70] Clough R.W. and Penzien J., *Dynamics of Structures*, McGraw-Hill, New York, 1975, 195 p.
[71] Craig R.J. Jr., *Structural Dynamics: An Introduction to computer Methods*, John Wiley & Sons, 1981, 527 p.
[72] Geist A., Beguelin A., Dongarra J., Jiang W., Manchek R. & Sunderam V., " PVM: Parallel Virtual Machine. A Users guide and tutorial for networked parallel computing. MIT Press, 1994, 280 p.

[73] Platform Computing Corporation, *LSF Users Guide*. Fourth Edition, Platform Computing Corporation, *December 1996, 165 p.*
[74] Ewing A.K., Richardson H., Simpson A.D. & Kulkarni R. " Writing data parallel programs with high performance Fortran: Version 1.3. ", Edinburgh parallel computing centre, The University of Edinburgh, 1996, 138 p.
[75] MacDonald N., Minty E., Harding T. & Brown S. " Writing Message-Passing Parallel programs with MPI. ", Edinburgh parallel computing centre, The University of Edinburgh, 1996, 92 p.
[76] Burns A. and Davies G., *Concurrent Programming*, Addison-Wesley, 1993, 377 p.
[77] Bustard D., Elder J. and Welsh., *Concurrent Program Structures*, Prentice Hall, New York, 1988, 321 p.
[78] Bräunl T., *Parallel Programming: An Introduction*, Prentice Hall, New York, 1993, 270 p.
[79] Lester B.P., *The Art of Parallel Programming*, Prentice Hall, New York, 1993, 375 p.
[80] Bjørstad P., Manne F., Sørevik T. and Vajteršic M.," Efficient Matrix Multiplication on SIMD Computers. ", *SIAM Journal Matrix Analysis and Applications*, Vol. 13, No. 1, January 1992, pp.386-401.
[81] Bunse-Gerstner A., Byers R. and Mehrmann V.," Numerical Methods for Simultaneous Diagonalisation.", *SIAM Journal Matrix Analysis and Applications*, Vol. 14, No. 4, October 1993, pp.927-949.
[82] Chawla M.M.," A Parallel Gaussian Elimination Method for General Linear Systems.", *International Journal of Computer Mathematics* , Vol. 42, 1992, pp.71-82.
[83] Gilbert R.J. and Schreiber R.," Highly Parallel Sparse Cholesky Factorisation.", *SIAM Journal on Scientific and Statistical Computing*, Vol. 13, No. 5, September 1992, pp.1151-1172.
[84] Lang B.," A Parallel Algorithm for Reducing Symmetric Banded Matrices to Tridiagonal Form.", *SIAM Journal on Scientific Computing*, Vol. 14, No. 6, November 1993, pp.1320-1338.
[85] Choi J., Dongarra J. & Walker D.W., " The Design of a parallel dense linear algebra software library: reduction to Hessenberg, Tridiagonal and Bidiagonal Form. ", Oak Ridge National Laboratory, Tennessee, January 1995, ORNL/TM-12472, pp.22.
[86] Pan V. and Reif J.," Fast and Efficient Parallel Solution of sparse linear systems. ", *SIAM Journal on Computing*, 1993, Vol.22, No.6, pp.1227-1250
[87] Duff I.S. and Scott J.A.," Computing Selected Eigenvalues of Sparse Unsymmetric Matrices using Subspace Iteration. ", *ACM Transactions on Mathematical Software*, 1993, Vol.19, No.2, pp.137-159
[88] Watson B.C. & Noor A.K.," Nonlinear Structural Analysis on Distributed-Memory Computers. ", *Computers & Structures*, 1995, Vol. 58, No. 2, pp.233-247
[89] Stefano G.D.," Dynamic Response Analysis on Nonlinear Structures Using Step-By-Step Integration Techniques. ", *Computers & Structures*, 1995, Vol. 57, No. 6, pp.1063-1070
[90] Jaques M.W.S., Ross C.T.F & Strickland P.," Exploiting Inherent Parallelism on Non-linear Finite Element Analysis. ", *Computers & Structures*, 1996, Vol. 58, No. 4, pp.801-807

[91] Demmel J.W.," Trading of parallelism and numerical stability. ", LAPACK working note 52, University of Tennessee Technical Report CS-92-179, 1992, pp.26.
[92] Anderson E., Dongarra J. & Ostrouchov S., "Installation Guide for LAPACK. ", LAPACK working note 41, University of Tennessee Technical Report, 1994, pp.123.
[93] Anderson E. & Dongarra J., "Performance of LAPACK: a portable library of numerical linear algebra routines. ", LAPACK working note 44, University of Tennessee Technical Report CS-92-156, May 1992, pp.16.
[94] Choi J., Dongarra J., Pozo R. & Walker D.W., "ScaLAPACK: A scalable linear algebra library for distributed memory concurrent computers. ", LAPACK working note 55, University of Tennessee Technical Report, 1992.
[95] Dongarra J.J. and Sidani M.," A Parallel Algorithm for the Nonsymmetric Eigenvalue Problem.", *SIAM Journal on Scientific Computing*, Vol. 14, No. 3, May 1993, pp.542-569.
[96] Dongarra J.J. and Sidani M.," A Parallel Algorithm for the Non-symmetric Eigenvalue Problem. ", University of Tennessee Technical Report, Technical Report CS-91-137, September 1991, pp. 43.
[97] Dongarra J. & Ostrouchov S.," Quick Installation guide for LAPACK on UNIX systems. ", LAPACK working note 81, University of Tennessee Technical Report, September 1994, pp. 19.
[98] Huo Y. and Schreiber R.," Efficient Massively Parallel Eigenvalue Computation. ", *The International Journal of Supercomputer Applications*, Vol. 7, No. 4, 1993, pp.292-303.
[99] Kalamboukis T.Z.," A Parallel Algorithm for the dense Symmetric Eigenvalue Problem on a Transputer Array.", *Parallel Computing*, Vol. 18, No. 2, 1992, pp.207-212.
[100] Götze J., Paul S. and Sauer M.," An efficient Jacobi-Like Algorithm for Parallel Eigenvalue Computation.", *IEEE Transactions on Computing*,Vol. 42, No. 9, September 1993, pp.1058-1065.
[101] Hendrickson B.," Parallel QR Factorization using Torus-Wrap Mapping.", *Parallel Computing*, Vol. 19, No. 11, 1993, pp.1259-1271. Petiton S.G.," Parallel Subspace Method for Non-Hermitian Eigenproblems on the Connection Machine (CM2).", *Applied Numerical Mathematics*, Vol. 10, No. 1, 1992, pp.19-35.
[102] Philippe B. and Vital B.," Parallel Implementations for Solving Generalized Eigenvalue Problems with Symmetric Sparse Matrices.", *Applied Numerical Mathematics*, Vol. 12, No. 5, 1993, pp.391-402.
[103] Rajendran S. and Narasimhan M.V.," An Accelerated Subspace Iteration Method.", *International Journal for Numerical methods in Engineering*, Vol. 37, 1994, pp.141-153.
[104] Swarztrauber P.N.," A Parallel Algorithm for Computing the Eigenvalues of a Symmetric Tridiagonal Matrix.", *Mathematics of Computation*, Vol. 60, No.202, April 1993, pp.651-668.
[105] Yau S. and Lu Y.Y.," Reducing the symmetric matrix eigenvalue problem to matrix multiplications.", *SIAM Journal on Scientific Computing*, Vol. 14, No. 1, January 1993, pp.121-136.
[106] Natarajan R," A Parallel Algorithm For The Generalized Symmetric Eigenvalue Problem On A Hybrid Multiprocessor. ", *Parallel Computing*, 1990, Vol.14, No.2, pp.129-150

[107] Kaufman L," A Parallel QR Algorithm For The Symmetrical Tridiagonal Eigenvalue Problem. ", *Journal of Parallel And Distributed Computing*, 1994, Vol.23, No.3, pp.429-434

[108] Choi J., Dongarra J., Ostrouchov S., Petitet A.P., Walker D.W. & Whaley R.C., " The Design and implementation of the SCALAPACK LU, QR and Cholesky factorization routines. ", Oak Ridge National Laboratory, Tennessee, September 1994, ORNL/TM-12470, pp.22.

[109] Henry G. & Van De Geijn R.," Parallelizing the QR algorithm for the unsymmetric algebraic eigenvalue problem: myths and reality, LAPACK working note 79, University of Tennessee Technical Report, 1994.

[110] Flynn M., "Very High Speed Computing Systems.", *proceedings of the IEEE*, 1966, Vol 54, pp.1901-1909.

[111] Viehöver G., "PARIX programmers Guide- Release 1.1", Parsytec Computer GmbH, September 1992.

11 Bibliography

[1] Giesing J.P.," Nonlinear Two-Dimensional Unsteady Potential Flow with Lift.", *Journal of Aircraft*, Vol. 5, No. 2, 1968, pp.135-143.

[2] Abdelghaffar AM, " Suspension Bridge Vibration - Continuum Formulation. ", *Journal of the Engineering Mechanics Division-ASCE*, 1982, Vol.108, No.6, pp.1215-1232

[3] Li QC, " Measuring Flutter Derivatives For Bridge Sectional Models In Water Channel. ", *Journal of Engineering Mechanics-ASCE*, 1995, Vol.121, No.1, pp.90-101

[4] Nazir CP, " Multispan Balanced Suspension Bridge. ", *Journal of Structural Engineering-ASCE*, 1986, Vol.112, No.11, pp.2512-2527

[5] Gjelsvik A.," Development Length for Single Wire in Suspension Bridge Cable.", *Journal of Structural Engineering - ASCE*, Vol. 117, No. 4, 1991, pp.1189-1200.

[6] Raoof M. and Huang Y.P.," Wire Recovery Length in Suspension Bridge Cable.", *Journal of Structural Engineering - ASCE*, Vol. 118, No. 12, 1992, pp.3255-3267.

[7] Matsuzaki M., Uchikawa C. and Mitamura T.," Advanced Fabrication and Erection Techniques for Long Suspension Bridge Cables.", *Journal of Construction Engineering and Management - ASCE*, Vol. 116, No. 1, 1990, pp.112-129.

[8] Birdsall B.," Advanced Fabrication and Erection Techniques for Long Suspension Bridge Cables - Discussion.", *Journal of Construction Engineering and Management - ASCE*, Vol. 118, No. 1, 1992, pp.200-205.

[9] Larsen A. and Gimsing N.J.," Wind Engineering Aspects of the East-Bridge-Tender Project. " *Journal of Wind Engineering and Industrial Aerodynamics*, 1992, Vol. 42, No. 1-3, pp. 1405-1416

[10] Leto I.V.," Preliminary Design of the Messina Strait Bridge.", *Proceedings of the Institution of Civil Engineers,* Vol. 102, Issue 3, 1994, pp.122-129.

[11] Leto I.V.," Preliminary Design of the Messina Strait Bridge - Discussion.", *Proceedings of the Institution of Civil Engineers,* Vol. 108, Issue 1, 1995, pp.39-42.

[12] Gasparetto M. and Bocciolone M.," Wind Measurements on Messina Straits. " *Journal of Wind Engineering and Industrial Aerodynamics*, 1992, Vol. 41, No. 1-3, pp. 393-404

[13] Bocciolone M., Gasparetto M., Lagomarsino S., Piccardo G., Ratto C.F. and Solari G.," Statistical-Analysis of Extreme Wind Speeds in the Straits of Messina. " *Journal of Wind Engineering and Industrial Aerodynamics*, 1993, Vol. 48, No. 2-3, pp. 359-377

[14] Diana G., Falco M., Bruni S., Cigada A., Larose G.L., Damsgaard A. and Collina A.," Wind- Tunnel Investigations of the Tower for the Stretto-Di-Messina Bridge. " *Journal of Wind Engineering and Industrial Aerodynamics*, 1993, Vol.48, No. 2-3, pp. 379-393

[15] Brancaleoni F. and Diana G.," The Aerodynamic Design of the Messina-Straits Bridge. " *Journal of Wind Engineering and Industrial Aerodynamics*, 1993, Vol. 48, No. 2-3, pp. 395-409

[16] Diana G., Falco M., Bruni S., Cigada A., Larose G.L., Damsgaard A. and Collina A.," Comparisons Between Wind-Tunnel Tests on a Full Aeroelastic

Model of the Proposed Bridge over Stretto-Di-Messina and Numerical Results. " *Journal of Wind Engineering and Industrial Aerodynamics*, 1995, Vol. 54, pp. 101-113

[17] Larose G.L., Falco M. and Cigada A.," Aeroelastic Response of the Towers for the proposed Bridge over Stretto-Di-Messina. " *Journal of Wind Engineering and Industrial Aerodynamics*, 1995, Vol. 57, No. 2-3, pp. 363-373

[18] Virlogeux M.," Wind Design and Analysis for the Normandy Bridge.", *Proceedings of the 1992 International Symposium on Aerodynamics of Large Bridges (ISALB)*, Balkema, Rotterdam, 1992, pp.183-216

[19] Abild J., Andersen E.Y. and Rosbjerg D.," The Climate of Extreme Winds at the Great Belt, Denmark. " *Journal of Wind Engineering and Industrial Aerodynamics*, 1992, Vol. 41, No. 1-3, pp. 521-532

[20] Gimsing N.J.," Wind Design of the Great Belt East Bridge - A Historic Retrospect. " *Journal of Wind Engineering and Industrial Aerodynamics*, 1993, Vol. 48, No. 2-3, pp. 253-259

[21] Larsen A.," Aerodynamic Aspects of the Final Design of the 1624m Suspension Bridge Across the Great Belt. " *Journal of Wind Engineering And Industrial Aerodynamics*, 1993, Vol. 48, No. 2-3, pp. 261-285

[22] Brown W.C. and Parsons M.F.," Bosporous bridge part 1: History of Design.", *Proceedings of the Institution of Civil Engineers, Part 1*, Vol. 58, 1975, pp.505-532.

[23] Knox H.S.G.," Bosporous bridge Part II: Construction of Superstructure.", *Proceedings of the Institution of Civil Engineers, Part 1*, Vol. 58, 1975, pp.533-567.

[24] Alwar R.S., Ramachandra Rao N. and Subba Rao M.," An Alternative Procedure in Dynamic Relaxation.", *Computers and Structures*, Vol. 5, 1975, pp.271-274.

[25] Ramesh G. and Krishnamoorthy C.S.," Post-Buckling Analysis of Structures by Dynamic Relaxation.", *International Journal for Numerical Methods in Engineering*, Vol. 36, 1993, pp.1339-1364.

[26] Kobayashi H. and Turvey G.J.," On the Application of a Limiting Process to the Dynamic Relaxation Analysis of Circular Membranes, Circular Plates and Spherical Shells.", *Computers and Structures*, Vol. 48, 1993, pp.1107-1116.

[27] Chaudhury N.K., Brotton D.M. and Merchant W.," A Numerical Method for Dynamic Analysis of Structural Frameworks.", *International Journal of Mechanical Science*, Vol. 8, No. 3, March 1966, pp.149-162.

[28] Chaudhury N.K. and Brotton D.M.," Analysis of Vertical Flexural Oscillations of Suspension Bridges by Digital Computer.", pp.209-219.

[29] Scanlan R.H.," Bridge Buffeting by Skew Winds in Erection Stages.", *Journal of the Engineering Mechanics - ASCE*, No. 2, Vol. 119, Febuary 1993, pp.251-269.

[30] Kumarasena T., Scanlan R.H. and Ehsan F.," Wind-Induced Motions of Deer Isle Bridge.", *Journal of Structural Engineering - ASCE*, No. 11, Vol. 117, November 1991, pp.3356-3374.

[31] Ehsan F., Jones N.P. and Scanlan R.H.," Effect of Sidewalk Vents on Bridge Response to Wind.", *Journal of Structural Engineering - ASCE*, No. 2, Vol. 119, February 1993, pp.484-504.

[32] Kumarasena T., Scanlan R.H. and Ehsan F.," Recent Observations in Bridge Deck Aeroelasticity.", *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 40, 1992, pp.225-247.

[33] Scanlan R.H.," State-of-the art methods for calculating Flutter, Vortex-induced, and Buffeting response of Bridge Structures.", Report Number FHWA/RD 80/50, Federal Highway Administration, Washington D.C., 1981.
[34] Scanlan R.H. and Jones N.P.," Stochastic aspects of bridge deck aeroelasticity under turbulent flow.", *Probabilistic Engineering Mechanics*, No. 3-4, Vol. 6 , 1991, pp.129-133.
[35] Jones N.P., Jain A. and Scanlan R.H.," Wind cross-spectrum effects on Long-Span Bridges.", Proceedings ASCE Engineering Mechanics Special Conference, May 1992, pp.63-66.
[36] Bryja D. and Sniady P.," Spatially Coupled Vibrations of a Suspension Bridge Under Random Highway Traffic.", *Earthquake Engineering and Structural Dynamics*, Vol. 20, No. 11, 1991, pp.999-1010.
[37] Abdelghaffar A.M.," Free Torsional Vibrations of Suspension Bridges. ", *Journal of the Structural Division-ASCE*, 1979, Vol. 105, No. ST4, pp. 767-788
[38] Ohishima H., Sato K. and Watanabe N.," Structural Analysis of Suspension Bridges. ", *Journal of the Engineering Mechanics Division-ASCE*, 1984, Vol. 110, No. 3, pp. 392-404
[39] Buckland P.G., Hooley R., Morgenstern B.D., Rainer J.H. and Van Selst A.M., " Suspension Bridge Vibrations - Computed and Measured.", *Journal of The Structural Division-ASCE*, 1979, Vol.105, No.ST5, pp.859-874.
[40] Scanlan R.H., " Suspension Bridge Vibrations - Computed and Measured - Discussion.", *Journal of The Structural Division-ASCE*, 1981, Vol.107, No.11, pp.2287-2289
[41] Hayashikawa_T, Watanabe_N, " Suspension Bridge Response to Moving Loads. ", *Journal of the Engineering Mechanics Division-ASCE*, 1982, Vol.108, No.6, pp.1051-1066
[42] Herzog M.A.M.," Aeroelastic Stability of Suspension Bridges Simplified.", *Proceedings of the Institution of Civil Engineers, Part 2*, Vol. 89, September 1990, pp.341-353.
[43] Fujino Y., Ito M., Shino I., Iwamoto M., Hikami Y.I., Tatsumi M. and Miyata T., "Wind-Tunnel Study of Long-Span Suspension Bridge Under Smooth and Turbulent-Flow.", *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 33, No. 1-2, August 1990, pp.313-322.
[44] Kobayashi H. and Nagaoka H.," Active Control of Flutter of a Suspension Bridge.", *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 41, No. 1-3, 1992, pp.143-151.
[45] Miyata T. and Yamada H.," Coulped Flutter Estimate of a Suspension Bridge.", *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 33, No. 1-2, 1990, pp.341-348.
[46] Fujino Y., Iwamoto M., Ito M. and Hikami Y.," Wind-Tunnel Experiments using 3D Models and Response Prediction for a Long-Span Suspension Bridge.", *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 42, No. 1-3, 1992, pp.1333-1344.
[47] Starossek U." Prediction of Bridge Flutter Through use of Finite Elements.", *Structural Engineering Review*, Vol. 5, No. 4, 1993, pp.301-307.
[48] Tanaka H., Yamamura N. and Shiraishi N.," Mutli-Mode Flutter Analysis and Two & Three Dimensional Model Tests on Bridges with Non-Analogous Modal Shapes.", *Japan Society of Civil Engineers*, Vol. 10, No. 2, July 1993, pp.35-46.

[49] Borri C., Majowiecki M. and Spinelli P.," The Aerodynamic Advantages Of A Double-Effect Large-Span Suspension Bridge Under Wind Loading. ", *Journal Of Wind Engineering And Industrial Aerodynamics*, 1993, Vol.48, No.2-3, pp.317-328

[50] Diana G., Bruni S., Cigada A. and Collina A.," Turbulence effect of flutter velocity in long span suspended bridges. " *Journal of Wind Engineering and Industrial Aerodynamics*, 1993, Vol. 48, No. 2-3, pp. 329-342

[51] Piccardo G.," A methodology for the study of coupled aeroelastic phenomena. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1993, Vol.48, No.2-3, pp.241-252

[52] Pecora M., Lecce L., Marulo F. and Coiro D.P.," Aeroelastic Behaviour of long span bridges with 'multibox' type deck sections. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1993, Vol.48, No.2-3, pp.343-358

[53] Diana G., Cheli F., Zasso A., Collina A. and Brownjohn J., " Suspension Bridge Parameter-Identification in Full Scale Test.", *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 41, No. 1-3, 1992, pp.165-176.

[54] Matteo J., Deodatis G. and Billington D.P.," Safety Analysis of Suspension-Bridge Cables: Williamsburg Bridge.", *Journal of Structural Engineering - ASCE*, Vol. 120, No. 11, November 1994, pp.3197-3211.

[55] Hasegawa K., Kojima H., Sasaki M. and Takena K.," Frictional Resistance Between Cable and Saddle Equipped with Friction Plate.", *Journal of Structural Engineering - ASCE*, Vol. 121, No. 1, January 1995, pp.1-14.

[56] Buonopane S.G. and Billington D.P.," Theory and History of Suspension Bridge Design from 1823 to 1940.", *Journal of Structural Engineering - ASCE*, Vol. 119, No. 3, 1993, pp.954-977.

[57] Puri S.P.S.," Theory and History of Suspension Bridge Design from 1823 to 1940- Discussion.", *Journal of Structural Engineering - ASCE*, Vol. 121, No. 1, 1995, pp.155-157.

[58] Brotton D.M.," A General Computer Programme for the solution of Suspension Bridge Problems.", *The Structural Engineer*, 1966, Vol 44, May, pp.161-167.

[59] Brancaleoni F. and Brotton D.M., " The Role of Time Integration In Suspension Bridge Dynamics. ", *International Journal for Numerical Methods in Engineering*, 1984, Vol.20, No.4, pp.715-732

[60] Sargious M.A., Dilger W.H. and Hawk H.," Box Girder Bridge Diaphragms with Openings. ", *Journal of the Structural Division-ASCE*, 1979, Vol. 105, No. ST1, pp. 53-65

[61] Simiu, E. and Scanlan R.H., *Wind Effects on Structures.*, John Wiley & Sons, New York, 1978, 458 p.

[62] Larsen A. (Editor)," Aerodynamics of Large Bridges.", *Proceedings of the 1992 International Symposium on Aerodynamics of Large Bridges (ISALB)*, Balkema, Rotterdam, 1992, 305 p.

[63] Garevski M., Dumanoglu A.A. and Severn R.T.," Dynamic Characteristics and seismic behaviour of Jindo Bridge, South Korea.", *Structural Engineering Review*, Vol. 1, 1988, pp.141-149.

[64] Mitchellbaker_D, Cullimore_MSG, " Operation and Maintenance of the Clifton Suspension Bridge. ", *Proceedings of the Institution of Civil Engineers Part 1-Design and Construction*, 1988, Vol.84, No.APR, pp.291-308

[65] Cullimore_MSG, Mason_PJ, " Fatigue and Fracture Investigation Carried out on the Clifton Suspension Bridge. ", *Proceedings of the Institution of Civil*

Engineers Part 1-Design and Construction, 1988, Vol.84, No.APR, pp.309-329

[66] Littler J.D., Brownjohn J.M.W., Dumanoglu A.A., Severn R.T., Taylor C.A. and Maguire J.R. " Ambient Vibration Measurements of the Humber Suspension Bridge and Comparison with Calculated Characteristics. ", *Proceedings of the Institution of Civil Engineers Part 2- Research and Theory*, 1987 Vol.83 p.561.

[67] Littler J.D., Brownjohn J.M.W., Dumanoglu A.A., Severn R.T., Taylor C.A. and Maguire J.R. " Ambient Vibration Measurements of the Humber Suspension Bridge and Comparison with Calculated Characteristics - Discussion. ", *Proceedings of the Institution of Civil Engineers Part 2-Research and Theory*, 1988, Vol.85, No.DEC, pp.725-730

[68] Brownjohn J.M.W., Dumanoglu A.A., Severn R.T., Taylor C.A., " Ambient Vibration Measurements of the Humber-Suspension-Bridge and Comparison with Calculated Characteristics. ", *Proceedings of the Institution of Civil Engineers Part 2-Research and Theory*, 1987, Vol.83, No. SEP, pp.561-600

[69] Brancaleoni F. and Brotton D.M., " Analysis and Prevention of Suspension Bridge Flutter in Construction. ", *Earthquake Engineering & Structural Dynamics*, 1981, Vol.9, No.5, pp.489-500

[70] Wilson J.C.," Analysis of the Observed Seismic Response of a Highway Bridge. ", *Earthquake Engineering & Structural Dynamics*, 1986, Vol.14, No.3, pp.339-354

[71] Higashihara_H, Moriya_T, Tajima_J, " Ambient Vibration Test Of An Anchorage Of South Bisan-Seto Suspension Bridge.", *Earthquake Engineering & Structural Dynamics*, 1987, Vol.15, No.6, pp.679-695

[72] Wilson J.C.," Stiffness of Non-Skew Monolithic Bridge Abutments for Seismic Analysis. ", *Earthquake Engineering & Structural Dynamics*,, 1988, Vol.16, No.6, pp.867-883

[73] Wilson J.C. and Gravelle W., " Modeling of a Cable-Stayed Bridge for Dynamic Analysis. ", *Earthquake Engineering & Structural Dynamics*, 1991, Vol.20, No.8, pp.707-722 ???

[74] Wilson J.C. and Liu T., " Ambient Vibration Measurements on a Cable-Stayed Bridge. ", *Earthquake Engineering & Structural Dynamics*, 1991, Vol.20, No.8, pp.723-747

[75] Brownjohn J.M.W.," Observations on Non-Linear Dynamic Characteristics of Suspension Bridges.", *Earthquake Engineering and Structural Dynamics*, Vol. 23, 1994, pp.1351-1367.

[76] Hirose_K, Fujii_M, Hara_T, Yokoyama_H, Matsui_T, Furusawa_H, Mohtai_T and Mizuya_Y, " Development of Installation Techniques for a 500-Kv Oil-Filled Cable on a Long-Span Suspension Bridge. ", *IEEE Transactions on Power Apparatus and Systems*, 1981, Vol.100, No.4, pp.1718-1728

[77] Minakuchi_D, Fujii_M, Inoguchi_H, Uno_H, Matsui_T, Toyoda_S, Gessei_I and Asai_S, " Installation of 187 Kv Xlpe Power Cable Along the Ohnaruto Suspension Bridge. " *IEEE Transactions on Power Delivery*, 1987, Vol.2, No.3, pp.603-609

[78] Kimbara S., Yanaka Y. and Kiyota R., " Dynamic Properties of A Suspension Bridge Constructed With Trapezoidal Steel Box Girders. ", *Journal of Constructional Steel Research*, 1994, Vol.30, No.3, pp.283-304

[79] Choi Y.S., Jen K.C and McKenna P.J., " The Structure of the Solution Set for Periodic Oscillations in a Suspension Bridge Model. ", *IMA Journal of Applied Mathematics,* 1991, Vol.47, No.3, pp.283-306
[80] McKenna_PJ, Walter_W, " Traveling Waves in a Suspension Bridge. ", *SIAM Journal on Applied Mathematics*, 1990, Vol.50, No.3, pp.703-715
[81] Franciosi_C, Franciosi_V, " Suspension Bridge Analysis Using Lagrangian Approach. ", *Computers & Structures*, 1987, Vol.26, No.3, pp.499-512
[82] Kemp_EL, " Roebling, Ellet, and the Wire-Suspension Bridge." *Annals of the New York Academy of Sciences*, 1984, Vol.424, No.MAY, pp.41-62
[83] Wilson J.C. and Jennings P.C.," Spatial Variation of Ground Motion Determined From Accelerograms Recorded on a Highway Bridge. ", *Bulletin of the Seismological Society of America*, 1985, Vol.75, No.6, pp.1515-1533
[84] Hubner O.," The Newton Method for Solving the Theodorsen Integral-Equation.", *Journal of Computational and Applied Mathematics*, Vol. 14, No. 1-2, 1986, pp.19-30.
[85] Fonda_A, Schneider_Z, Zanolin_F, " Periodic Oscillations For A Nonlinear Suspension Bridge Model. ", *Journal Of Computational And Applied Mathematics*, 1994, Vol.52, No.1-3, pp.113-140
[86] Semper B.A.," Mathematical-Model For Suspension Bridge Vibration. ", *Mathematical And Computer Modelling*, 1993, Vol.18, No.11, pp.17-28
[87] Semper_B, "Finite-Element Methods for Suspension Bridge Models. ", *Computers & Mathematics With Applications*, 1993, Vol.26, No.5, pp.77-91
[88] Buckland P.G., Morgenstern B.D., " Conversion Of A Suspension Bridge Into A Cable-Stayed Bridge. ", *Canadian Journal of Civil Engineering*, 1991, Vol.18, No.2, pp.273-281
[89] Paultre P., Chaallal O. and Proulx J.," Bridge Dynamics and Dynamic Amplification Factors - A Review of Analytical and Experimental Findings.", *Canadian Journal of Civil Engineering*, Vol. 19, No. 2, 1992, pp.260-278.
[90] McKenna_PJ, Walter_W, " Nonlinear Oscillations in a Suspension Bridge. ", *Archive For Rational Mechanics and Analysis*, 1987, Vol.98, No.2, pp.167-177
[91] Schuessler_R," The Great East River Suspension Bridge - Brooklyn Bridge Straddles 100 Years. ", *Oceans*, 1983, Vol.16, No.2, pp.26-30
[92] Iwaki_Y, Satoh_T, Minami_M, Tohyama_Y and Inoue_H, " A Cable Damper System For Preventing Wind-Induced Oscillations of Suspension-Bridge Towers During Erection. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1983, Vol.12, No.2, pp.165-188
[93] Huston_DR, Bosch_HR and Scanlan_RH," The Effects of Fairings and of Turbulence on the Flutter Derivatives of a Notably Unstable Bridge Deck. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1988, Vol.29, No.1-3, pp.339-349
[94] Miyata_T, Okauchi_I, Shiraishi_N, Narita_N and Narahira_T, " Preliminary Design Considerations for wind Effects on a Very Long-Span Suspension Bridge. " *Journal of Wind Engineering And Industrial Aerodynamics*, 1988, Vol.29, No.1-3, pp.379-388
[95] Ueda_T, Yasuda_M, Nakagaki_R, " Mechanism of Aerodynamic Stabilization for Long-Span Suspension Bridge with Stiffening Truss-Girder. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1990, Vol.33, No.1-2, pp.333-340

[96] Bosch_HR, " Section Model Studies of the Deer Isle-Sedgwick Suspension Bridge. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1990, Vol.36, No.1-3, pp.601-610
[97] Namini A, " Investigation of Analytical Modeling For Long-Span Bridge Flutter. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1992, Vol.42, No.1-3, pp.1277-1278
[98] Xiang_H, Zhao_Z and Zhu_L," Flutter Pattern of Cable-Stayed Bridge and a Numerical-Method for Flutter Analysis. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1992, Vol.42, No.1-3, p.1291
[99] Miyata_T, Matsumoto_H, Yasuda_M, " Circumstances of Wind-Resistant Design Examinations for Very Long Suspension Bridge. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1992, Vol.42, No.1-3, pp.1371-1382
[100] Hughes T.J.R and Jansen K.," Finite Element Methods in Wind Engineering.", *Journal of Wind Engineering and Industrial Aerodynamics*, Vol. 46, No. 7, August 1993, pp.297-313.
[101] Kwon S.D., Chang S.P., Kim Y.S. and Park S.Y., " Aerodynamic Stability Of Self-Anchored Double Deck Suspension Bridge. ", *Journal of Wind Engineering and Industrial Aerodynamics*, 1995, Vol.54, pp.25-34
[102] Iwamoto_M, Fujino_Y, " Identification of Flutter Derivatives of Bridge Deck From Free Vibration Data. ", *Journal of Wind Engineering and Industrial*
[103] Veer B., " The CDL Guide. ", Perihelion Software Limited, 1990, 126 p.
[104] Pugsley A., *The Theory of Suspension Bridges*, Edward Arnold Ltd., London, 1957, 131p.
[105] Bishop R.E.D., Gladwell G.M.L. & Michaelson S., *The Matrix Analysis of Vibration*, Cambridge University Press,1965, 404 p.
[106] Gourlay A.R. & Watson G.A., *Computational Methods for Matrix Eigenproblems*, John Wiley & Sons Ltd., New York, 1973, 132 p.
[107] Troitsky M.S., *Cable-Stayed Bridges: Theory and Design*, Crosby Lockwood Staples, London, 1977, 385p.
[108] Gimsing N.J., Cable Supported Bridges: Concept and Design, John Wiley & Sons, 1983, 400p.
[109] Gould P.L. & Abu-Sitta S.H., *Dynamic Response of structures to Wind and Earthquake Loading*., Pentech Press Ltd, London, 1980, 175p.
[110] Sesma J., *A parametric Study of Assumed Aerodynamic Flutter Derivatives for Suspension Bridge Flutter*, Final Year Project, University of Glasgow, May 1995.
[111] Agar T.J.A., *Computational Methods for obtaining Vibration Frequencies with Particular Reference to Ship Hulls*, PhD Thesis, the Queens University of Belfast, April 1978.
[112] T.R.R.L. Report, "Symposium on Dynamic Behaviour of Bridges", Supplementary Report 275, 1977
[113] Simpson B. and Walshe D.E., "Investigation into the wind effects on Breydon bascule bridge.", *Engineering Structures*, Vol 7, No 1, Jan 1985, pp.10-17.
[114] Cullen Wallace A..A., "Wind influence on Kessock bridge.", *Engineering Structures*, Vol 7, No 1, Jan 1985, pp 18-22.
[115] Curtis D.J., Hart J.J., Scruton C. and Walshe D.E., "An aerodynamic investigation of the suspend structure of the proposed Tsing Ma bridge.", *Engineering Structures*, Vol 7, No 1, Jan 1985, pp 23-34.

[116] Willford M.R., "The prediction of wind-induced responses of the new Hongkong and Shanghai Banking Corporation headquarters, Hong Kong.", *Engineering Structures*, Vol 7, No 1, Jan 1985, pp. 35-45.
[117] Anthony K.C., "The background to the statistical approach", *Proceedings of the Seminar at The Institution of Civil Engineers*, The modern design of wind-sensitive structures, June 1970, pp.17-28.
[118] Harris R.I., "The nature of wind.", *Proceedings of the Seminar at The Institution of Civil Engineers*, The modern design of wind-sensitive structures, June 1970, pp.29-56.
[119] Wootton L.R. & Scruton C., "Aerodynamic stability." *Proceedings of the Seminar at The Institution of Civil Engineers*, The modern design of wind-sensitive structures, June 1970, pp.65-82.
[120] Steffens R.J., "Structural vibration and damage.", Building Research Establishment Report,1974

Appendix 1 Derivation of Characteristic Displaced Shape from Complex Conjugate Eigenvalues and Eigenvectors

For a real non-symmetric general matrix the eigenvalues and corresponding eigenvectors are of complex conjugate form

$$\lambda = \mu \pm i\omega \qquad \xi = p \pm iq$$

The characteristic motion of the structure can be calculated from these complex conjugate components as follows

$$\xi(t) = (a + ib)e^{(\mu+i\omega)t}(p + iq) + (a - ib)e^{(\mu-i\omega)t}(p - iq)$$

Where a, b are arbitrary constants of integration.

Re-arranging this expression and simplifying we obtain

$$\xi(t) = e^{\mu t}\left\{e^{i\omega t}(a + ib)(p + iq) + e^{-i\omega t}(a - ib)(p - iq)\right\}$$

$$\xi(t) = e^{\mu t}\left\{e^{i\omega t}\left(ap - bq + i(aq + bp)\right) + e^{-i\omega t}\left(ap - bq - i(aq + bp)\right)\right\}$$

Grouping common sub-expressions

$$\xi(t) = e^{\mu t}\left\{(ap - bq)(e^{i\omega t} + e^{-i\omega t}) + i(aq + bp)(e^{i\omega t} - e^{-i\omega t})\right\}$$

Noting the two identities

$$(e^{i\omega t} + e^{-i\omega t}) = \left(\mathrm{Cos}(\omega t) + i \cdot \mathrm{Sin}(\omega t)\right) + \left(\mathrm{Cos}(-\omega t) + i \cdot \mathrm{Sin}(-\omega t)\right) = 2\mathrm{Cos}(\omega t)$$

$$(e^{i\omega t} - e^{-i\omega t}) = \left(\mathrm{Cos}(\omega t) + i \cdot \mathrm{Sin}(\omega t)\right) - \left(\mathrm{Cos}(-\omega t) + i \cdot \mathrm{Sin}(-\omega t)\right) = 2i \cdot \mathrm{Sin}(\omega t)$$

We obtain

$$\xi(t) = e^{\mu t}\{2\mathrm{Cos}(\omega t)(ap - bq) + i \cdot 2i \cdot \mathrm{Sin}(\omega t)(aq + bp)\}$$

Which is reduced to its final form

$$\xi(t) = e^{\mu t}\{W_1 \cdot \mathrm{Cos}(\omega t) + W_2 \cdot \mathrm{Sin}(\omega t)\}$$

Where

$$W_1 = 2(ap - bq) \qquad W_2 = -2(aq + bp)$$

If the arbitrary constants of integration a & b are set to equal a = b= -½ the weighting functions reduce to

$$W_1 = (q - p) \qquad W_2 = (q + p)$$

Note:- The solution of this expression for the characteristic equation quoted in Jennings[62], is incorrect. In his formulation the weighting functions are given as $W_1 = -(p - q)$ and $W_2 = (p - q)$, these solutions are unobtainable with two weighting functions.