UNIVERSITY
of
GLASGOW

DEPARTMENT of CIVIL ENGINEERING

Integrated Implementation System for Pseudodynamic Testing

by

William H. Algaard, M.Eng

Thesis submitted to the University of Glasgow
in candidacy for the degree of
Doctor of Philosophy

Glasgow
UNITED KINGDOM

November 2001

Til Eliane

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to several people who have made this research and completion of the thesis possible.

First of all I would like to thank my supervisors, Prof. Nenad Bićanić and Dr. Alan Agar for their encouragement, support, ideas and assistance throughout this Ph.D.

I would then like to thank the Faculty of Engineering for the scholarship that has enabled me to work towards this degree and Department of Civil Engineering for providing all the experimental and computational facilities I have required. I would also like to thank the technical staff for their work and assistance in the laboratory.

My thanks extend further to all my office mates who have suffered me for the last three years, Andrej, Domenico, Kourosh, Kevin, Joy, Gerrit, Alan and Stuart and all my other friends who have made my time in Glasgow so enjoyable, Anne, Benny, Boris, Chris P., Chris S., Colin, Craig, Dave, Hervé, Hua Feng, Lindsey, Marc, Nenad, Steve, Zu and others.

Last, and most of all, I would like to thank my parents for encouraging me to start the degree and Eliane for encouraging and assisting me to complete it.

ABSTRACT

The pseudodynamic test method is a tool for obtaining the non-linear response of structures to transient ground acceleration. The modelling technique relies on representing the inertial and viscous damping components of the equation of motion computationally, while obtaining a measure of the non-linear elastic restoring forces experimentally. A pseudodynamic implementation system is presented, displaying innovations within both the computational and experimental domains.

A SDOF pseudodynamic test facility has been designed and manufactured employing a computer controlled servo-hydraulic actuator system. The experimental facility enables displacements of up to ±50mm under forces of up to ±50kN with all required instrumentation. The experimental apparatus is controlled by algorithms running in the LabView environment, fully integrated within the execution system, rendering the requirement for a hardware controller obsolete. The execution system allows interactive control of the experiments, and offers a large range options with respect to both control and time integration. The execution routine incorporates both the time integration and control algorithms, and combines these such that they effectively execute as an integrated system. This enables semi-continuous implementation of the pseudodynamic tests with very limited resources.

A novel, integral form time stepping scheme is proposed, based on an explicit integral form algorithm (Chang *et al.* 1998) and the Newmark Implicit scheme. The proposed formulation offers an implicit, and thus unconditionally stable alternative to Chang's algorithm without introducing further approximations. This yields improved dissipation and accuracy properties in addition to enabling combination of the integral form schemes' advantages of representing non-linear force variations during a time step with an unlimited time step size. The improvements have been shown both through analytical analyses and numerical examples in linear and non-linear systems. Implementation of the implicit integral form algorithm has been enabled by coding parts of the algorithm directly into the digital controller.

The pseudodynamic implementation system has been thoroughly evaluated and verified, and sensitivities with respect to control and time integration assessed. Finally, the test system has been employed to carry out an experimental investigation of the effects of repeated exposure of reinforced concrete structural details to seismic ground acceleration. This examination revealed that structures exhibiting fundamental frequencies below the prevailing excitation frequency were superior in handling repeated exposure to structures with fundamental frequencies higher than the excitation frequency.

TABLE OF CONTENTS

a	Acceleration vector (scalar for SDOF system)
b	Breadth of RC section
c	Viscous damping constant for SDOF system
d	General displacement notation
d	Effective depth of RC section
d_n	Displacement vector (scalar for SDOF system) at start of time step
d_{n+1}	Displacement vector (scalar for SDOF system) at end of time step
d_m	Displacement vector (scalar for SDOF system) at start of sub-step
d_{m+1}	Displacement vector (scalar for SDOF system) at end of sub-step
f	Applied force vector (scalar for SDOF system)
f_{cu}	Concrete cube strength
f_y	Steel yield stress
$\bar{f}_{n+1}$	Time integral of excitation force at end of time step
k	Stiffness constant for SDOF system
k_o	Initial stiffness constant in SDOF system
m	Mass constant for SDOF system
r	General restoring force notation
r_n	Restoring force vector (scalar for SDOF system) at start of time step
r_{n+1}	Restoring force vector (scalar for SDOF system) at end of time step
r_m	Restoring force vector (scalar for SDOF system) at start of sub-step
r_{m+1}	Restoring force vector (scalar for SDOF system) at end of sub-step
$\bar{r}_{n+1}$	Time integral of restoring force at end of time step
s_n	Time integral of displacement at start of time step
s_{n+1}	Time integral of displacement at end of time step
s^{-1}	Radians per second
s_v	Spacing of links in RC member
t	General time notation
t_m	Equivalent time at start of sub-step
t_{m+1}	Equivalent time at end of sub-step
v	General velocity notation
$\tilde{v}$	Predictor velocity vector (scalar for SDOF system)
v_c	Permissible shear stress in RC section
w	Maximum displacement at beam end in Bernoulli-Navier thin beam theory
x	Length of beam in Bernoulli-Navier thin beam theory
x	General displacement notation
x_t	Displacement vector (scalar for SDOF system) at current time step
$x_{t-\Delta t}$	Displacement vector (scalar for SDOF system) at previous time step
$x_{t+\Delta t}$	Displacement vector (scalar for SDOF system) at next time step

$\tilde{x}$	Predictor displacement vector (scalar for SDOF system)
$\bar{y}$	Distance from neutral axis to centre of section bolted on for shear calculations
z	Distance to steel rebar from RC section neutral axis
z	Lever arm for strength calculation of RC section counting rebar only
A	Area of RC section
$[A]$	Recursive amplification matrix
AI	Implicit LabView expression for acceleration variable
A_s	Area of steel rebar in RC section
A_{sv}	Effective area of shear links in RC section calculations
C	Viscous damping matrix
C	Percentage of viscous damping
$Del\ T$	Time step size
E	Young's modulus
Flt	Pseudo force vector as LabView variable
I	Second moment of area
I_{xx}	Second moment of area around the x-axis (strong direction)
I_{yy}	Second moment of area around the y-axis (weak direction)
K	Stiffness matrix
K^I	Tangent or initial stiffness matrix
L	Overall length of section
M	Mass matrix
M	Moment capacity in RC section calculations
M_{cap}	Moment capacity in steel section calculations
P	Force at tip of cantilever for deflection calculations
P_1	Pressure in chamber 1 of actuator
P_2	Pressure in chamber 2 of actuator
Q	Shear force per unit length
RI	Implicit measure of restoring force as LabView variable
S	Maximum shear force
T_a	Observed period of vibration produced numerically
T_{min}	Minimum period of vibration in stability calculations
T_n	Exact period of vibration
VI	Implicit expression for velocity as LabView variable
V_c	Shear capacity of RC section
Vt	Explicit expression for velocity as LabView variable
XI	Implicit expression for displacement as LabView variable
X_n	State vector at start of time step in recursive analysis
X_{n+1}	State vector at end of time step in recursive analysis

Xt	Explicit expression for displacement as LabView variable
Z_{xx}	Elastic modulus of section

Greek letters:

α	Parameter in α-Operator Splitting algorithm, determines α-damping
β	Constant in Newmark based time stepping schemes (normally = ¼)
γ	Constant in Newmark based time stepping schemes (normally = ½)
δ	General deflection notation
λ	Eigenvalues of recursive amplification matrix
ξ^2	Expression for $\Omega^2/(1+\Omega^2)$
π	Pi, relationship between circle's radius and perimeter
$\rho(A)$	Spectral radius of $[A]$
ω	Circular velocity
ω_n	Natural frequency of dynamic system
Δ	General notation for change over time step
Δt	Time step size
Δt_{max}	Maximum time step size for stability
ΔT	Period error
Ω^2	Expression for $\Delta t^2 k/m$

Chapter I:

INTRODUCTION

1.1 STATEMENT OF PROBLEM

In order to ensure safe and economical design and construction of structures in earthquake prone regions, it is vital to fully understand the behaviour exhibited by the structures when subjected to strong ground motion. This behaviour can generally not be well described in terms of linear elastic behaviour at the ultimate or even the serviceability limit state. In a reinforced concrete structure, crack opening and closing, yielding and de-bonding of steel and increasing internal friction all influence the structural behaviour as soon as displacements grow beyond a certain threshold. Similarly, in a steel or timber structure, inelastic deformations within joints and connections and material yielding result in unrecoverable deformations and energy dissipation.

The actual ability of structures to resist earthquake loading can only rarely be observed directly. This is naturally due to the highly irregular spacing of significant earthquakes in both space and time. Additionally, whenever an earthquake strikes, the response of the structures can very seldom be properly monitored, and the behaviour is not well accounted. Only the consequences of an earthquake can be studied systematically, and even then the evaluation can be difficult to perform, as the exact original configuration of the structures is often unknown. It is not uncommon to observe two similar buildings next to each other following an earthquake where one suffers a complete destruction and the other apparently sustains negligible damage.

As it is so difficult to evaluate the performance of structures directly during real ground motion, various modelling techniques have been employed to attempt to obtain the desired results. These range from computational techniques, employing both linear elastic and non-linear finite element methods, to experimental techniques of various levels of sophistication. While computational methods have experienced significant advancement recently with an exponential growth in computational power, there are still difficulties in accurately modelling the inelastic deformation and local energy dissipation that takes place on an element level. On the other hand, simple experimental techniques, like cyclic quasistatic tests, are well suited for evaluating structural performance, e.g. reinforced concrete members and joints. However, these experimental techniques are unable to model the actual earthquake response. This

implies that they cannot directly determine the structural behaviour during strong ground motion. More sophisticated experimental procedures may have the ability to provide such information. These include the shaking-table testing technique, which can directly mimic the ground motion that takes place during an earthquake. With this technique, the structure concerned is constructed to full or, more commonly, to a reduced scale, and then placed upon the shaking-table. The table can then reproduce the ground motion, controlling a single to all 6 degrees of freedom. Major drawbacks of these tests are that the specimen structures generally have to be constructed on a reduced scale, that they are expensive and can only be carried out in highly specialised laboratories. Recently, a combined experimental-computational testing technique has been proposed and successfully employed to model inelastic behaviour of large-scale structures under seismic loading. This method, described as the *on-line* or *pseudodynamic* test method, aims to obtain a realistic non-linear response of structures at ultimate limit state without the use of the resource demanding shaking-table apparatus.

The main principle of the pseudodynamic test method is to split the components of the equation of motion governing the dynamic behaviour of a structure, equation (2.1), into computational and experimental components. This allows the linear and well defined terms to be represented computationally while the non-linear and unpredictable terms are obtained directly from an experimental model. The response in then obtained through direct integration of the equation of motion in a step-by-step procedure. In effect, the inertia and viscous damping of a structure are expressed computationally while the non-linear structural restoring forces are accounted for experimentally. This procedure allows the overall performance of a structure subjected to strong ground motion to be evaluated accurately as well as displaying detailed behaviour on the element level.

1.2 MAIN OBJECTIVES

The main objectives of this thesis are to offer a novel system for pseudodynamic implementation requiring fewer resources than traditional methods and to present a range of other potential improvements to this test method. This involves the

facilitation of a pseudodynamic test facility, including the development of both the experimental and computational components. The individual tasks can be sub-categorised into the following main areas:

- Development of an experimental (physical) test facility.
- Development of a computational control and execution system.
- Integration between the experimental and computational components by development of a communications system.
- Research novel time stepping schemes and work on the integration of these into a pseudodynamic framework.
- Development of a novel time stepping scheme specifically for use in fast, continuous pseudodynamic implementation.
- Verification and evaluation of the implementation system.
- Application of the testing facility on realistic structural components.

Completion of the above tasks was realised through the development and application of a closely integrated control-time stepping-execution system, existing entirely on a software level combined with the hardware communication through a single high-speed card. A fully implicit time stepping algorithm is partly coded into the controller to enable an integral form representation.

1.3 THESIS ORGANISATION

A review of the historical development and the current status of the pseudodynamic test method is provided in Chapter II. This puts the work documented in this thesis in perspective and provides a background for the test method in general. It contains discussions on the implementation requirements, advantages and limitations of the pseudodynamic test method and typical areas of application. Chapter II also contains a theoretical discussion on a range of both explicit and implicit time stepping schemes.

The experimental set-up is documented in Chapter III. This includes the design of the reaction rig, the hydraulic system, the test specimens and all the various connections required to join the experimental components together. The communications system and instrumentation is also detailed in this chapter along with a short discussion on the mechanics of the pseudodynamic test method.

In Chapter IV, the development of a novel time stepping scheme is documented. This scheme, denoted the Newmark Implicit – Integral Form algorithm, is an unconditionally stable time stepping scheme that can be applied with non-linear pseudodynamic systems without iterations. The advantages of the integral form representation are detailed and the improvement from the explicit version of the scheme (Chang *et al.* 1998) explained. A full theoretical analysis of the stability and accuracy properties of the two algorithms is included as well as the practical implementation system. Lastly, the performance of the algorithms under non-linear conditions is discussed with respect to the abilities in handling rapidly varying external forces and non-linearities in the stiffness of the system.

Chapter V documents the theoretical development of the controller. This includes a discussion on the theoretical control modes and the actual interaction between them in the control loop. Some additional objectives for the controller are also described as well as an evaluation of the performance. The actual coding and operation of the controller is described in Chapter VI, which documents all the coding in the LabView environment. Chapter VI introduces the graphical programming environment LabView and describes briefly some of its main operating principles. Then, the coding of the pseudodynamic control and execution system in the same environment is documented in detail. In addition to the controller running as a sub-loop in this system, the overall control and time integration is also coded within the same main program and detailed here. The chapter also includes a discussion on the actual operation of the system and an evaluation.

In Chapter VII, verification tests and sensitivity studies are documented. The verification is done both in terms of numerical and so-called snap back tests, while the sensitivity study aims to investigate the sensitivity of the implementation system to a

range of factors. Essentially, the sensitivity study obtains relationships between *time step size*, *accuracy*, *speed of implementation* and *response*.

An application of the pseudodynamic testing system is subsequently carried out in Chapter VIII. Here, the consequence of repeated earthquake loading on the same specimen is investigated to determine the changes following initial damage. The scenario is interesting as both the earthquake demand and structural capacity may change. Overall conclusions and recommendations are included in the final Chapter IX.

Chapter II:

REVIEW OF THE PSEUDODYNAMIC TEST METHOD

Dynamic testing can be carried out by two classes of methods, computationally – through for example dynamic finite element analyses, or experimentally, typically through shaking table testing. However, both methods display limitations. Despite their sophistication, computational models still have only a limited ability to recreate the realistic non-linear behaviour of damaged reinforced concrete, while for example shaking table tests require highly specialised laboratories and normally need to be carried out on a reduced scale. Pseudodynamic testing, on the other hand, is a part computational-part experimental testing technique for structures undergoing dynamic loading. The motivation behind the method is to enable simulation of dynamic systems beyond their strength limit without the complexity and costs of shaking tables, while retaining the more realistic non-linear behaviour obtained through experimental tests. The method facilitates modelling of such dynamic systems by representing inertial and viscous damping forces computationally, while measuring the non-linear restoring forces experimentally. Combined, these forces form the equation of motion, which computes a step-by-step response of a system to an excitation.

2.1 INTRODUCTION

According to Takanashi & Nakashima (1987), Hakuno, Shidawara and Hara formulated the initial concept of pseudodynamic testing in Japan in 1969. However, Takanashi & Nakashima were the first to succeed in obtaining a satisfactory system response in 1975, and they called the method the "computer-actuator on-line system". By the mid 80's, pseudodynamic testing was being carried out on a significant scale in Japan, e.g. (Takanashi & Nakashima 1987), (Yamazaki *et al.* 1986) and in the USA, e.g. (Shing & Mahin 1986), (Thewalt *et al.* 1986), (Beck & Jayakumar 1986), (Mahin *et al.* 1989), (Aktan & Hashish 1986), (Mahin & Shing 1985). Later on, new centres developed, including Italy, e.g. (Combescure & Pegon 1997), (Negro *et al.* 1996), (Pegon & Pinto 2000), Taiwan (Chang *et al.* 1998), Korea (Chung *et al.* 1999) and the UK, e.g. (Darby *et al.* 1999), (Williams *et al.* 1999), as well as a number of smaller laboratories.

Today, most research groups working on pseudodynamics concentrat their efforts on particular aspects of the technique. For example, these include real-time/high-speed implementation (Darby *et al.* 1999), the so-called substructuring techniques (Pegon & Pinto 2000), (Horiuchi *et al.* 1999) and combined real-time substructuring tests (Williams *et al.* 1999). Following the research carried out over the last 25 years, and in particular numerous comparisons with shaking table tests (Yamazaki *et al.* 1986), (Chung *et al.* 1999), the pseudodynamic method is now generally recognised as a reliable method for testing at least some types of dynamic systems.

The concept of the pseudodynamic test method builds on the fact that inertial forces acting on a structure during motion may be correctly represented numerically as these are simply linear functions of *mass* and *acceleration*. The restoring forces created when a structure undergoes large amplitude oscillations, for example during seismic excitation, are on the other hand still too complex to account for with numerical models alone. These need therefore be represented experimentally. Combining forces from these two different sources also offers a clear advantage over simple quasistatic tests; although such tests may determine the structural response to a given displacement, they cannot predict the response of a dynamic system to a given ground motion.

In order to test a structure pseudodynamically, the dynamic system is represented in terms of a finite number of discrete springs, masses and dampers. The equilibrium equation governing the motion of a body exposed to inertial, damping and non-linear restoring forces can be expressed as:

$$Ma + Cv + r(x) = f \tag{2.1}$$

where M and C are the mass and viscous damping matrices and a, v, r, x and f the acceleration, velocity, restoring force, displacement and applied force vectors respectively. The response is obtained by discretising time and calculating the displacements in a step-by-step manner. A time-stepping algorithm computes displacement steps, based on the acceleration, velocity, applied force and restoring force at the start and/or the end of each time step. These displacement steps are

subsequently imposed on the structure by means of computer controlled servo-hydraulic actuators, each controlling a separate degree of freedom. Once the structure has been deformed, the resulting restoring forces are measured and used for further calculations.

2.2 IMPLEMENTATION

The implementation of a pseudodynamic test is concerned with how the conceptual idea of using an experimentally measured restoring force and numerically expressed inertia can generate the dynamic response of a structure. It is known in general terms that a time integration algorithm computes a displacement step based on the equation of motion, equation (2.1), which is subsequently imposed on the structure by hydraulic actuators. The force required to do this will equal the restoring force created within the structure, and this value is fed back into equation (2.1) to allow further computation steps to be carried out. The data flow is thus as indicated within figure 2.1.

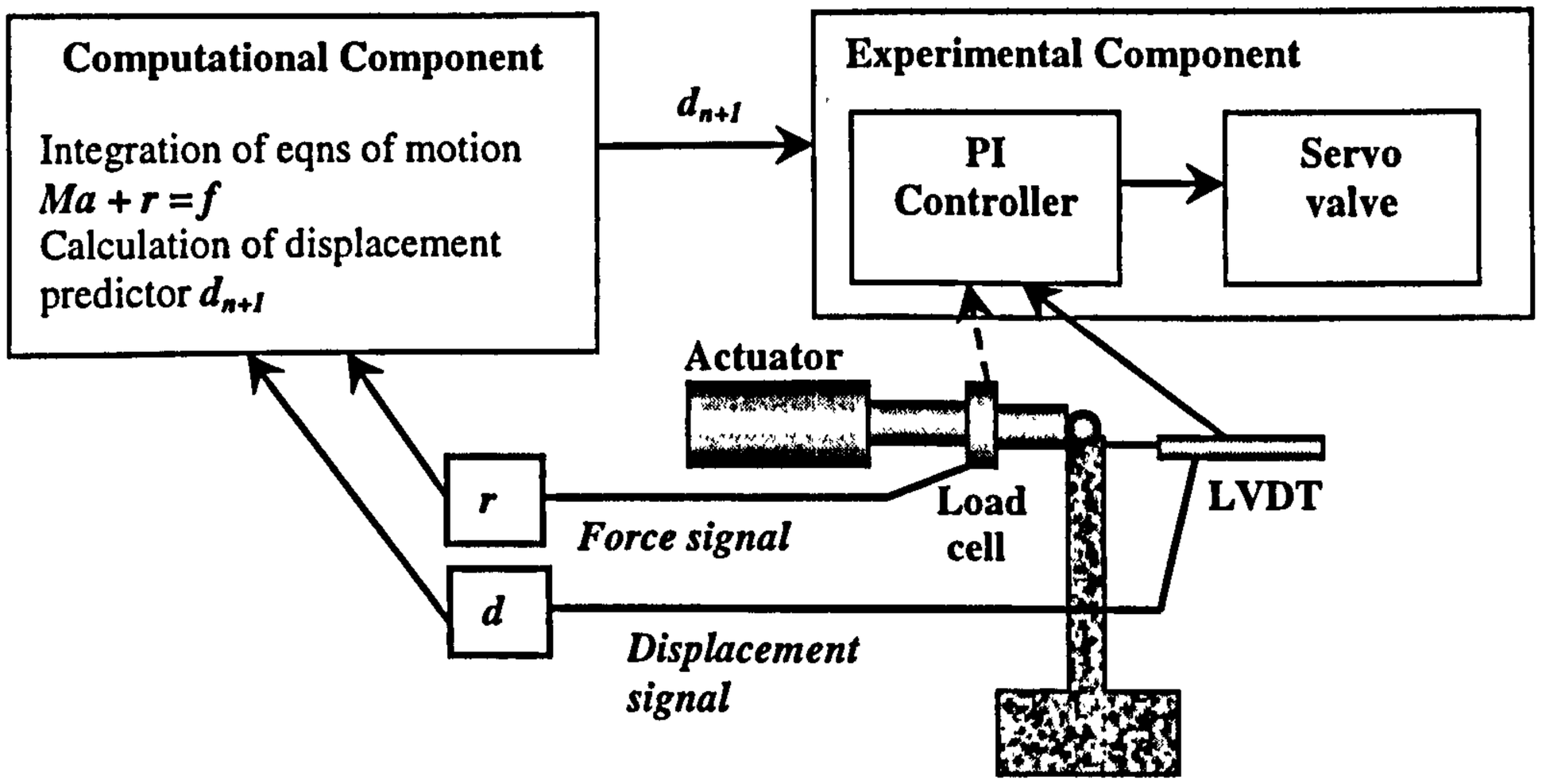


Fig. 2.1 Conceptual data flow in pseudodynamic tests

Although the pseudodynamic test method in principle proceeds in a step-wise manner, it may also be implemented more or less continuously. The classical implementation of the method involved a "hold-period" of the order of 1 second after the displacement step was imposed to allow force measurements to take place. This

clearly left room for considerable force relaxation to take place, potentially corrupting the response.

The idea of performing pseudodynamic tests in a continuous manner was suggested by Thewalt & Mahin (1994), but not implemented at this stage. At ELSA, JRC-Ispra, Italy, a continuous system using a sub-stepping technique has successfully been implemented (Magonette *et al.* 1998) and has since become a standard method. Most recently, attempts at implementing tests in real time have been made, e.g. (Horiuchi *et al.* 1999), (Magonette *et al.* 2000), (Williams *et al.* 1999), (Nakashima & Masaoka 1999). This would avoid any discrepancies resulting from differential rate of loading.

The implementation of the pseudodynamic technique involves experimental instrumentation, hardware communication, control, time integration and execution routines. While numerous publications exist on time integration and algorithmic details within the test method, very few elaborate on the implementation on the whole, including execution, control and communication systems. In terms of electronic instrumentation, the general requirements comprise one load cell and one LVDT (displacement transducer) for every degree of freedom concerned. In addition, the actuator servo valves need to be connected to controllers.

At ELSA, JRC-Ispra, Italy, an extremely accurate, fully digital instrumentation and control system is utilised (Negro 1997). Optical LVDTs deliver digital, noise free, displacement signals to an accuracy of 2μm, which can be read directly by the digital controller unit. This unit consists of 80486-based computer and a discrete PID hardware controller, also digital. The valve signals are converted through a digital-to-analogue converter. Force, on the other hand, is measured with conventional load cells, but these signals go through an analogue-to-digital converter before entering the system. Time integration is however carried out remotely on the main computer, and communication between the main computer and the controller unit is carried out over a Local Area Network (LAN). The test on the whole is controlled and executed on the main computer by a program called MiNi-PDTM, in which the time integration is incorporated (Buchet & Pegon 1994).

A different pseudodynamic implementation system is described by Vannan (1991) at the University of Colorado. In this case, analogue instrumentation has been opted for, including a system of dual displacement control, and hardware controller units. These, and the other instrumentation, communicate with the execution routine through a system based on so-called MTS-Software Toolkits. The execution routines incorporating time integration, as well as the communication toolkits, are programmed in Pascal, but Fortran 77 has also been utilised when including substructuring capabilities.

Recently, Cuadra & Ogawa (2001) from the Akita Prefectural University, Japan presented a system for pseudodynamic implementation using conventional testing devices. This system eludes the requirement for actuators and uses simple hydraulic jacks instead. Rather than being controlled by servovalves, the oil flow to the jacks is controlled by inverter motors and high-speed on-off valves. This implies that loading and unloading are essentially carried out using two different systems. Still, satisfactory accuracy and system responses can be obtained, however under low strain rates.

2.3 ADVANTAGES

The pseudodynamic test method offers a range of advantages to alternative techniques. As already indicated, numerical techniques cannot, in general, sufficiently well represent the complex stiffness degradation that takes place during severe seismic excitation of structures.

The most direct test method, the shaking table method, offers probably the most realistic option to model non-linear dynamic behaviour of structures. With this method, the structure to be evaluated is built to either full or a reduced scale, and placed upon the test floor. This floor can reproduce the ground motion that takes place during an earthquake in terms of acceleration, velocity and displacement. To achieve this, powerful hydraulic actuators are connected to the floor, and control this with any number between one and six degrees of freedom. Most shaking tables do however permit only a quite severely limited size of test structure (Chung *et al.* 1999). This can

be in terms of dimensions, weight or stiffness. For this reason, shaking table tests are usually carried out on a reduced scale. In itself, this reduces the value and reliability of the test, but also raises a range of questions related to the size effects and concerned with satisfying the similitude conditions. For example, when scaling mass in a shaking table test, time also has to be scaled accordingly, necessitating higher accelerations and velocities to be provided by the test floor (Kumar *et al.* 1997). These factors are normally fairly restricted and are functions of the payload on the floor. Additionally, shaking table tests need to be carried out in real time to capture the dynamic behaviour. This not only requires substantial actuator capacity, but also generally makes measuring difficult.

Pseudodynamic tests have historically been carried out on an expanded time scale. As dynamic effects only exist computationally within the pseudodynamic context, the structure may be strained at any desired rate, and tests can even be stopped and restarted (Negro 1997). This reduces the required actuator capacity and enables use of conventional instrumentation for measurements (Yamazaki *et al.* 1986). In fact, pseudodynamic tests can largely be carried out with classic quasi-static test laboratory equipment. Testing of large, full-scale structures is also significantly easier as the maximum size or weight is not limited by the capacity of the shaking table.

Among the largest pseudodynamic tests carried out are tests on a 6 storey building in Japan (Takanashi & Nakashima 1987), and a 4 story building in Italy (Negro *et al.* 1996), both to full scale. Furthermore, as the primary dynamic effect of the mass in the structure is modelled computationally, this needs in fact often not be included in the experimental set-up, potentially reducing construction costs of the specimen.

2.4 LIMITATIONS

Although the pseudodynamic test method offers a number of advantages to other dynamic testing techniques, it does also display some inherent limitations. These are principally related to the following points: *spatial* and *temporal discretisation*, *number of degrees of freedom (DOF)*, *strain rate effects* and finally *experimental*

errors. These points will be discussed briefly in order to assess their effect and appreciate their importance.

2.4.1 Spatial discretisation

The pseudodynamic test method cannot directly model a continuum structure. This is due to fact that only distinct degrees of freedom can be controlled, i.e. points where hydraulic actuators are connected. The method relies on the possibility of representing continuous structures as discretised systems of stiffnesses and masses. For certain types of structures, this approximation appears fair. These include frame structures where the mass is largely concentrated in the floors and the flexibility (stiffness) in the columns. For other structures however, such idealisation is not as evident. Nevertheless, even structures with no clear concentration of mass can be represented sufficiently well with a very limited number of DOFs, if suitable condensation techniques are employed (Negro 1997).

A reasonable number of degrees of freedom is limited both by the experimental facilities and the test structure. Increasing numbers of actuators greatly complicates control and implementation overall, and requires more sophisticated execution systems. One is also dependent on being able to connect the actuators to strong, rigid points on the structure to avoid unrealistic local effects. In a frame structure, the column-floor connection offers excellent such properties.

2.4.2 Temporal discretisation

Temporal discretisation is concerned with the way the pseudodynamic tests progress in a step-wise manner. As clearly no analytical method exists to obtain the response of a highly non-linear structure subjected to a transient ground motion, a numerical solution technique is required. These rely on the linearisation over a finite time step, i.e. on breaking the time into steps over which variables are assumed to vary linearly.

In pseudodynamics, the time stepping algorithm normally operates with a constant time step size, although some research has been carried out using adaptive steps (Bursi *et al.* 1994). However, this was done primarily to ensure stability of Newton-

type iterations in severely strain softening systems, and as such iterations are generally not desired (Shing & Vannan 1990), the technique is of limited use.

The maximum size of a time step is limited by two factors, one required for stability and the other required for accuracy. There is a general consensus that to enable a reasonably accurate response, around 10 steps per vibration cycle of interest are required. Stability for explicit time integration schemes limits the maximum time step size, Δt_{max}, to T_{min}/π, where T_{min} is the shortest eigenperiod of the structure. In the single degree of freedom (SDOF) case, the stability limitation thus clearly becomes irrelevant for practical purposes, as the accuracy argument prevails.

For multiple degree of freedom (MDOF) systems however, the situation may be different. In such systems, typically only the first few modes will be contributing noticeably to the overall response. Therefore, the accuracy of only these modes will be important. Stability considerations now conversely become more important. As a number of higher vibration modes exist, the T_{min}/π relationship requires that Δt_{max} for explicit methods is now limited by the highest frequency of the system. This can easily restrict Δt_{max} below that required for accuracy. A small time step is not a problem in itself, and ensures minor linearisation error during each time step. It may however become impractical to utilise too small time steps. Depending on the quality of implementation, a certain control error will exist for every time step. Smaller time steps clearly imply more steps are required to obtain the response, and this increases the scope for overall error propagation. Additionally, reduction in the time step size may increase the implementation time (Algaard *et al.* 2001a).

2.4.3 Strain rate effects

Numerous questions have been raised regarding the reduced strain rates normally existing in pseudodynamic tests. Historically, structures are strained at a much slower rate than during real earthquakes. The rates are typically reduced by 1 to 3 orders of magnitude, something that could potentially affect the stiffness properties. Classic pseudodynamic tests are further implemented in a step-wise manner, in which a hold period is included where the actuator is stopped for force measurements.

Conflicting opinions exist on whether the rate of loading developed during earthquakes is fast enough to significantly affect the change in stiffness. While Shing & Mahin (1988) claim the dynamic yield strength of steel structures may be up to 30% higher during a seismic response than during pseudodynamic tests, Yamazaki *et al.* (1986) found the effect of the loading rate on the response to be "insignificant" when comparing with shaking table tests. It is however undisputed that earthquake isolation and damping devices display highly rate dependent restoring force properties. In order to realistically test these, it is imperative that tests progress in as close as possible to real time.

Currently, at lot of research is being carried out to enable and optimise real time pseudodynamic testing, e.g. (Williams *et al.* 1999), (Horiuchi *et al.* 1999), (Nakashima & Masaoka 1999), (Magonette *et al.* 1998), (Magonette *et al.* 2000). Facilities now exist where pseudodynamic tests, also with substructuring, can be carried out in real time, enabling for example testing of base isolated structures.

2.4.4 Experimental errors

Pseudodynamic tests are particularly sensitive to experimental errors as the effects of such may be carried over from step to step and allowed to accumulate. It is therefore essential that these are well monitored and controlled. A range of studies have been carried out to evaluate the propagation effects of experimental errors, e.g. (Beck & Jayakumar 1986), (Shing & Mahin 1986), (Thewalt & Roman 1994), (Shing & Mahin 1990), (Peek & Yi 1990a) and (Peek & Yi 1990b).

Errors can originate from various sources within pseudodynamic tests. They can be simple measurement errors or stem from control inaccuracies. Some might be noise related and random, while others are systematic. It is generally concluded that random errors are of much less significance than systematic ones. As the response of structures can be described as oscillatory, it is clear that the effect of a single, or random error cannot accumulate. Some systematic errors are however of a nature which can result in this. These are particularly concerned with control errors, as measurement errors can largely be avoided through well-calibrated, high quality instrumentation.

Control errors result from difficulties in controlling actuator motion and are often systematic in nature. The most usual errors are related to consistent *overshoot* and *undershoot* errors resulting from the actuator moving too far or too short during a displacement step. These errors are usually in phase with the velocity of the structure, resulting in energy dissipation or addition, respectively. Overshot displacement increments will result in increased restoring forces during loading, but will be associated with decreased restoring forces during unloading as the structure is in a more (or less) displaced position. This will clearly have a damping effect on the system, as there will be an "inflated" force acting against the velocity when moving away from the neutral position, while when moving toward the neutral position, there will be a "deflated" force working with the velocity. With undershot displacements, the situation will of course be exactly opposite, and energy will be added during every oscillation.

Pseudodynamic systems are particularly sensitive to the errors mentioned above. The cumulative error growth has in some cases in fact been found to dominate the response (Negro 1997). In particular for MDOF structures, the frequency of the experimental errors may be such that high, otherwise structurally insignificant, modes may be exited, resulting in entirely unreliable results.

Overshoot and undershoot errors may additionally be interpreted in an alternative way. Rather than referring to the actuator moving too far or too short in each step, it may be understood as the total (not incremental) displacement being too high or too low. During loading, the meaning will be identical to the earlier definition, but during unloading on the other hand, the meaning will be opposite. Overshoot now still refers to the actual displacements being too high and undershoot the displacements being too low. The errors are thus no longer in phase with the velocity, but in phase with displacement. This implies that with overshoot, the restoring force is inflated during both loading and unloading, and with undershoot deflated during both. The net energy change due to these errors in a linear system is zero, although amplitudes will be affected. In non-linear systems, the likely effect of such overshoot under strain softening conditions is dissipation, as inflated displacements during loading result in additional damage and softening of the structure before velocity reversal. With undershoot; the effect should be better energy conservation rather than energy

addition, as slightly less damage and hysteretic damping will take place. In any case though, the effect of this type of overshoot and undershoot errors, possibly stemming from poor calibration of LVDT's, is considerably less severe than the one described previously.

2.5 TIME INTEGRATION

The objective of the time stepping algorithm in a pseudodynamic test is to solve the equation of motion, equation (2.1) in a step-wise manner in order to build up the full system response through repeated generation of displacement steps. They generally require a measure of the acceleration and velocity present as well as the forces acting at the start and/or end of the time step. In effect, the time stepping algorithms produce an extrapolation, or update, of each of the variables describing the state of the dynamic structure: displacement, velocity and acceleration.

Time integration algorithms are described as being either *explicit* or *implicit*. This refers to whether they solve the equilibrium equation at the start or end of the time steps. In the context of pseudodynamics, explicit methods provide a solution using only information available at the start of the time step, while implicit methods require an assessment of the forces acting at the end.

Explicit schemes imply equilibrium conditions at the current time t, hence they utilise measures of acceleration, velocity and total force at the start of the time step, in some cases in addition to values during previous steps, to effectively extrapolate the effect to the end of the step, resulting in a final displacement. Implicit methods, on the other hand, deal with equilibrium at time $t+\Delta t$, hence they require the total force at the end of the step in order to calculate the final acceleration. The implicit methods therefore rather interpolate the changes during the step to obtain the final displacement.

The explicit or implicit nature of an algorithm can severely affect the stability properties. Explicit methods are only conditionally stable, i.e. there is a maximum size of time step that may be used. Breach of stability leads to unbounded error growth.

Unconditional stability may be obtained with implicit methods on linear systems, and often also for a non-linear system. This implies that there exists no limitation on the maximum time step size that may be employed.

2.5.1 Explicit schemes

Explicit time integration schemes were the first to be used within pseudodynamic testing. This was clearly because of their ease of implementation and inherent ability to handle non-linear systems. The ability to handle non-linearity, in particular, is important as all pseudodynamic tests deal in effect with non-linear stiffness properties, as the restoring forces for given displacements are measured. The drawback of explicit schemes is that they are as known only conditionally stable, which puts a limitation on the maximum time step size required for stability purposes. Normally, this limitation will be a function of the natural frequency of the system, defined as $\omega_n \Delta t_{max}=2$, where ω_n is the highest natural frequency and Δt_{max} is the maximum allowable time step duration, as discussed in section 2.4.2. The two most commonly adopted explicit time integration algorithms are the central difference method and the Newmark Explicit method.

2.5.1.1 The central difference method

The central difference method has been widely used in pseudodynamic testing, e.g. (Takanashi & Nakashima 1987), (Mahin & Shing 1985), (Shing & Mahin 1986), (Shing & Mahin 1990), (Peek & Yi 1990a), as it requires neither a measure of the stiffness matrix, nor the restoring force at the end of the step, and is simple to implement. It relies on an approximation of the acceleration acting at the centre of two time steps defined as (Wilson & Bathe 1976):

$$a_t = \frac{1}{\Delta t^2}\left(x_{t-\Delta t} - 2x_t + x_{t+\Delta t}\right) \tag{2.2}$$

where a is the acceleration, x the displacement and Δt the duration of a time step. This approximation is second order accurate. A similar expression can be formed for the velocity, yielding

$$v_t = \frac{1}{2\Delta t}\left(-x_{t-\Delta t} + x_{t+\Delta t}\right) \tag{2.3}$$

where v is the velocity. Now considering the equilibrium equation, equation (2.1) at time t, equation (2.4) is formed.

$$Ma_t + Cv_t + r(x_t) = f_t \tag{2.4}$$

Substituting equations (2.2) and (2.3) into the equation (2.4) yields an expression that can be solved for $x_{t+\Delta t}$, equation (2.5).

$$x_{t+\Delta t} = \left[\frac{M}{\Delta t^2} + \frac{C}{2t}\right]^{-1}\left[f_t - r(x_t) - \frac{M}{\Delta t^2}(x_{t-\Delta t} - 2x_t) + \frac{C}{2t}(x_{t-\Delta t})\right] \tag{2.5}$$

This expression clearly does not contain any variables at time $t+\Delta t$, and can therefore in principle be solved by using information known at the start of the time step. It does however contain entries from previous time steps (i.e. $x_{t-\Delta t}$), and therefore requires a special start-up procedure (Mahin & Shing 1985).

The central difference method is energy stable, and thus introducing no amplitude error. It does however exhibit a noticeable periodicity error. This error, a period shortening, is a function of the natural frequency of the system and the size of the time step. Unless the stability limit is being approached, the periodicity error can be defined as (Géradin & Rixen 1994):

$$\frac{\Delta T}{T} = -\frac{\omega^2 \Delta t^2}{24} \tag{2.6}$$

where ΔT is the change in period and T the period. For a relatively large time step, say $\Delta t = T/10$, the period error will be of the order of 1.6%

2.5.1.2 The Newmark explicit method

The Newmark explicit method is derived from the general Newmark direct time integration method. This is based on a Taylor series expansion of the state vector, describing displacement, velocity and acceleration, at time $t+\Delta t$ in terms of the state at time t (Géradin & Rixen 1994). The method provides expressions for displacement and velocity at time $t+\Delta t$, as seen in equations (2.7) and (2.8) respectively.

$$x_{t+\Delta t} = x_t + \Delta t v_t + \Delta t^2 \left(\frac{1}{2} - \beta \right) a_t + \Delta t^2 \beta a_{t+\Delta t} \tag{2.7}$$

$$v_{t+\Delta t} = v_t + (1-\gamma)\Delta t a_t + \gamma \Delta t a_{t+\Delta t} \tag{2.8}$$

where the constants β and γ are related to the numerical integration of remainders in the expansion. Effectively, they apply the weighting to the acceleration at the start and end of the time step. It is immediately obvious that unless β is zero, the method would require a measure of the acceleration at the end of the time step and thus be implicit as the acceleration only becomes available once the equilibrium equation has been solved. For the displacement expression, the Newmark explicit method therefore places the entire weighting on the acceleration at the start of the step, and in effect assumes this acceleration remains constant during the step. The displacement predictor thus reduces to:

$$x_{t+\Delta t} = x_t + \Delta t v_t + \frac{\Delta t^2}{2} a_t \tag{2.9}$$

which is explicit as it only requires the acceleration at time t to be computed from:

$$a_t = M^{-1}\left[f_t - r(x_t) - C v_t \right] \tag{2.10}$$

When implemented in pseudodynamics, $v_{t+\Delta t}$ is not required until the step had been completed. This may therefore be computed when the displacement step has been imposed, the restoring force measured and the new acceleration found through calculation of the equilibrium equation at time $t+\Delta t$.

As the Newmark methods do not require any start-up procedure, they are somewhat easier implemented than the central difference method. Apart from this however, the central difference and Newmark explicit methods are mathematically equivalent (Mahin & Shing 1985) and have identical numerical properties (Shing & Mahin 1986). The Newmark explicit method is therefore often preferred.

2.5.2 Implicit schemes

As pseudodynamic tests expanded to include more degrees of freedom, the limitations of the explicit schemes became increasingly apparent. The conditional stability exhibited by explicit schemes normally limits the time step size to $2/\omega_n$, where ω_n is the highest frequency present in the structure. Even with just a few degrees of freedom, high frequency modes may exist that only negligibly contribute to the overall response. If the stability limit is breached, these will however grow without bound and the response obtained will rapidly lose any value.

Pseudodynamic tests with substructuring (section 2.6.1) are particularly reliant on implicit time integration schemes. The computational model typically contains a substantial number of degrees of freedom, thus creating a large number of frequencies, of which some may be very high. Employing small enough time steps to ensure the stability limit is not breached may introduce some problems. It may increase the total duration of the test, as a certain time allowance per step may be required. More importantly, it increases the severity of error propagation problems (Thewalt & Mahin 1994), and very small steps can be problematic to implement.

2.5.2.1 The Newmark implicit method

The best know implicit time integration method is probably the Newmark method applied with γ and β values set such that unconditional stability is achieved. When considering equation (2.1) solved for acceleration at time $t+\Delta t$ and equations 2.7 and 2.8, the method can be summarised in equations 2.11.

$$
\begin{aligned}
x_{t+\Delta t} &= x_t + \Delta t v_t + \Delta t^2 \left(\frac{1}{2} - \beta\right) a_t + \Delta t^2 \beta a_{t+\Delta t} \\
v_{t+\Delta t} &= v_t + (1-\gamma)\Delta t a_t + \gamma \Delta t a_{t+\Delta t} \\
a_{t+\Delta t} &= M^{-1}\left[f_{t+\Delta t} - C v_{t+\Delta t} - r(x_{t+\Delta t})\right]
\end{aligned}
\tag{2.11}
$$

Stability is ensured when $\gamma \geq ½$ and $\beta \geq ¼(\gamma+½)^2$ (Géradin & Rixen 1994). The method is most commonly applied using constant average acceleration, which is achieved by setting γ to ½ and β to ¼.

Implementation is carried out by substituting the expressions for $x_{t+\Delta t}$ and $v_{t+\Delta t}$ into the expression for $a_{t+\Delta t}$ in equation 2.11 above. This leaves only $a_{t+\Delta t}$ and $r(x_{t+\Delta t})$ as unknowns, as all the other variables are known from the previous time step. The latter may introduce some problems, but the equation may be solved if $r(x_{t+\Delta t})$ can be expressed as a function of $x_{t+\Delta t}$ (i.e. $r_{t+\Delta t}=kx_{t+\Delta t}$ in linear systems) or through an iterative procedure (in non-linear systems). However, when applying the method to pseudodynamic tests, the fundamental idea is that the restoring force should be experimentally measured to obtain the most realistic results. The method is therefore not applicable as it stands.

2.5.2.2 Iterative approaches

Implicit time integration schemes may be implemented and solved directly in linear systems. This is however not possible with non-linear systems as the restoring force at the end of the time step is required. In non-linear numerical analyses, implicit methods may still be utilised. Solution is enabled by employing iterative procedures such as Newton, or modified Newton, to effectively determine the tangent stiffness in that step. In pseudodynamics on the other hand, iterations are undesirable (Shing & Vannan 1990). This is because the tests are sensitive not only to the final displacement imposed by the actuator, but also on the path taken to reach it.

If larger displacements are imposed prior to reaching equilibrium, further damage may have taken place in the structure, permanently changing the stiffness properties (Shing *et al.* 1991). Even if the final displacement is not exceeded, partial unloading during the iterations may also affect force readings to a critical extent. Implicit time

integration in pseudodynamic tests relying on iterations is therefore only rarely attempted (Bursi *et al.* 1994). Still, implicit schemes are commonly used in pseudodynamics. This is possible through some relatively recent techniques. Shing *et al.* (1991) present an implicit scheme that relies on numerical iterations. This should in principle avoid displacement overshoot and unloading problems, but it is only exact if the tangent stiffness is known at any time. As this is usually not the case, overshoot and unloading will not necessarily be avoided in MDOF structures. Additional problems may arise if stiffening takes place in the tested structure, something that cannot always be ruled out.

2.5.2.3 Hybrid methods

A functioning implicit implementation scheme has been proposed by Thewalt & Mahin (1994). This intriguing idea builds on a part digital – part analogue solution scheme where a summing amplifier alters the signal sent to the servo-controller according to restoring force signals obtained during the step. Although the method has been proven to enable implicit implementation, it has not become widely used. The majority of the current implicit time integration schemes applied in pseudodynamics rely on some form of an estimate of the restoring force at the end of the step.

2.5.2.4 The α-Operator Splitting methods

The α-Operator Splitting technique presented by Combescure & Pegon (1997) relies on a predictor-corrector procedure built on the α method (Shing *et al.* 1991). It requires a definition of the restoring force contributed partly by the explicit expression available and partly by the force created from a function of an assumed stiffness and the implicit displacement corrector. The assumed stiffness is normally taken as the initial stiffness of the structure, as an accurate measure of the tangent stiffness is difficult to obtain. The fundamental equations behind this method are firstly:

$$\begin{aligned} \tilde{x}_{t+\Delta t} &= x_t + \Delta t v_t + \frac{\Delta t^2}{2}(1-2\beta)a_t \\ \tilde{v}_{t+\Delta t} &= v_t + (1-\gamma)\Delta t a_t \end{aligned} \tag{2.12}$$

where $\tilde{x}$ and $\tilde{y}$ form the explicit *predictor* step and γ and β are now defined as $\beta = (1\text{-}\alpha)^2/4$ and $\gamma = (1\text{-}2\alpha)/2$, where α determines the level of α shifting, tuning the numerical damping, and ranges from –1/3 to 0. If α equals zero, the method reduces to the standard Newmark Implicit (constant average acceleration) method. Secondly, the *corrector* step is formed as follows:

$$\begin{aligned} x_{t+\Delta t} &= \tilde{x}_t + \Delta t^2 \beta a_{t+\Delta t} \\ v_{t+\Delta t} &= \tilde{v}_t + \Delta t \gamma a_{t+\Delta t} \end{aligned} \tag{2.13}$$

where x and v form implicit expressions for the corrector of the displacement and velocity respectively. Furthermore, instead of using equation (2.1), an 'α-shifted' equilibrium position is defined, as seen in equation (2.14) below.

$$Ma_{t+\Delta t} + (1+\alpha)Cv_{t+\Delta t} - \alpha Cv_t + (1+\alpha)r(x_{t+\Delta t}) - \alpha r(x_t) = (1+\alpha)f_{t+\Delta t} - \alpha f_t \tag{2.14}$$

For the method to be implemented pseudodynamically without iterations, the expression for $x_{t+\Delta t}$ has to be evaluated at the start of each step. As with the standard Newmark implicit method this is not immediately possible as the acceleration forms an implicit term that is not available before the equilibrium equation has been solved. This of course contains the restoring force term, which is a function of the displacement. Solution is enabled through the so-called *operator splitting* method. This is based on an explicit approximation of the restoring force term as follows:

$$r(x_{t+\Delta t}) \approx r(\tilde{x}_{t+\Delta t}) + K^I (x_{t+\Delta t} - \tilde{x}_{t+\Delta t}) \tag{2.15}$$

where K^I ideally represents the tangent stiffness matrix, but lacking this, a representation of the initial stiffness is used instead. The explicit part of the restoring force is thus exact, while only the implicit part has to be estimated. If the tangent stiffness is available, this estimation will in principle be exact. For SDOF and some MDOF structures, it is theoretically possible to obtain a measure of the tangent stiffness. However, this requires particularly accurate force and displacement measurements, and is generally not a viable option.

The error involved in simply using the *initial stiffness* of the system is not as large as immediately imagined, as it enters the equilibrium equation only as a second order term (Chang *et al.* 1998). In the case of a SDOF structure, the initial stiffness may be simply measured experimentally using the pseudodynamic test set-up. As long as the chosen magnitude of the scalar representing the stiffness at any point in time is equal to or higher than the tangent stiffness, the method is, according to Combescure & Pegon (1997) unconditionally stable. However, it should also be ensured that the stiffness during unloading is not higher than that present during loading, as this would represent an energy-adding scenario that would lead to instability. In principle, as long as the assumed stiffness during loading and unloading are the same, the method should be energy stable. In the MDOF case, the situation will be largely similar. However, to form the *initial stiffness matrix*, the effective stiffness for each degree of freedom must be obtained separately, and it must be ensured that each component at any time is at least as stiff as the equivalent tangent stiffness. The pseudodynamic test set-up does also facilitate measurement of the individual stiffness components.

In addition to being unconditionally stable, through the operator splitting implementation, the α-Operator Splitting method also displays the desirable α-damping. This is especially useful when testing structures with a high number of degrees of freedom. With these structures, the higher, structurally insignificant modes may display a tendency to be excited by experimental errors, sometimes dominating the response. To avoid this, numerical damping is often included in the algorithms, but this may however adversely affect the overall response.

Contrary to other dissipative methods, which tend to damp significantly also the lower frequencies, α-damping can be tuned to minimally affect the lower modes but grow with frequency to effectively damp out spurious high-mode response (Géradin & Rixen 1994). The α-Operator Splitting method is therefore well recognised as an effective and versatile method, and is in common use in one of the world leading pseudodynamic research laboratories, ELSA at JRC-Ispra, Italy.

2.5.2.5 Force control

It has been suggested to carry out pseudodynamic tests under force control rather than displacement control (Thewalt & Mahin 1994). The motivation is the susceptibility of stiff structures to imposed displacement errors. Very small displacement steps may be difficult to impose accurately, and any inaccuracies in this will result in significant restoring force errors. If the test were to be run under force control, higher accuracy could be obtained, as it is significantly easier to control force than displacement.

There should, in principle, be no reason why pseudodynamic tests could not be run under force control. Thewalt & Mahin (1994) have suggested rearranging the displacement expression to solve for restoring force in the implicit hybrid method, and using the displacement signal as feedback. However, the method should be applicable to any time integration scheme. Thewalt *et al.* (1986) earlier suggested a slightly different method of implementation by not using a time integration scheme at all. Arguably pseudodynamic, the method relies on the existence of real inertia and viscous damping, and solves the equation of motion in an analogue form using transducer outputs. Requiring satisfaction of all standard similitude relationships, the method resembles shaking table tests.

2.5.2.6 Integral form time stepping algorithms

Time integration algorithms build on the principle of linearising the forces and state variables (displacement, velocity and acceleration) over the duration of the time step. This is a necessary condition to enable solution of the in-principle non-linear equation of motion, and forms the base of the numerical time integration. The potential for error of such linearisation may not be excessive, and is clearly dependent on the relationship between the rate of change of the variables and the size of the time step. One might argue that as long the size of the time step is chosen appropriately, the linearisation effects may be limited such that they do not represent a problem. However, it may not be possible to choose the time step size arbitrarily in pseudodynamic testing. This is due to the fact that the time step size greatly influences a range of other parameters of a test, as discussed under section 2.4.2 in this chapter. Under most circumstances, the time step size required for stability or to accurately

capture the structural motion that takes place will be so small that linearisation errors within the time integration will be sufficiently limited. Nevertheless, under certain conditions, linearisation of rapidly varying excitation or restoring forces may adversely affect the response to an unsatisfactory level. This is the motivation behind the development of the so-called *integral* form time stepping algorithms.

As an example, consider figure 2.2 below, illustrating a rapidly varying accelerogram, acting on a low frequency SDOF structure. In such a case, the time step size convenient when modelling the structure is considerably larger than the sampling period of the base excitation.

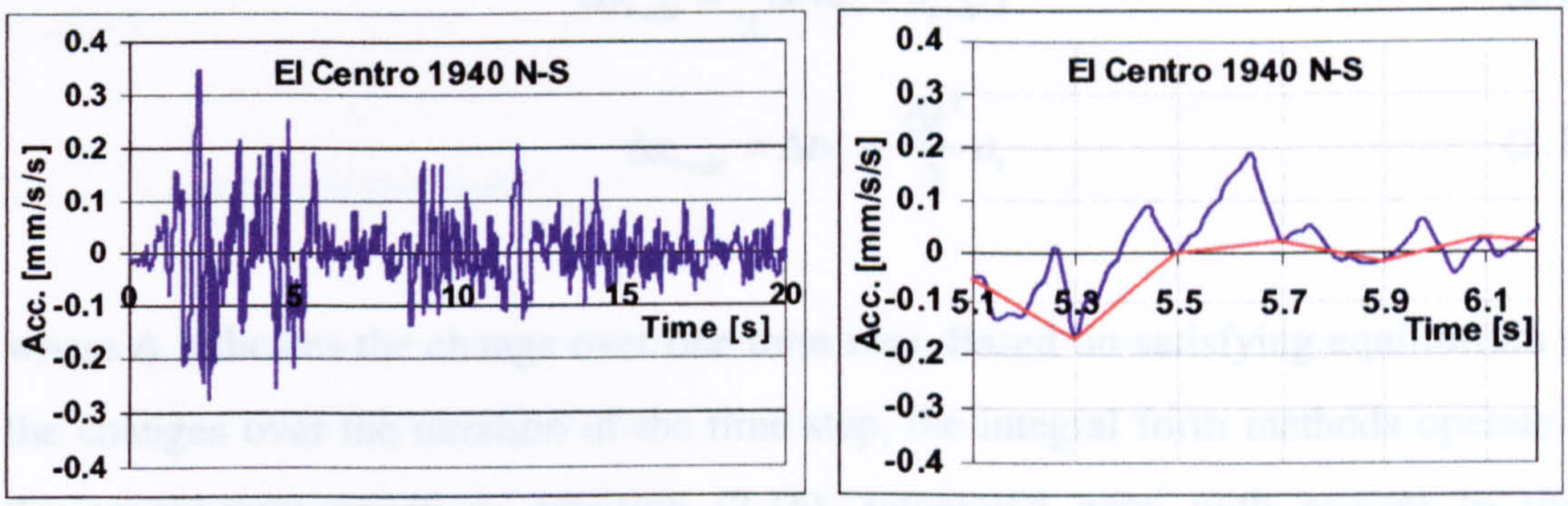


Fig. 2.2 Rapidly varying accelerogram

A considerable amount of information relating to the base motion is lost when truncating the accelerogram to only one value per time step (as indicated with red line in figure 2.2). This can, for example, be seen when linearising between say 5.5s and 5.7s on the accelerogram above. As indicated, the severity of this is of course a function of the frequency content of the accelerogram and the time step size adopted during the test.

Another situation where the linearisation of forces over a time step may result in an unacceptable loss of accuracy is where sudden changes in elastic restoring force may take place. An example of this may be when the yield point of high strength reinforcing steel in a reinforced concrete structure is reached, or when concrete crushing takes place. Conventional algorithms will typically base the calculation of

acceleration, and thus velocity and displacement, on the restoring force acting at the end of the step.

The idea of the so-called integral form was originally proposed by Chang *et al.* (1998). He suggested a procedure based on integrating an incremental form of the Newmark Explicit method with respect to time. The incremental form of this method is in principle mathematically identical to the standard, differential, form of the method described earlier. The incremental form can be expressed as follows:

$$M\Delta a_t + C\Delta v_t + \Delta r(x_t) = \Delta f_t \tag{2.16}$$

$$\Delta v_{t+\Delta t} = \frac{1}{2}\Delta t(a_t + a_{t+\Delta t}) \tag{2.17}$$

$$\Delta x_{t+\Delta t} = \Delta t v_t + \frac{\Delta t^2}{2} a_t \tag{2.18}$$

where Δ indicates the change over one time step. Based on satisfying equilibrium for the changes over the duration of the time step, the integral form methods operate on the equilibrium equation, equation (2.16), integrated once with respect to time. Integrating the three equations above with respect to time yields:

$$M\Delta v_t + C\Delta x_t + \Delta\int r(x_t)dt = \Delta\int f_t dt \tag{2.19}$$

$$\Delta x_{t+\Delta t} = \frac{1}{2}\Delta t(v_t + v_{t+\Delta t}) \tag{2.20}$$

$$\Delta\int x_{t+\Delta t}dt = \Delta t x_t + \frac{\Delta t^2}{2} v_t \tag{2.21}$$

Effectively, the equation of motion is now solved for the *change in velocity* rather than for acceleration, and expressions for the change in displacement (equation 2.20) and the change in an integral of displacement (equation 2.21) exist. However, there is no longer an explicit expression for the displacement available to provide the target displacement essential to pseudodynamic tests.

In order to enable pseudodynamic implementation, Chang *et al.* (1998) suggested a procedure building on equation (2.20). As this equation is implicit, solution can only be obtained by approximating the tangent stiffness, or replacing it with the initial stiffness, in a similar manner as in the α-Operator Splitting method described in section 2.5.2.4.

The result is an algorithm which is more capable of picking up effects of rapidly varying excitation forces and stiffness changes, while at the same time displaying improved error propagation characteristics over the standard Newmark Explicit representation (Chang *et al.* 1998). The scheme does however exhibit some numerical dissipation, and displays the inherent stability limitations of the Newmark Explicit method.

2.5.2.7 Dissipative methods

As pseudodynamic tests are highly susceptible to experimental errors, efforts have been made to develop dissipative time integration schemes. With MDOF structures, a particular problem may be spurious, unwanted excitation of high vibration modes. This is often caused by small experimental errors applied with the frequency of the time steps.

As indicated in section 2.5.2.4, it may be desirable to damp out high frequency oscillations in MDOF structures. While inclusion of viscous damping could be considered the simplest means to mitigate spurious growth of high modes in pseudodynamic tests, Shing & Mahin (1987) showed early that this was not a satisfactory method. In addition to potentially creating unrealistic results as stiffness properties degrade significantly, this form of damping may also excessively damp the fundamental and other lower modes.

A selection of schemes that display numerical dissipation exist. These include the Wilson's θ method, Houbolt's method and the family of α-methods. For the pseudodynamic application, the schemes based on the α-method are the most commonly used. The α-method, in general, displays the favourable effect that damping increases proportionally to the square of the frequency. This allows

significant damping of higher modes while the fundamental modes remain essentially unaffected.

The α-method has been incorporated in some implicit algorithms for pseudodynamic testing, including the α-Operator Splitting method mentioned in section 2.5.2.4 and a method presented by Shing *et al.* (1991), both referring to the α-HHT method (Hilber *et al.* 1977). With the two methods, the α value may be varied between 0 and –1/3 depending on the level of damping desired. This influences the β and γ variables present in the Newmark family algorithms (see section 2.5.2.1) and shifts the position of the equilibrium position (see equation 2.14).

Implicit schemes are not always employed in pseudodynamic testing. If only one or two degrees of freedom exist, explicit schemes may be preferred. Often, these may not be made dissipative, with for example the commonly used central difference method being energy stable. Although the Newmark Explicit method can be set to incorporate numerical dissipation, this tends to damp the lower modes too strongly (Chang 1997). Chang (1997) has proposed two explicit methods with improved numerical damping properties. These should essentially enable equally good damping properties as the α-method, preserving second order accuracy, but the author has not carried out a thorough analysis of these proposals.

2.6 APPLICATION

The pseudodynamic test method has its application in testing structures that display significantly non-linear behaviour during dynamic loading. Such structures would be difficult to model using numerical models alone and would in general require shaking table testing for assessment. As elaborated on in 2.4.1, certain types of structures lend themselves better to pseudodynamic testing than others. In principle, it is required that the structure may be represented reasonably well as a system of discrete masses, stiffnesses and dampers. Furthermore, the number of DOF's must be limited, as each DOF needs to be controlled by an actuator. Lastly, particularly stiff structures are more difficult to test than flexible ones, as the stiff structures are more susceptible to experimental errors during testing. Among the most commonly pseudodynamically

tested structures are reinforced concrete frame structures and reinforced concrete bridge piers.

Some centres possess the ability of modelling entire structures pseudodynamically, e.g. (Negro 1997), (Takanashi & Nakashima 1987), but most research institutions involved in such testing only have facilities with one or two degrees of freedom. This limits the specimen complexity to such an extent that is often difficult to imagine it representing an entire structure. In such cases, the research can be concentrated on testing structural components, like beam-column connections, or on the so-called substructuring technique.

2.6.1 Substructuring

The substructuring technique in pseudodynamics refers to a process where the entire structure is subdivided into substructures of which one or more parts are tested pseudodynamically but the remainder entirely computationally. The primary application of such a technique is in modelling structures whose behaviour can largely be considered essentially linearly elastic, but where a certain component will display considerable non-linearity. In such a case, the portion with a predictable response is modelled using any desired numerical technique, while the unpredictable part is modelled pseudodynamically. The two are then coupled together via several degrees of freedom.

Some examples of application of the substructuring techniques include testing of suspension bridges subject to synchronous (Negro 1997) and asynchronous (Pegon & Pinto 2000) ground excitation. In these cases, the bridge decks were considered to remain linearly elastic and could be modelled computationally, while the reinforced concrete piers, displaying significant damage accumulation and hysteretic behaviour, were modelled pseudodynamically. Other examples include steel frame structures (Shing *et al.* 1994) where only the bottom half is modelled experimentally and the remainder computationally. It is further envisaged that the technique is highly suited for soil structure interaction problems (Vannan 1991). Real-time testing of both linear and non-linear substructure systems is currently being researched at Oxford University (Williams *et al.* 1999), (Williams & Blakeborough 1998).

Chapter III:

EXPERIMENTAL SET-UP AND IMPLEMENTATION

This chapter describes the general set-up of the experimental apparatus employed in the implementation of the pseudodynamic test method. It includes an explanation of the workings behind the method, and extends to discuss the experimental system developed for the particular tests described in this thesis. The main components of the apparatus include the reaction wall, the hydraulic system, the instrumentation and the communications system, but also the specimens and their interaction with the apparatus will be discussed.

3.1 INTRODUCTION

As the pseudodynamic test method is a combined experimental/computational technique, it clearly requires some experimental apparatus. The objective of the experimental component of the test is to obtain a measure of the restoring force offered by the test structure for a given displacement history. The *interaction* between the computational component defining the displacement and the experimental component furnishing the restoring forces will be discussed in the following section.

The experimental apparatus is required to enable given displacements to be imposed on the test structure. This necessitates a substantial servo-hydraulic actuator system, which can be controlled accurately, as well as a reaction wall and a solid fixing system for the test structure. The design and construction of these are included in section 3.3. This section also includes the design of the specimen structures as well as the connection system between these and the hydraulic actuator.

The last section of this chapter describes the instrumentation system. This includes the set-up of the transducers, which generally measure displacements, force and pressure. The communication system for the transducer output to the computational space is also discussed here.

3.2 WORKINGS AND PROCESSES OF THE PSEUDODYNAMIC TEST METHOD

The pseudodynamic test method is a combined experimental/computational technique for obtaining the dynamic response of a structure to an external excitation. As discussed in section 2.1, the method relies on expressing the equation of motion, equation (2.1), with both computational and experimental terms and solving this in a step-by-step manner. This is enabled through utilisation of a time stepping scheme. The procedure in which the test progresses is illustrated in figure 2.1 and in the flowchart shown below in figure 3.1.

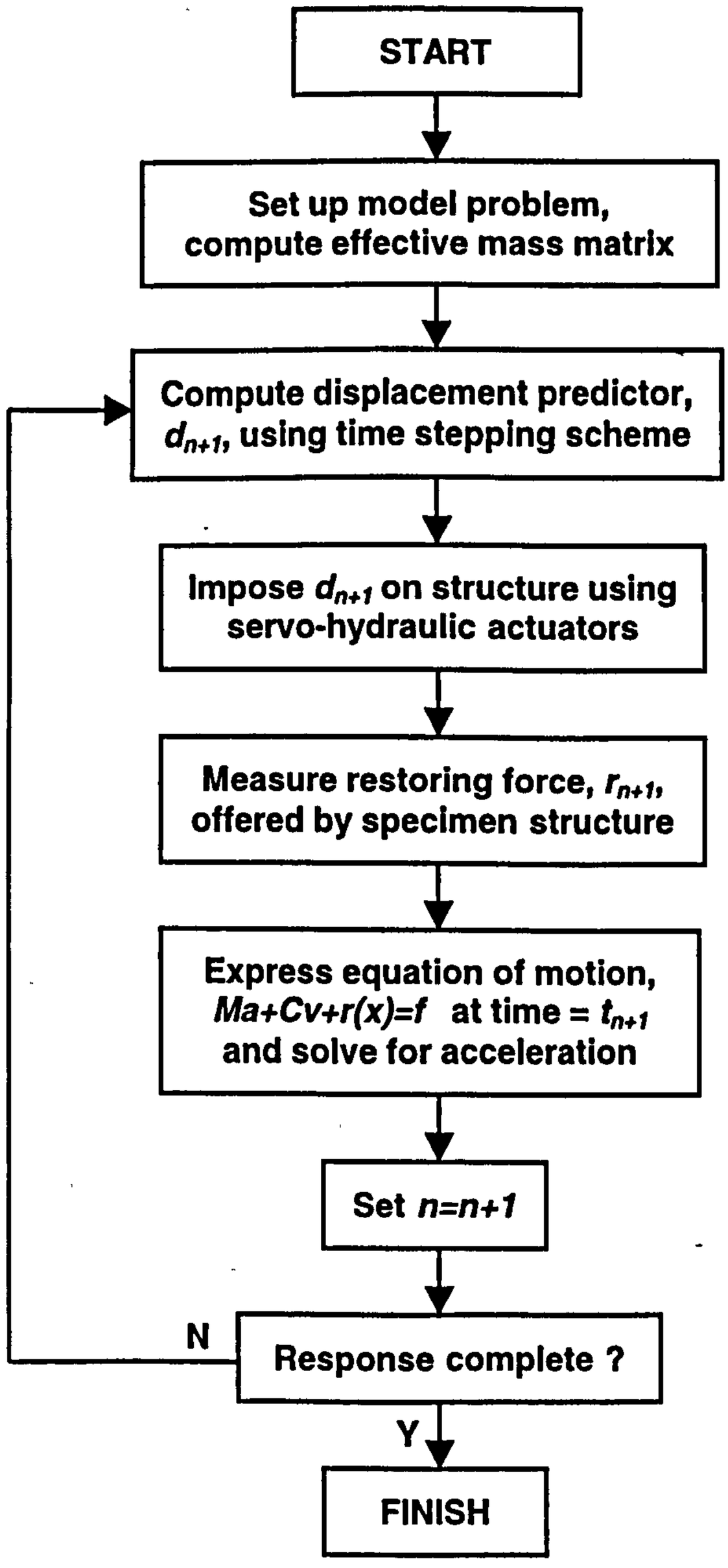


Fig. 3.1 Flowchart of main processes in pseudodynamic tests

The displacements are computed in a step-wise manner, and essentially imposed the same way. The servo-hydraulic system will work with one target displacement at a time, as the next target cannot be computed before the current displacement target has been reached and the restoring force measured at that point.

The displacement steps may be imposed at arbitrarily low speeds as inertial effects are accounted for numerically. This, to an extent, enables the use of conventional quasi-static testing apparatus to be utilised. In general, higher velocities require larger, more powerful hydraulic pumps. Therefore, if only smaller hydraulic systems are available, modest velocities should be anticipated. Furthermore, higher speeds require the controller to operate at a higher rate, and may compromise the implementation accuracy as discussed later in chapter V.

The time stepping scheme works by linearising the equation of motion over the duration of the time step. The non-linear system response is essentially obtained by keeping each time step small enough to enable neglect of the variations within each step. The discrete displacement points will together form the response when plotted as a function of the corresponding time points.

3.3 EXPERIMENTAL APPARATUS

Pseudodynamic implementation requires experimental apparatus that is able to displace the nodal point(s) of the specimen structure to the magnitude required by the computational algorithm. The system adopted for the experimentation described in this thesis consists of a horizontally orientated actuator, which is mounted to a reaction rig and operates under a hydraulic system with a remotely located servo valve. A schematic of the overall structure of the SDOF testing rig can be seen in figure 3.2 below. Each individual component of the system will be described in the following sections.

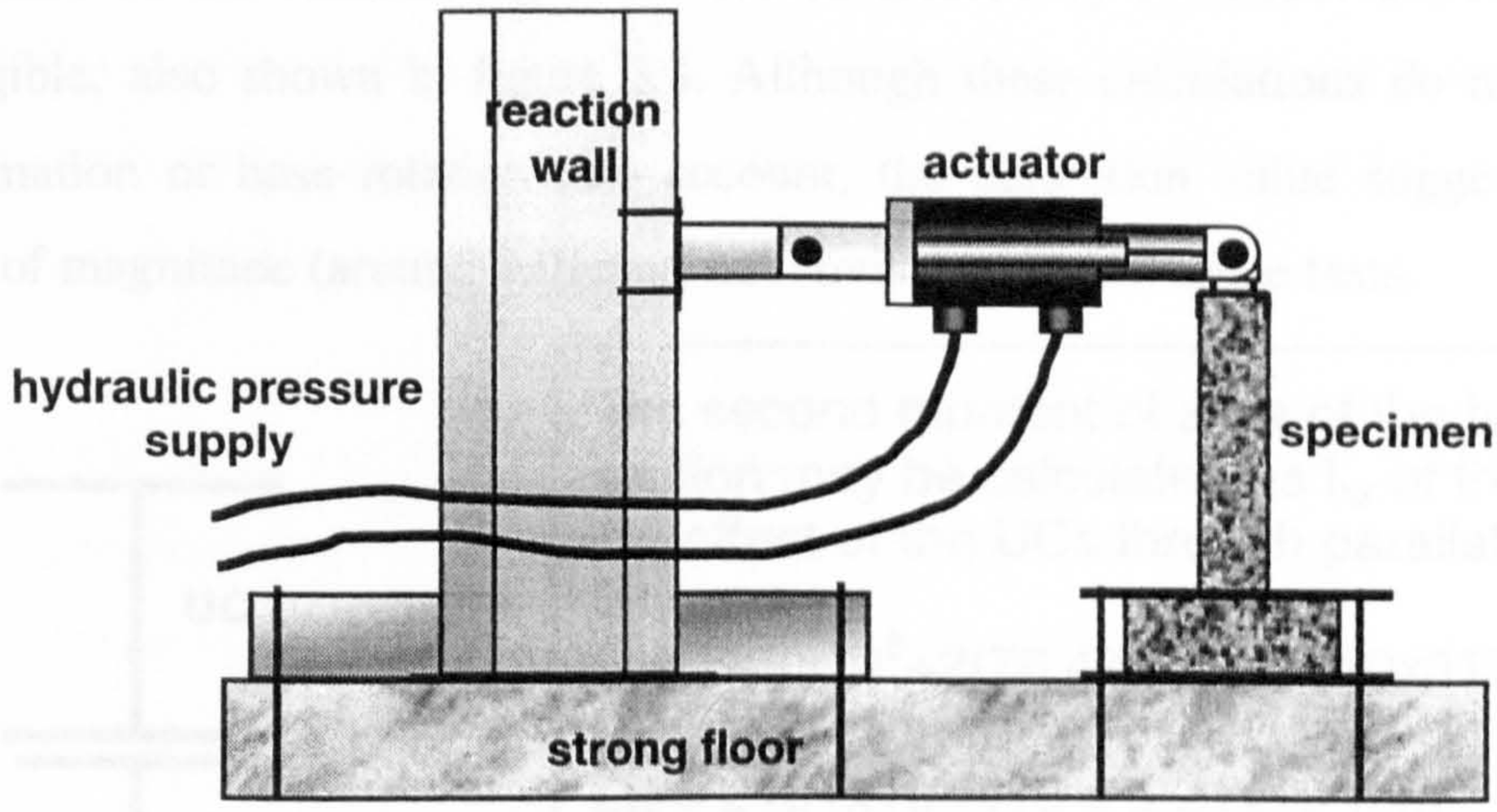


Fig. 3.2 Schematic of experimental apparatus

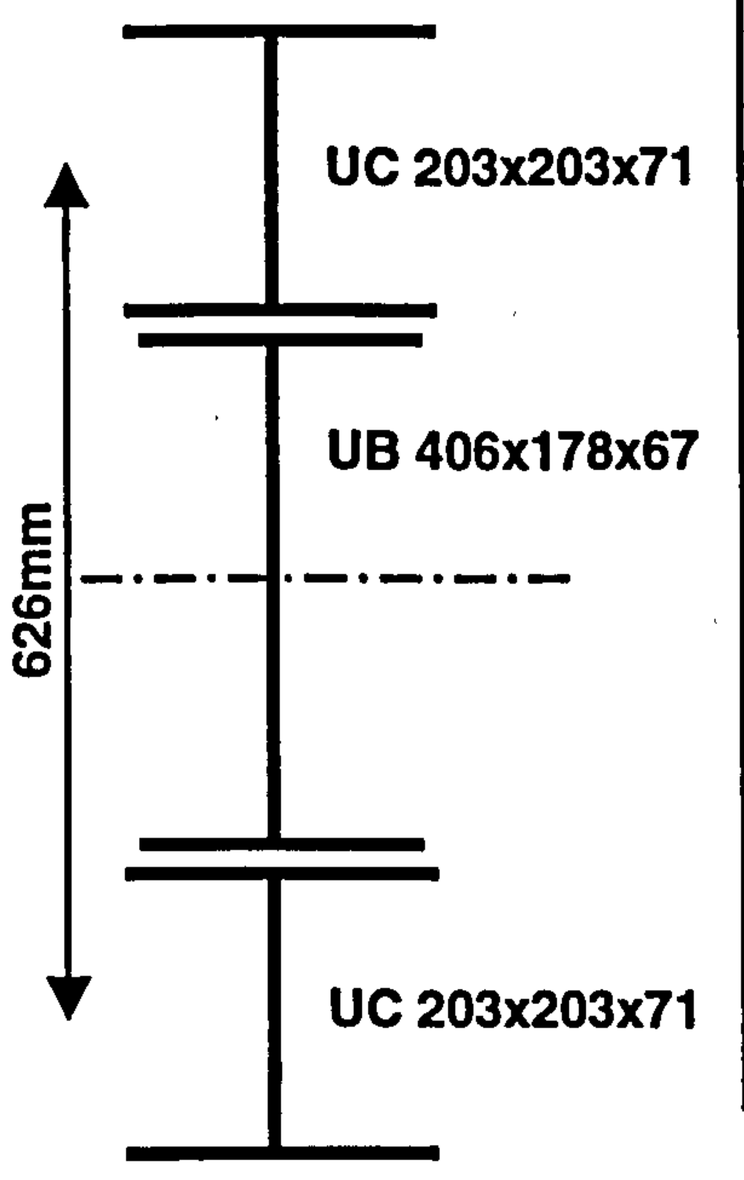
3.3.1 Reaction wall

The primary objective of the reaction wall is to offer a solid reaction to the horizontal forces created by the actuator, and to transmit these to the strong floor, preferably without any measurable deflection. It is also desirable if the actuator may be mounted at various heights on the reaction wall to accommodate a range of specimen sizes. The reaction wall is also referred to as the reaction rig or simply the rig.

It was envisaged that a range of actuators might be used, possibly up to the capacity of 200kN, so the rig was therefore designed to withstand a maximum force of 300kN, with a factor of safety of 1.5. The rig was designed to take the maximum force at a height of 2m above the strong floor, effectively creating a maximum bending moment of 600kNm. The capacity of the actuator available was however somewhat smaller, 50kN, and generally positioned at around 1m above the floor. The typical bending moment present would therefore be more likely to be of the order of 50kNm.

Following investigation of a range of conceptual design ideas, a system using a triple section for the main vertical member was elected. This was built up of two 203x203x71 universal columns (UC) separated by a 406x178x67 universal beam (UB). The total second moment of area for bending around the strong axis can be approximated to 2181×10^6mm4 as shown in figure 3.3. This provides a theoretical moment capacity of around 1431kNm. Basic deflection calculations suggest that

deflection of the reaction rig under forces created by the 50kN actuator would be negligible, also shown in figure 3.3. Although these calculations do not take shear deformation or base rotation into account, the deflection value suggested is of an order of magnitude (around 0.01mm) not envisaged to influence tests.



The second moment of area of the built up section may be calculated as I_{xx} of the UB plus the effect of the UCs through parallel axis theorem.

$I_{xx} \approx 243x10^6+2(76.47x10^6+9110x313^2)$

$I_{xx} \approx 2181x10^6 mm^4$

$Z_{xx} = 2181x10^6/419 = 5.21x10^6 mm^3$

$M_{cap} = Z_{xx} \times p_y = 5.21x10^6 x275 =$ 1431kNm

Shear area $\approx 360 \times 8.8 = 3168 mm^2$

Shear stress $= 3000000/3168 \approx 95.7 N/mm^2$

Low Shear.

Deflection:

$\delta = PL^3/3EI$

$= 50000x600^3/3x210x10^3x2181x10^6$

$=$ 0.0079mm

Fig. 3.3 Moment and shear capacity of built up beam.

The three sections were bolted together using 2x20mm grade 8.8 bolts at a spacing of 300mm along the height of the beam. The theoretical maximum shear force of 300kN would develop 747kN per metre spacing as shown in equation (3.1)

$$Q = \frac{SA\bar{y}}{I_{xx}} = \frac{300000 \times 9110 \times 310}{1134 \times 10^6} = 747 kN/m \qquad (3.1)$$

where S is the maximum shear force, A the area of the section bolted on, $\bar{y}$ the distance from the neutral axis to the centre of the section bolted on and I_{xx} the second moment of area of the total section. Assuming a bolt spacing of 250mm generates a force in each set of bolts of 747kN/m x 0.25m = 186kN. The specified bolts provide 184kN capacity, which should be sufficient in all expected cases.

The reaction rig requires a substantial base to transmit the design moment of 600kNm. This base relies in turn on being connected to the strong floor through holes at a

spacing of 863mm in both directions. Because of the potentially large moment and the size of the vertical beam, it was decided to span two spacings in the direction of the loading and one gap in the transverse direction as shown in figure 3.4. A general view of the rig can also be seen in figure 3.13.

The vertical beam was welded to the 25mm thick base plate with heavy welds on all edges. The beam was assumed to bend around the centre of the UC sections, so the lever arm was taken as 620mm. With a design maximum moment acting at the base of 600kNm, the maximum uplift force will be of the order 600/0.620 = 968kN. In addition to this, a maximum shear force of 300kN will be acting perpendicularly. Assuming the weld around the UC's will be resisting the moment, and the weld around the web of the UB the shear, the maximum stress on weld is around 0.9kN/mm. An 8mm fillet weld was applied throughout, providing 1.2kN/mm.

The base consisted of a 25mm thick plate stiffened by two 406x178x60 UBs. The stiffeners were required to carry the moment developed at the base of the beam to the two connection points on the floor, and were welded to the plate. With the moment from the vertical beam transferred to the base plate, this would also have to be able to carry 600kNm. This moment requires an elastic modulus of $600x10^6/275 = 2.18x10^6 mm^3$. Two 406x178x60 UBs provide an elastic modulus of $2.12x10^6 mm^3$, so together with the plate itself, this provided sufficient stiffness.

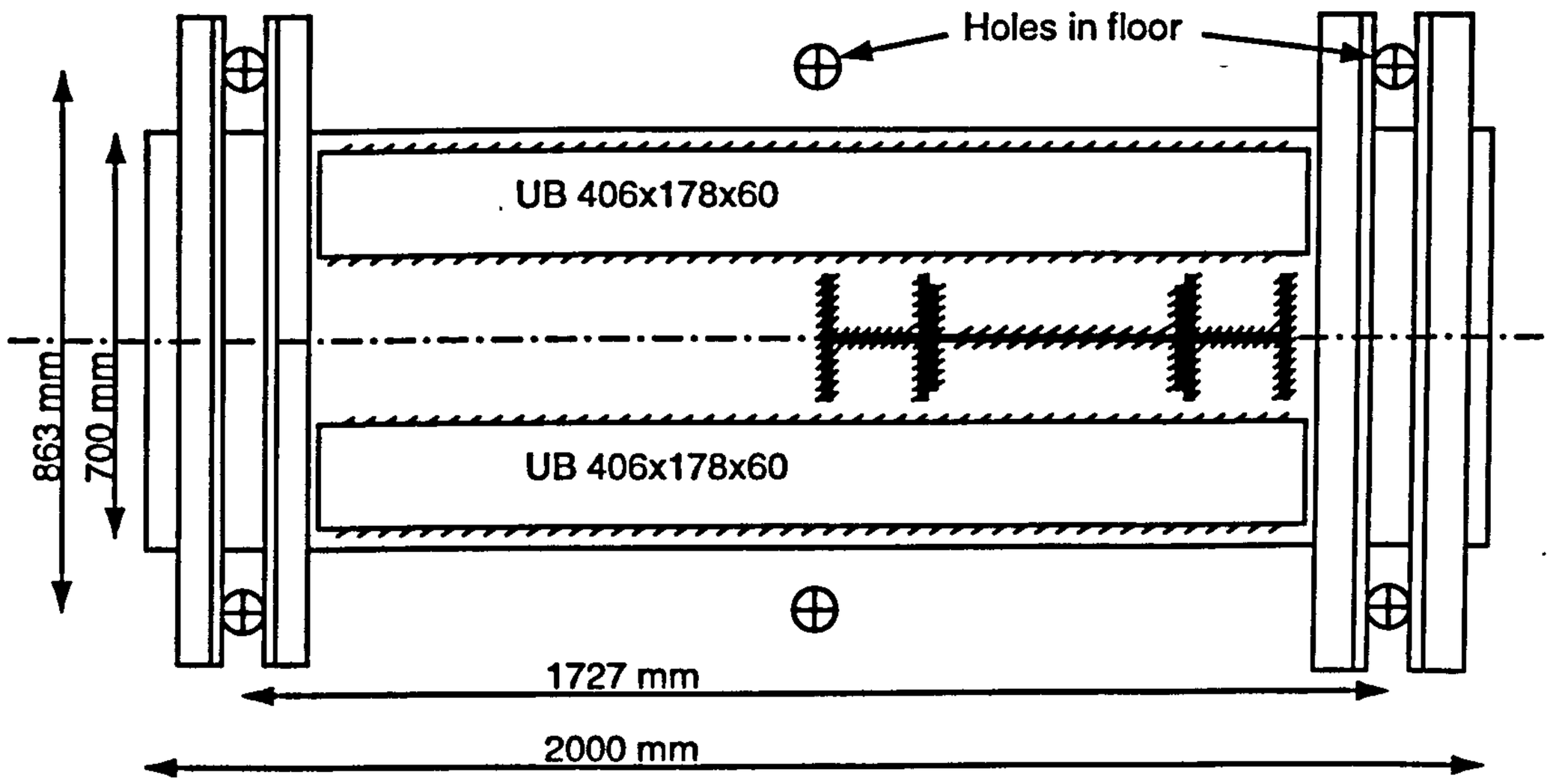


Fig. 3.4 Plan view of reaction wall base.

In order to connect the base plate to the strong floor, two pairs of channel sections were welded onto the plate. These extruded beyond the plate to coincide with the hole locations in the floor, as shown in figure 3.4. Screwed rod could be passed between the channels and through the holes in the floor. The sections used were universal channels, 176x76 and 254x89, with the largest situated on the side closest to the beam, as seen in figure 3.5.

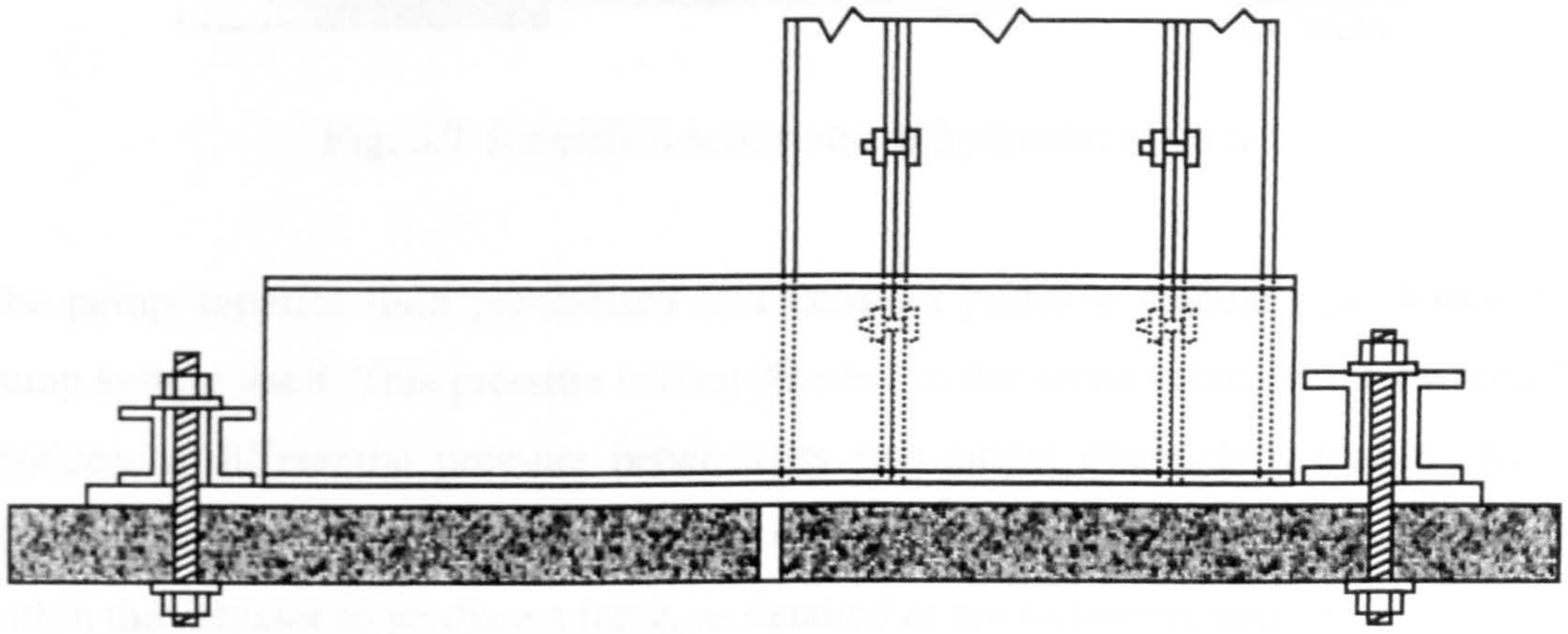


Fig. 3.5 Side elevation of reaction wall base

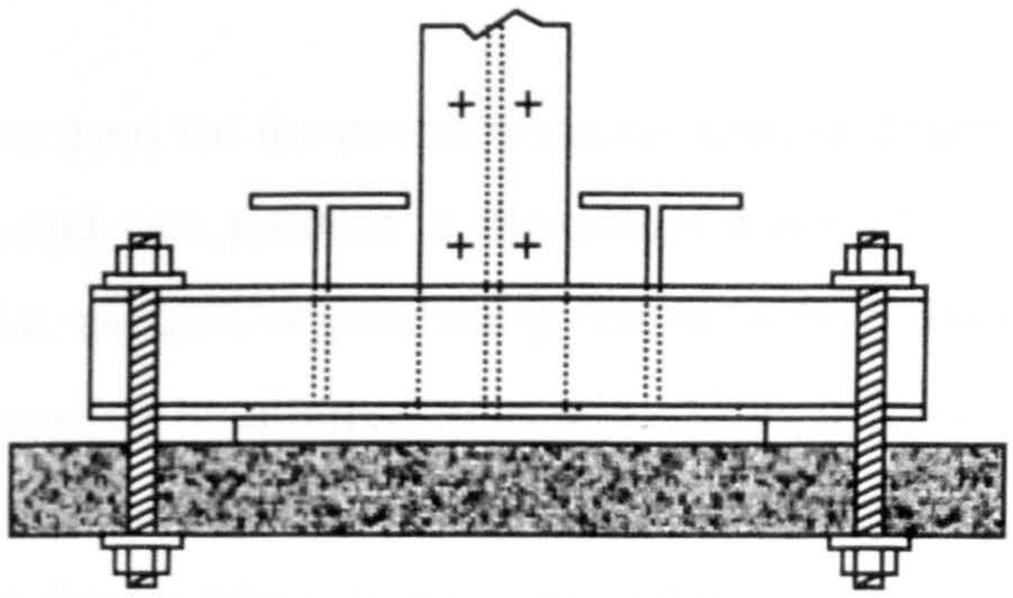


Fig. 3.6 End elevation of reaction wall base

3.3.2 Hydraulic system

The hydraulic system comprises all the components that enable a force to be applied to the specimen structure. This includes first of all the actuator, which, of course actually produces the force. However, the actuator requires hydraulic fluid under pressure to exert a force. The pressure is created by the pump system, which further requires a valve to control the pressure supply. A schematic of the main components of the hydraulic system can be seen in figure 3.7 below.

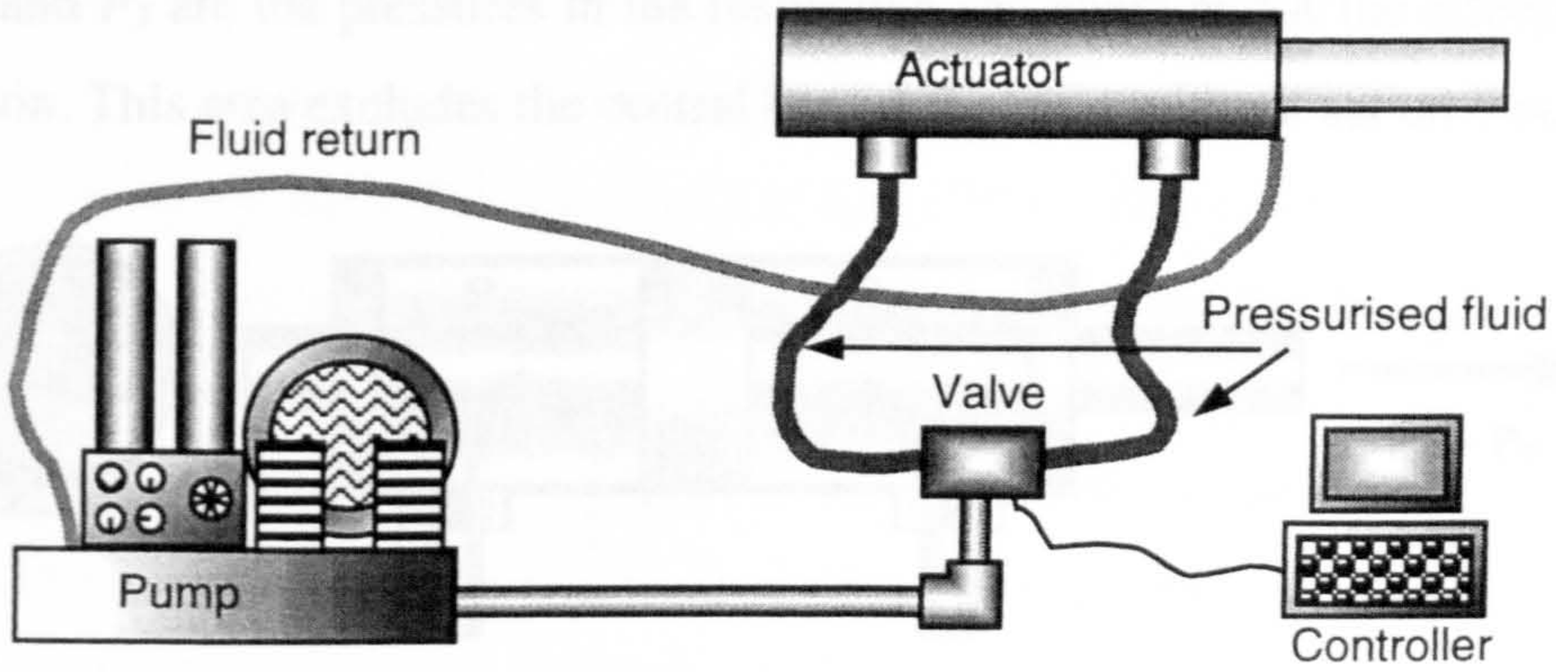


Fig. 3.7 Simplified schematic of hydraulic system

The pump supplies fluid pressurised to a constant pressure manually set within the pump system itself. This pressure is then directed to the servo valve, which essentially produces a differential pressure between its two output channels controlled by an electrical signal from the controller. The two pressure hoses connect to the chambers within the actuator to produce a force, as detailed in the following section.

3.3.2.1 Hydraulic actuator

The hydraulic actuator used for the pseudodynamic testing described in this thesis has a stroke of ±50mm and can produce a maximum force of ±50kN. Although of a double acting type, i.e. capable of producing forces in both directions, it is a simple device, and should possibly be referred to as a "jack" as it does not contain any servo valve. However, as it operates with a remotely located valve, it is referred to as "the hydraulic actuator" or simply "the actuator" throughout this text.

The hydraulic actuator operates under the principle of creating a force by differential pressures acting over a surface. Essentially, it contains two chambers connected to the two output points on the servo valve. The fluid in these two chambers both act on the same piston, and the force exerted will in principle equal the pressure difference multiplied by the piston area. The principle is depicted in figure 3.8 and defined in equation (3.2),

$$F = (P_1 - P_2)A \tag{3.2}$$

where P_1 and P_2 are the pressures in the respective chambers and A the effective area of the piston. This area excludes the central bar, as the fluid will not act on this.

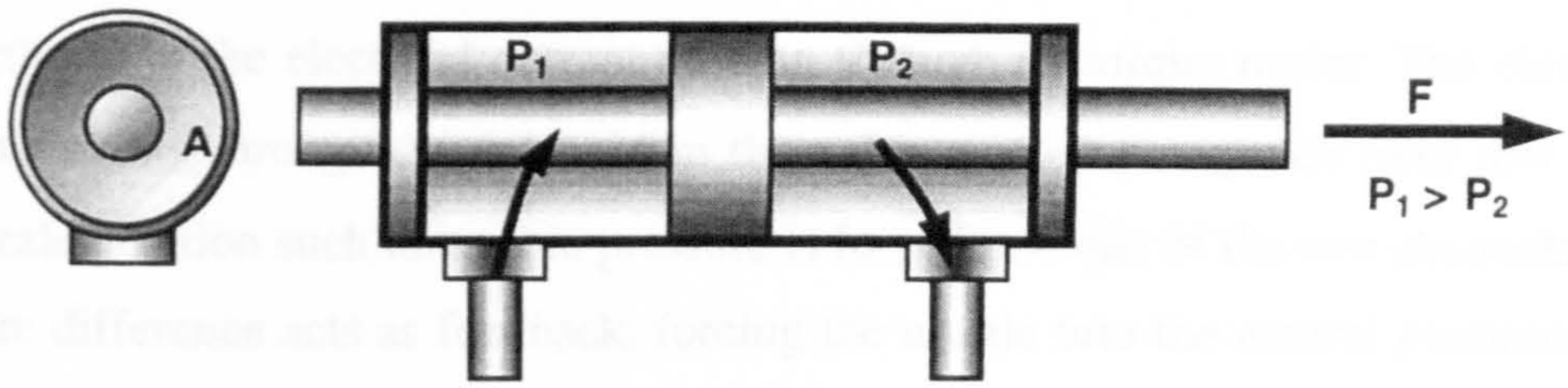


Fig. 3.9 Principle of double acting hydraulic actuator

By sending a higher pressure to one chamber, say the one on the left hand side in the figure above, the pressure on the left hand side of the piston will be greater than on the right hand side. This creates a force in the direction of the arrow. Depending, of course, on the resistance acting on the actuator, the piston will move, and hydraulic fluid will flow into the left hand chamber and out of the right hand one. A system also exists for the returning fluid, which invariably seeps through the seals and rings on the piston.

The effective force exerted by the actuator is in practise not exactly as defined in equation (3.2). This is due to the internal friction in the actuator, which first has to be overcome before any movement can be initiated. The internal friction may affect the force measurements if a certain system is adopted. This is further discussed in section 3.4.1.1.

3.3.2.2 Hydraulic pumps and servo valve system

The experimental facility described here has been constructed and operated in the Heavy Structures Laboratory in the Civil Engineering Department at the University of Glasgow. This laboratory benefits from a large-scale built-in hydraulic pump and valve system. The pumps are capable of producing a pressure up to around 250bar and a flow of 25 litres per minute (l/m). Essentially, the pump supplies a constant pressure to the servo valve. The pressure can be set manually to any value between 0 and 250bar, and this pressure then represents the maximum the valve is capable of delivering to the actuator.

The servo valve is a Moog type pressure control valve. It supplies pressure to two output ports connected to the two chambers in the actuator through pipes and pressure hoses. In broad terms, the valve produces a differential pressure between the two ports proportional to the electrical current passing through its torque motor. The electrical current flowing through the solenoid in the valve creates a magnetic field that shifts the nozzle position such that more pressure is forced into one of the two channels. The pressure difference acts as feedback, forcing the nozzle into the neutral position once the force created from the pressure difference equals that generated by the magnetic field. Higher electrical currents thus result in higher pressures, and reversing the electrical current will reverse the pressure.

3.3.3 Test specimens

The design of the test specimens was naturally limited by the experimental testing facilities available. In particular, the fairly modest stroke of the actuator, ±50mm, and the maximum force of ±50kN had to be respected. Following these limitations, it was vital that the specimens were designed such that useful and relevant tests could be carried out. As pseudodynamic tests generally aim to obtain the response of structures under non-linear conditions, it was considered that the apparatus should enable displacements well beyond the elastic range of the tested structures. Consequently, this put limitations on both the strength and stiffness of the specimens.

Two specimen types were employed in the experimentation described in this thesis. These were a *reinforced concrete stub column* and a *slender steel column*. Both operated with a virtual mass at the top yielding inverted pendulum structures. However, they were different in the sense that the reinforced concrete column was designed to display significantly non-linear behaviour while the steel column should remain elastic within the stroke range of the actuator.

3.3.3.1 Reinforced concrete test specimen

The reinforced concrete specimen was designed to validate the pseudodynamic tests system rather than to investigate the specimen behaviour itself. It was therefore important that the specimen could be failed with the experimental apparatus available.

To ensure failure, the specimen should not be able to resist the maximum force of 50kN in bending or shear and, equally importantly, should not be so flexible that displacements of 50mm could be reacted without significant damage.

The requirements in terms of strength were such that the section could not be made too large, nor the length too short. However, smaller sections and longer lengths produce a more flexible structure, which might not satisfy the stiffness requirements. It was found that a 100x200mm section and a length of only 600mm, as shown in figure 3.10, would fulfil all the requirements. The column was reinforced with 4x12mm high strength bars and built into a solid, reinforced base. The full engineering drawings of the concrete specimen can be found in Appendix D.

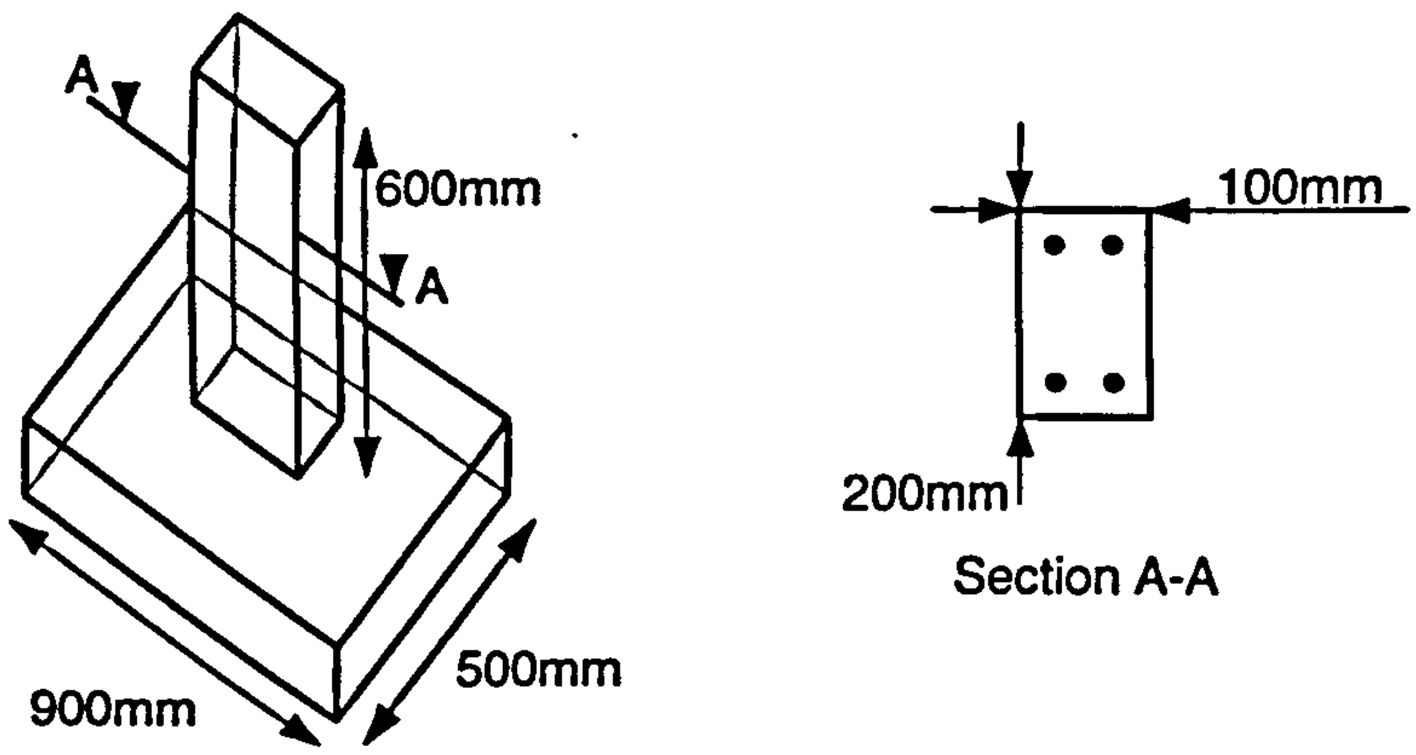


Fig. 3.10 Schematic representation of reinforced concrete specimen

The force from the actuator is applied horizontally, so assuming cantilever behaviour about the strong axis, the maximum moment that can be applied at the base is 50kNx0.6m=30kNm. The moment capacity of the section has been very roughly estimated using two methods. First, by ignoring the compression reinforcement, the moment capacity can be computed as in equation (3.3). Applying British Standard BS8110, using 30N/mm^2 concrete cube strength, 20mm cover and excluding partial safety factors yields:

$$M = 0.234bd^2 f_{cu} = 0.234 \times 100 \times 174^2 \times 30 \times 10^{-6} = 21.3kNm \qquad (3.3)$$

Motivation of this moment capacity requires a steel area of 342mm^2. As only 226mm^2 is provided, the strength will be somewhat lower, possibly around 15kNm. However,

compression steel is effectively provided although no corresponding additional tension steel is present. The strength can alternatively be calculated ignoring the concrete (to develop the full lever arm between the bars), as shown in equation (3.4), using a steel yield strength of 460N/mm^2.

$$M = f_y A_s z = 460 \times 226 \times 148 \times 10^{-6} = 15.4 kNm \tag{3.4}$$

Both the calculation in equation (3.3) and equation (3.4) indicate approximately the same strength in bending and should be well within the capacity of the actuator. In terms of shear capacity, this can also be assessed using BS8110. Links are provided at 100mm intervals, and 6mm mild steel (f_y=250N/mm^2) is used. The shear capacity of the section, V_c, can be computed as $V_c=v_c bd$, where v_c is the design shear stress, b the breadth of the section and d the effective depth. v_c is computed according to equation (3.5) from Table 3.9 in BS8110 (1985).

$$\begin{aligned} v_c &= 0.79\left(\frac{100\times A_s}{bd}\right)^{\frac{1}{3}}\left(\frac{400}{d}\right)^{\frac{1}{4}}\left(\frac{f_{cu}}{25}\right)^{\frac{1}{3}} \\ &= 0.79\left(\frac{100\times 226}{100\times 174}\right)^{\frac{1}{3}}\left(\frac{400}{174}\right)^{\frac{1}{4}}\left(\frac{30}{25}\right)^{\frac{1}{3}} = 1.13 N/mm^2 \end{aligned} \tag{3.5}$$

This provides a shear strength of 1.13x100x174x10^{-3}=19.7kN. The contribution from the links, v_l, can be computed according to equation (3.6) from Table 3.9 in BS8110, where A_{sv} is the effective area of the shear links and s_v the spacing.

$$v_l = \frac{f_y A_{sv}}{b s_v} = \frac{250\times 56}{100\times 100} = 1.4 N/mm^2 \tag{3.6}$$

This increases the shear capacity by 1.4x100x174x10^{-3}=24.4kN. The total shear capacity is thus around 44kN, somewhat less than the 50kN capacity of the actuator.

Lastly, the flexibility has to be limited so that failure occurs within the ±50mm stroke of the actuator. Assuming a maximum inelastic strain of the reinforcing bars in tension and Bernoulli-Navier thin beam theory, the maximum deflection at the top of

the beam can be computed according to equation (3.7) below. Here, w is the maximum deflection at the tip of the beam, 0.023 the maximum strain in the steel, x the length of the beam and z the distance from the steel to the neutral axis of the section, which in this case is a minimum of 100mm.

$$w = \int \frac{u}{z} dx = \int \frac{0.023x}{z} dx = \frac{0.023x^2}{2z} = \frac{0.023 \times 600^2}{2 \times 100} = 41.4mm \qquad (3.7)$$

The approximated maximum displacement of 41.4mm is within the actuator stroke of ±50mm, but not by a clear margin. However, it was anticipated that the cyclic nature of the test would in any case result in failure of the specimen structure.

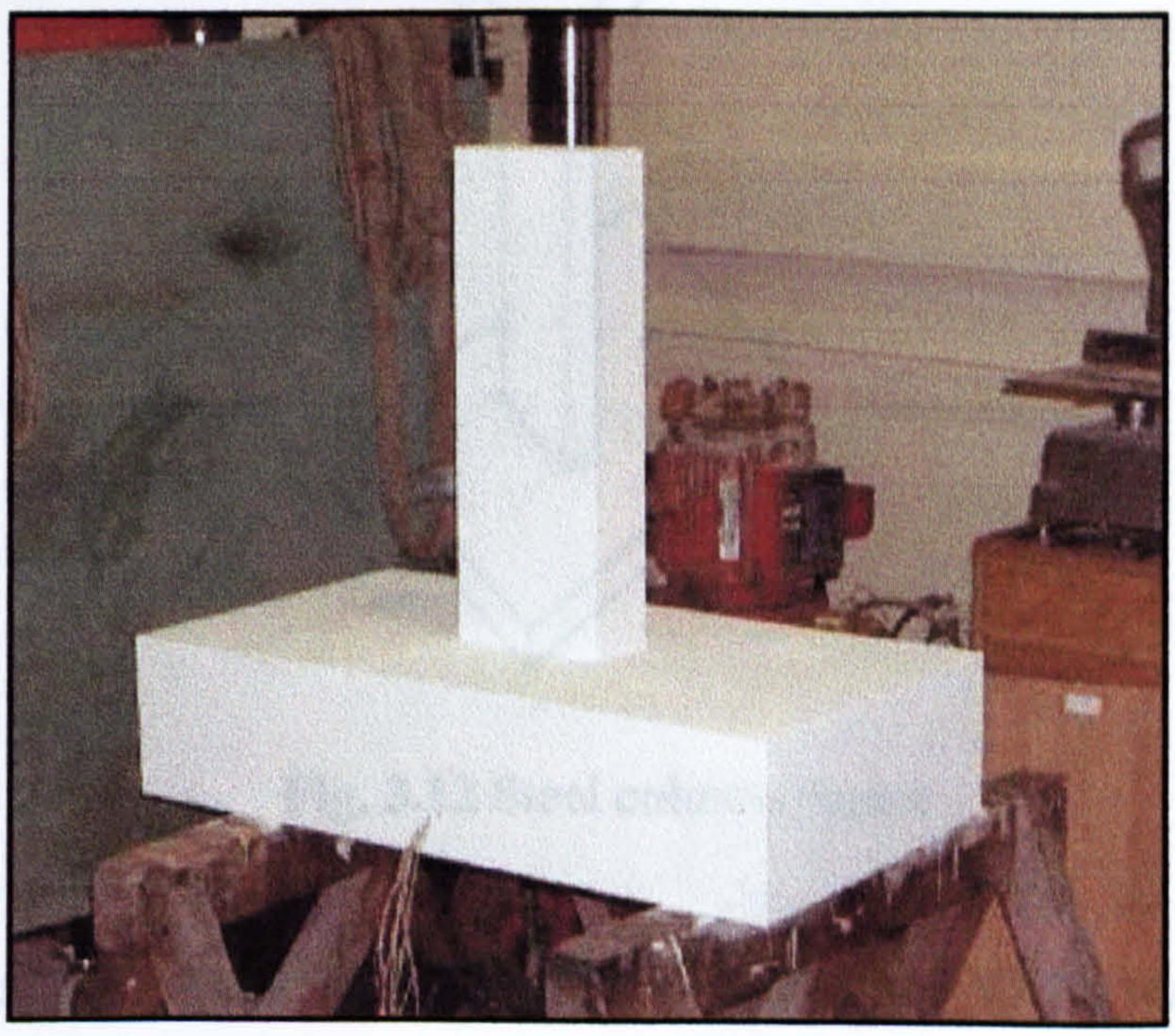


Fig. 3.11 Photograph of reinforced concrete column

3.3.3.2 Steel test specimen

The main objective of the steel specimen was to obtain a reference solution and enable a verification of the pseudodynamic implementation system. As explained in section 7.1.2.2, the specimen had to effectively remain elastic throughout these tests. Furthermore, as a fairly low fundamental frequency was desired, again explained in chapter VII, a flexible specimen had to be designed.

A hollow steel section welded to a steel base with a concentrated mass at the top was envisaged, yielding an inverted pendulum structure. A schematic of the layout can be seen in figure 3.12 below.

The design criteria were somewhat different from the concrete specimen. Again, the 50mm stroke had to be respected, but in this case the objective was to utilise as much as possible of this while keeping the specimen structure within the elastic limit. The force limitation was not expected to be critical in this design as the structure would be much more flexible.

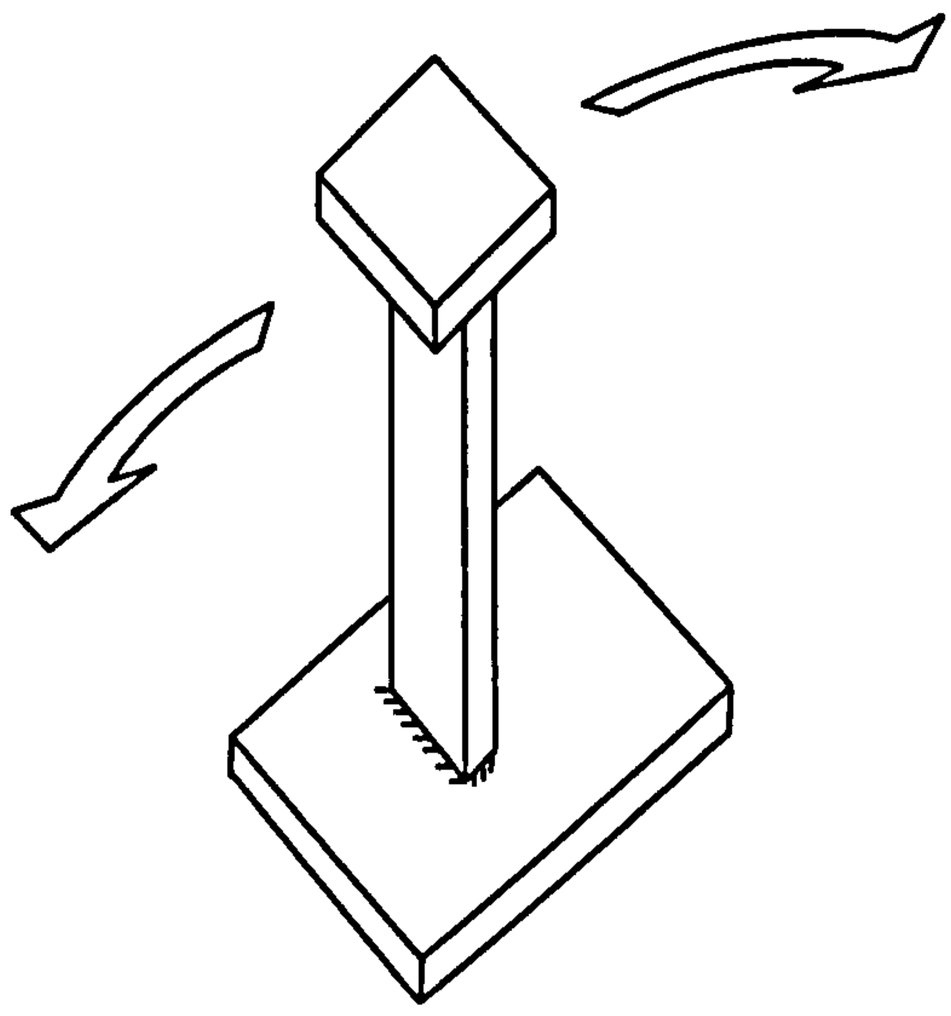


Fig. 3.12 Steel column layout

For the so-called "snap-back" test documented in Chapter VII, it was decided to aim for a maximum natural frequency of around 3Hz, or 18.85rad/s, for the dynamic structure. It was further desirable to limit the mass to around 200kg for practical and safety reasons. This yielded a maximum structural stiffness of 71kN/m as computed in equation (3.8).

$$\omega = \sqrt{\frac{k}{200}} < 18.85 rad/s \qquad k < 71000 N/m \tag{3.8}$$

Assuming cantilever behaviour, the stiffness may be computed as $k=3EI/L^3$, where E is Young's Modulus of the steel, I the second moment of area and L the length of the

beam. This yields the relationship between I and L to produce a stiffness of k=71000N/m as shown in table 3.1

I [cm^4]	L [mm]
11.5	1000
38.9	1500
46.7	1593

Table 3.1 Combinations of I and L to yield a stiffness of 71kN/m

Selecting a length of 1600mm and an available rectangular hollow section (RHS) providing a second moment of area of 46.7cm^4 will provide a stiffness of approximately 70100N/m, yielding a natural frequency of approximately $\omega_n = \sqrt{70100/200} = 18.72 rad/s = 2.97 Hz$.

Carrying out deflection calculations will give an indication of the load required for maximum displacements, where δ is the deflection and P the corresponding force, as shown in equation (3.9) below.

$$\delta = \frac{PL^3}{3EI} \Rightarrow 50 = \frac{P \times 1600^3}{3 \times 205000 \times 467000} \Rightarrow P = 3506\,N \qquad (3.9)$$

Equation (3.9) shows that a fairly modest force of the order of 3.5kN would be required to initiate the maximum displacement of 50mm. This force would create a moment at the base of the order of 3.5x1.6=5.6kNm. The uplift force on the weld at the base of the column will be of the order of 5.6kNm/0.05m=112kN, resulting in a stress of approximately 112kN/100mm=1.12kN/mm. This normally requires a weld with 8mm leg length (providing 1.2 kN/mm).

The section chosen, RHS 100x50x4, has an elastic modulus around its weak axis of 18.7cm^3. This provides an elastic bending strength of 18.7x275=5.15kNm, and a plastic bending capacity of 21.7x275=5.97kNm. The maximum applied load should in other words result in a moment somewhere between the elastic and plastic limits, which should be satisfactory for this application.

The mass for the top of the column was provided by 4 steel blocks, each weighing around 50kg rigidly connected in two pairs, as shown in the top left corner of figure 3.13. The effective point mass on the column, consisting of the total mass of the steel pieces and fittings on top in addition to the equivalent point mass of the column self weight, was approximately 275kg.

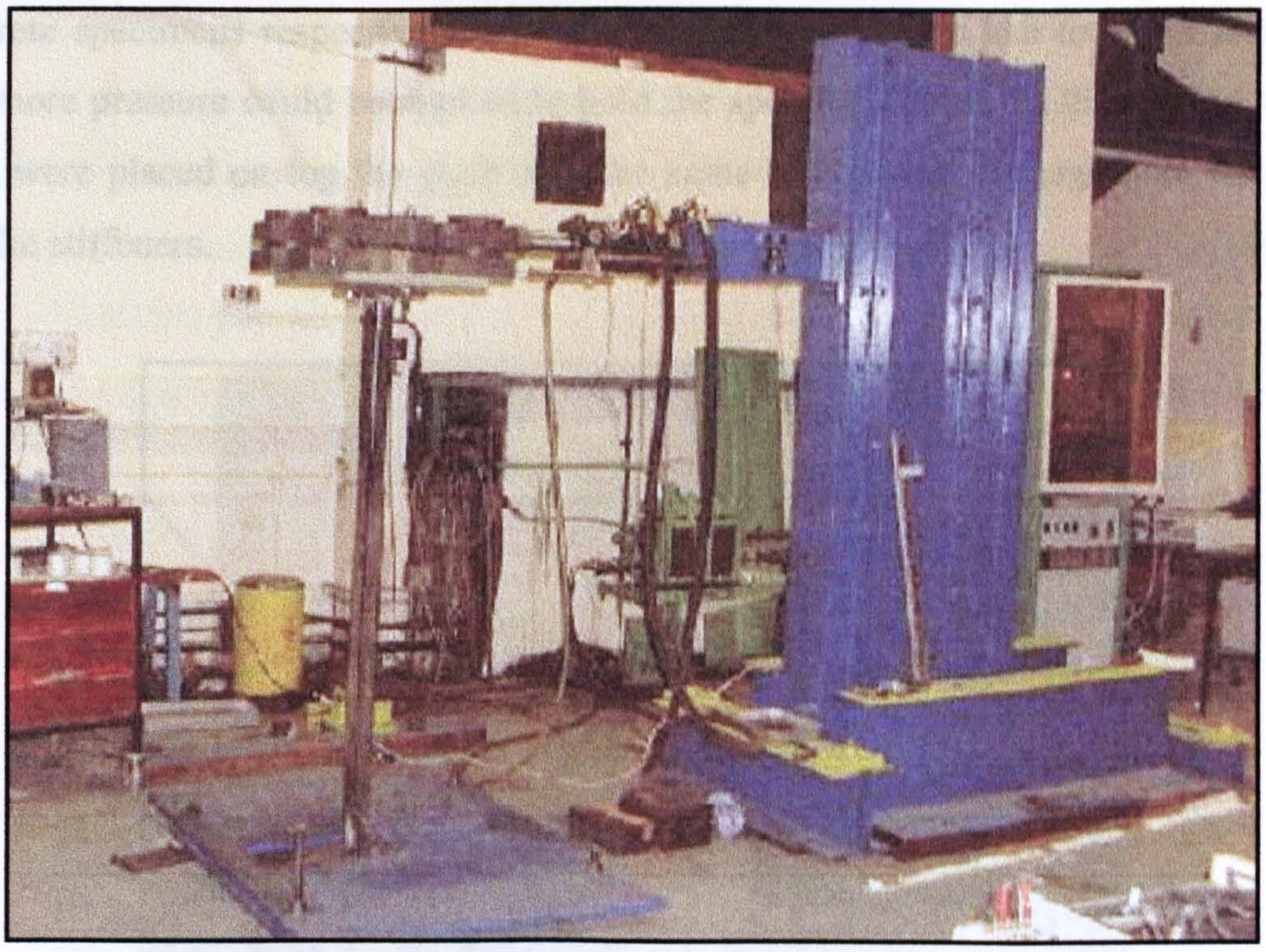


Fig. 3.13 Steel column in testing position

3.3.4 Fixings and connection

Both the reinforced concrete and steel columns required a solid fixing system to ensure that the bases did not move during tests. In pseudodynamic testing, it is generally specified that the base should remain stationary so that the relative displacements equal those induced by the actuator. This implies that the structure not only needs to be fixed against translation but also against rotation. Essentially, the structure needs to be rigidly connected to the strong floor.

The reinforced concrete column was built into a 900x500x200mm concrete base while the steel column was welded onto a 500x500x25mm steel plate. Both structures were connected to the strong floor by sandwiching them between two 25mm steel plates, where the top one had a 200x400mm hole cut out. The bottom plate was placed directly upon the strong floor, followed by the specimen and the top plate without any kind of liner, as seen in the bottom left of figure 3.13 and figure 3.14 for the steel and concrete specimens respectively. In some cases, the top plate had to be stiffened so that more pressure could be applied to hold the specimen down. In these cases, some RHS were placed on top the plate with the same rods passed through both the plate and the stiffeners.

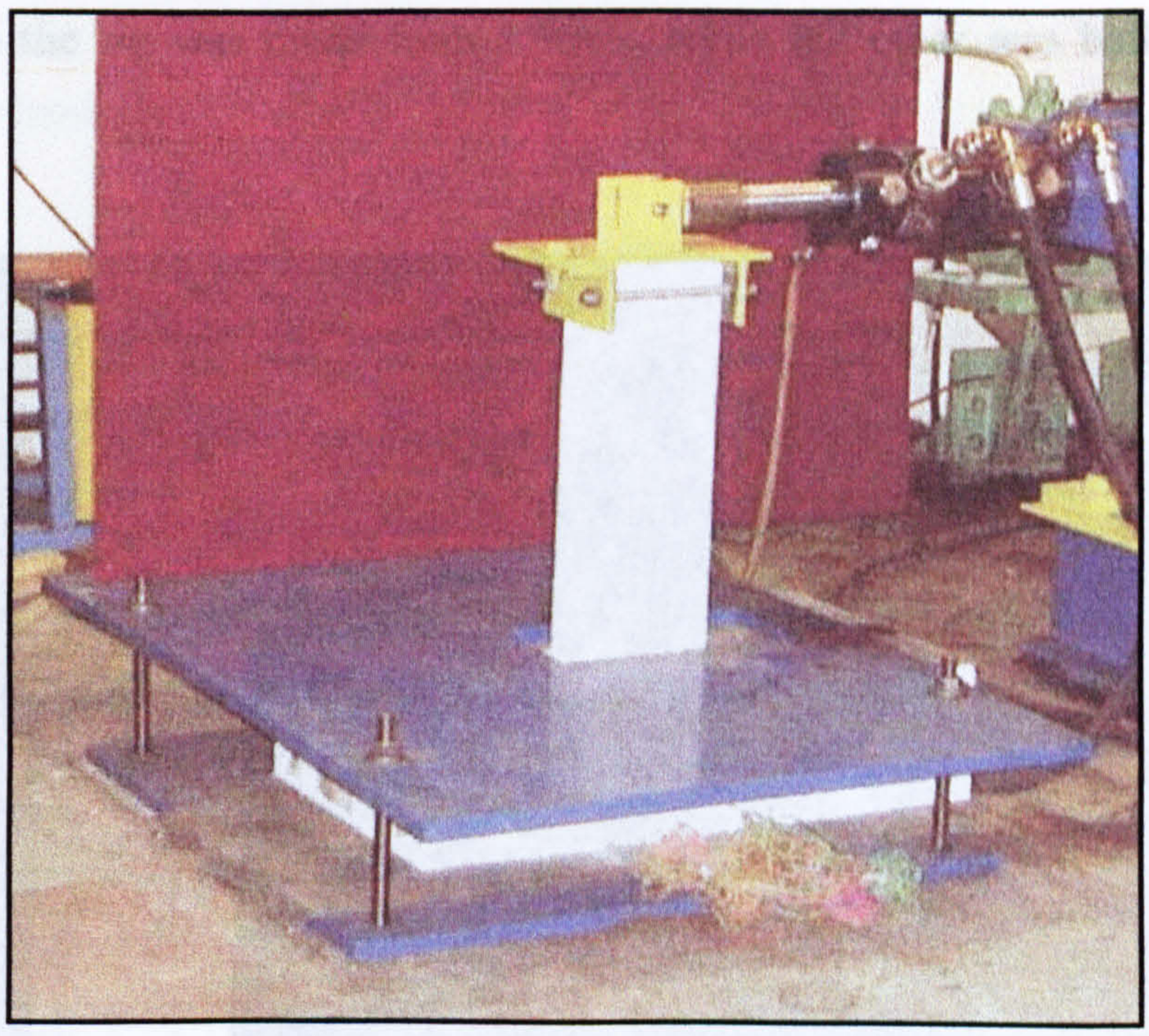


Fig. 3.14 Concrete column fixings

Figure 3.14 and 3.15 also show the connection between the actuator and the top of the concrete column. This connector needed to transfer the full force of up to 50kN from the actuator to the column, but as no moment transfer was assumed in the analysis, the fitting was pinned to the actuator. During full-scale tests, a significant rotation took place around this pin.

With the steel specimen, connection to the actuator was somewhat simpler. In this case, the fixing could simply be bolted onto the plate, which also carried the mass. The force transferred here is also of the order of one tenth of that in the concrete column.

The connection between the reaction rig and the actuator required detailed design and analysis as this "arm" effectively cantilevered out from the rig. It had to carry its self-weight in addition to the axial force created in the actuator. Furthermore, it also needed a pinned joint to avoid moment transfer. The arm was constructed in two pieces, where one was rigidly connected to the reaction rig and one to the actuator, with a lockable pin linking the two together, as seen in figure 3.15. The piece connected to the rig was made from a RHS, while the other was built up of steel plates.

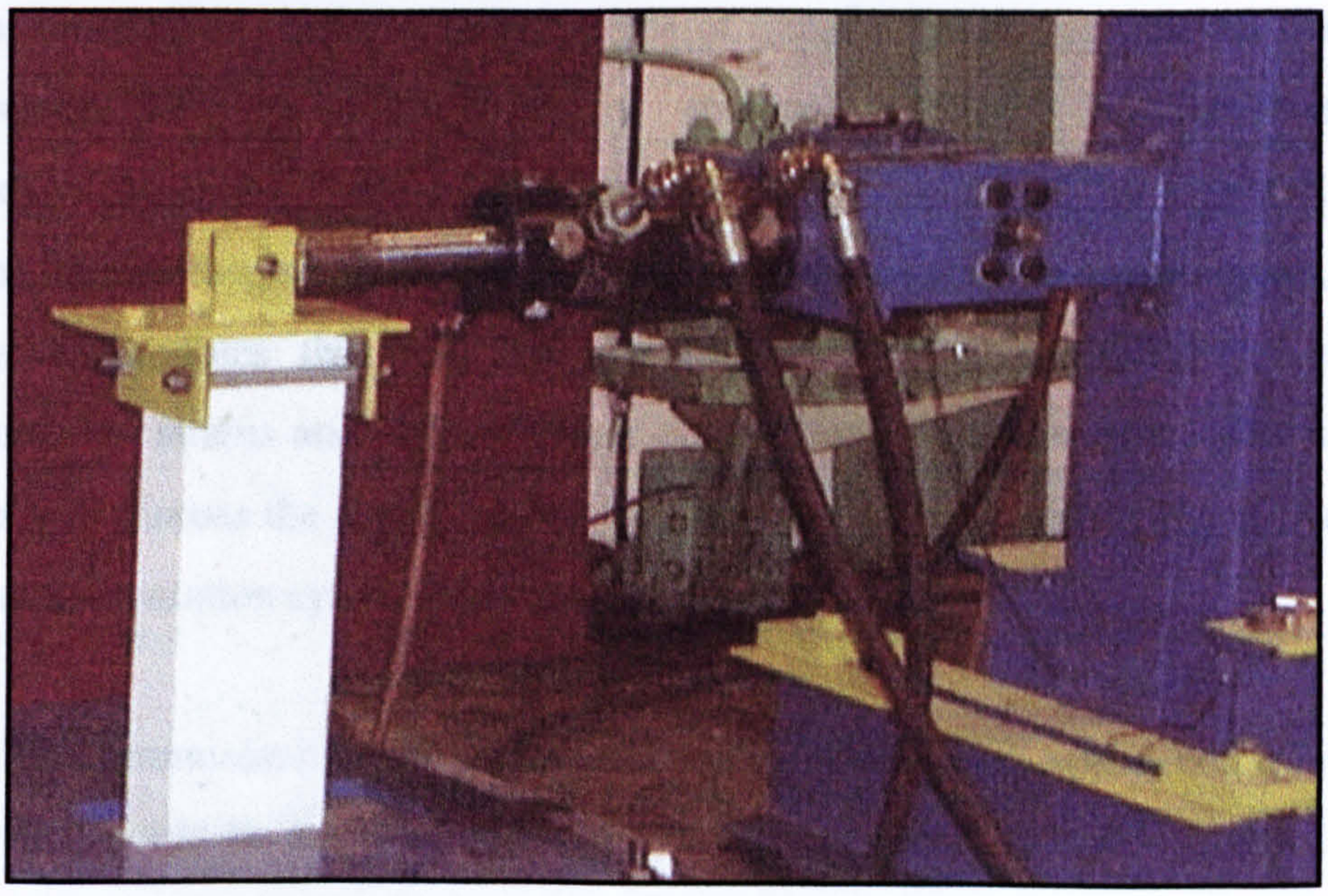


Fig. 3.15 Actuator arm and fixings.

3.4 COMMUNICATIONS

The communications system is concerned with the devices required to obtain, transmit and interpret measurements and signals. The signals can be categorised into input and output signals, where the input signals are communicated from the instrumentation to the computer while the output are from the computer to the instrumentation. Essentially, the inputs are the displacement and force signals, in addition to non-essential signals for example for strain measurements, while the output is the signal to the servo valve. The input and output are elaborated on in section 5.3.1, while the following sections will refer to the instrumentation and how the signals are passed from one place to another.

3.4.1 Instrumentation

The instrumentation comprises the systems of transducers and peripherals that enable measurement of variables normally required for the implementation of pseudodynamic tests. These include the active variables that are essential for the implementation: displacement and force measurements and the valve signals, as well as passive variables that only provide further information about the structural behaviour, i.e. strains and possibly other displacement measurements. The following sections will discuss the actual method of obtaining and transferring the active signals in the implementation system described in this thesis.

All active communication took place through a single high-speed communications card, which could be read directly by the controller that was coded in the LabView environment, as detailed in Chapter VI. Although signal conditioning was required for some of the instrumentation, the interpretation of all the conditioned signals was done in the software controller. The communications card could read a number of channels simultaneously, and presented no limitation to the communication speed.

3.4.1.1 Force measurements

The force measurements in this SDOF implementation represent a measure of the force exerted by the hydraulic actuator and correspond to the restoring force offered

by the specimen structure resisting displacements in the corresponding degree of freedom. In all the tests described here, the only degree of freedom is horizontal displacement at the top of the column.

The most direct method of obtaining a measure of the force acting is by using a load cell placed in series between the actuator and the specimen. This kind of system was therefore employed with the tests on the steel column. Here, a butterfly type load cell was used, as shown in figure 3.16. This load cell enabled accurate force measurements even under small loads. The signal from the load cell required amplification before it could be read by the computer, so an amplifier card was used to condition the signal.

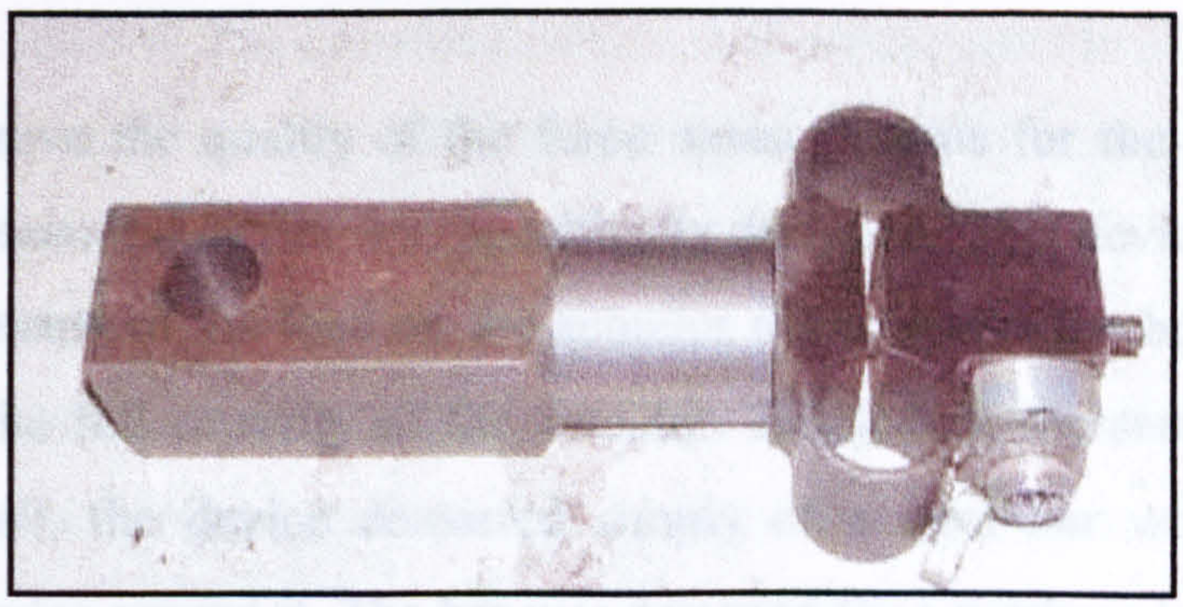


Fig. 3.16 Butterfly type load cell with connector

The range of the load cell was ±10kN, but for the tests on the steel column, only a maximum force of ±5kN was required. During calibration, the amplifying circuit was set such that ±10V (maximum voltage for communication card) corresponded to ±5kN. This maximised the accuracy of the force measurement, as the relative magnitude of electrical noise was minimised. The force measurements could be obtained to an accuracy of 5N, or 0.1% error, with this system.

Regrettably, the butterfly type load cell could not be used for the tests on the reinforced concrete column. This was because the forces developed during these tests would be significantly beyond the capacity of the cell. Two alternative methods were therefore used to obtain the force measurements, discussed below.

The pump and valve system in the laboratory also included a pressure transducer, measuring the relative pressure between the two output ports on the valve. This signal could be read directly by the communication card and interpreted by the controller. Unfortunately, this signal did not represent an equally accurate measurement as the load cell. The electrical noise was somewhat greater, creating a random error of the order of a maximum of ±500N, or in this case around 1-1.5% of the maximum force. A more significant problem was the fact that this pressure measurement did not take the internal friction of the actuator into account. This created a large drop in force each time the actuator changed direction, effectively opening up the hysteretic force-displacement loops creating artificial dissipation. This inaccuracy was critical during low-level tests, as the error could reach 20-30% of the force. During full-scale tests, were the full force range was employed, satisfactory results could still be obtained.

In order to improve the quality of the force measurements for the tests under high forces, a load-measuring device was specifically designed. This device was created to enable measurements of the load on the actuator directly, like the butterfly type load cell, but up to the full capacity of the actuator. Employing the same principle as a standard load cell, the device consisted simply of a steel bar with strain gauges forming a full bridge around it. The bar was designed for a working load of 50kN, and had a diameter of 25mm and an overall length of 150mm. The working load should result in strains of around 4.8E-04, enabling force to be measured to between 10N and 100N within the full ±50kN range. A picture of the bar type load cell placed in series with the actuator and specimen can be seen in figure 3.17.

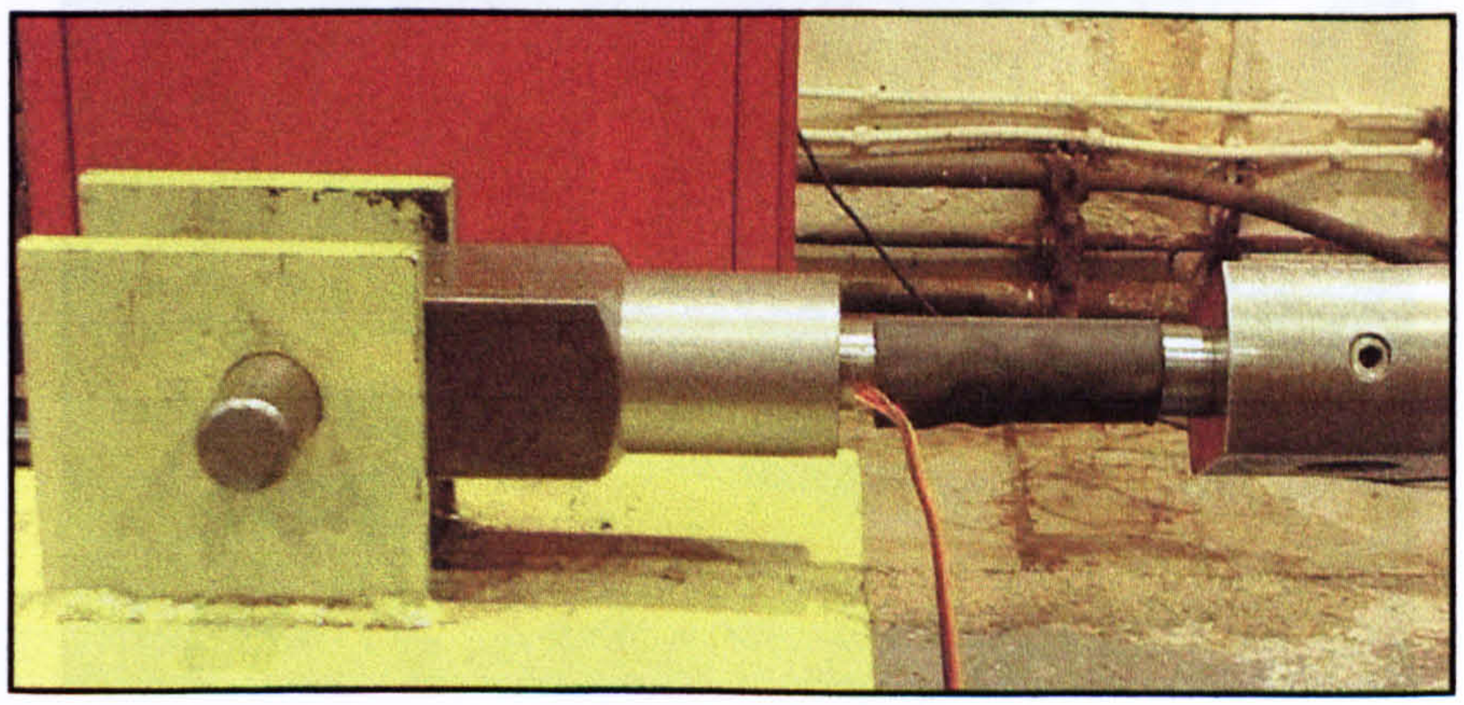


Fig. 3.17 Bar type load cell specifically designed for tests on concrete specimen

3.4.1.2 Displacement measurements

During the tests on both the reinforced concrete and steel columns, the active displacement measurements were taken through an LVDT (linear voltage differential transducer) internal to the actuator. This was the most practical place to position the transducer and, as the reaction rig was so stiff, no measurable errors would be introduced.

The total range of the LVDT was ±60mm, so the maximum displacements of ±50mm should lie within a fairly linear range of the LVDT. However, during calibration, it was found that non-linearities in the LVDT introduced errors of up to 2.5mm. Through software compensation, this error was reduced to a maximum of 100μm, as detailed in section 5.3.6. The transducer signal was conditioned to a DC signal within the ±10V range used by the communication card with an appropriate signal-conditioning card.

When carrying out the "snap-back" tests described in Chapter VII, an external transducer set-up was required. This was to allow the column to oscillate freely in real-time without any resistance. In this case, a similar LVDT was mounted outside the actuator, with the sprung rod pushing against a small, lubricated Plexiglas plate to minimise friction, as shown in figure 3.18.

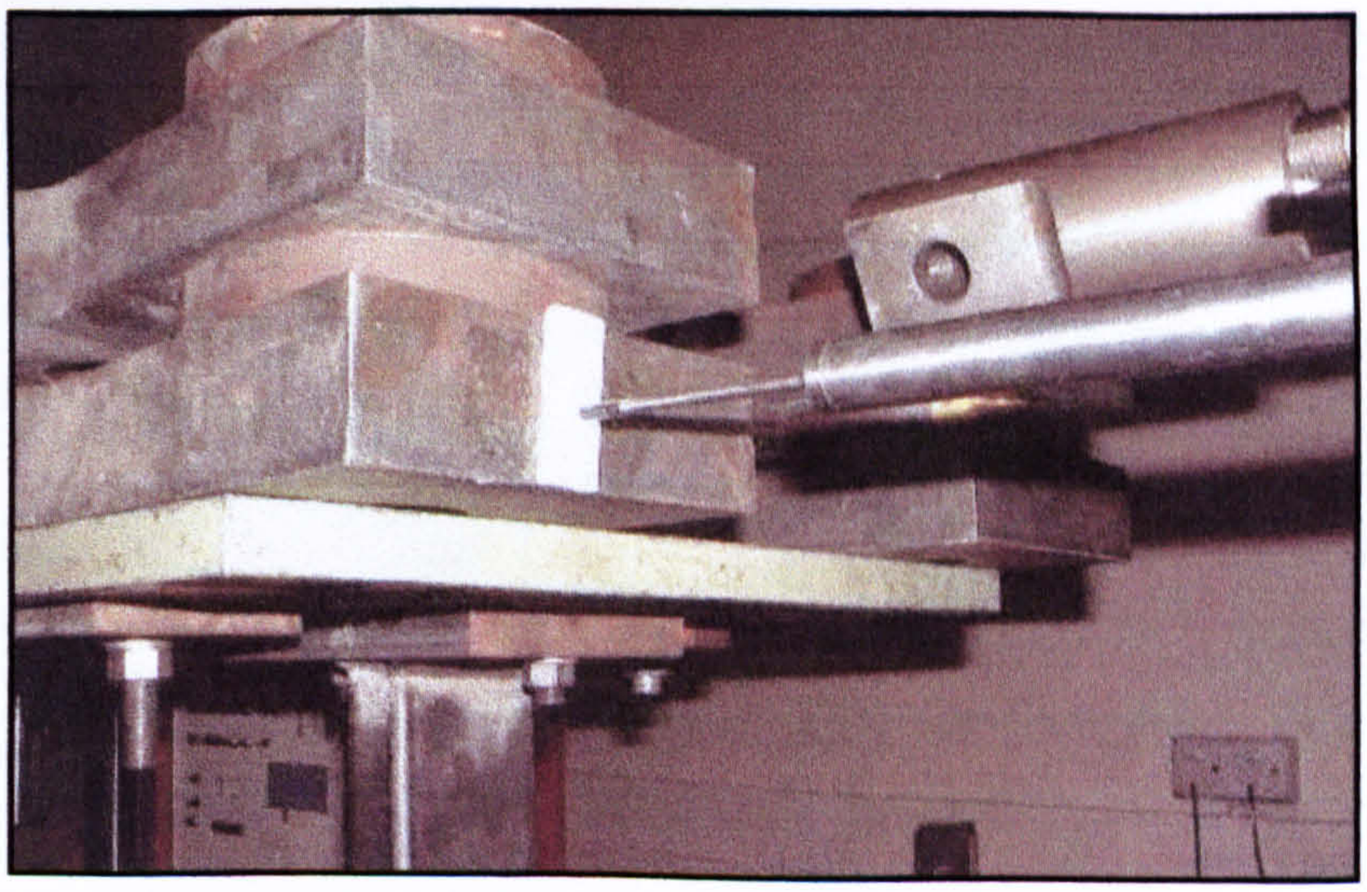


Fig 3.18 LVDT positioning during "snap-back" tests

3.4.1.3 Servo valve signal

The valve signal is generated within the controller running on the PC as detailed in Chapters V and VI. The signal specified by the algorithm is then created electronically in the communication card and sent directly to the valve. Although the valve technically operates with a current, the resistance within the valve is linear enough to simply specify a DC voltage. As this resistance is fairly small, the voltage range used by the valve is narrow, spanning typically only between 0.5 and -0.5V.

Chapter IV:

THE NEWMARK IMPLICIT – INTEGRAL FORM

A novel integral form time-integration algorithm for pseudodynamic testing is proposed, based on the Newmark Implicit algorithm. The scheme builds on the recently published integral form of the Newmark Explicit algorithm (Chang *et al.* 1998), which exhibits improved abilities to deal with rapidly varying loads and stiffness properties during pseudodynamic testing, but displays some numerical damping and only conditional stability. The enhancement is based on the inclusion of an additional term in the displacement predictor, which not only renders the algorithm more consistent, but it also eliminates numerical damping and makes the algorithm unconditionally stable. Under non-linear conditions, both the explicit and implicit forms of the integral form algorithm require a substitute for the tangent stiffness, and the sensitivity of the methods to the choice of this has been investigated. It is found that the proposed implicit method displays significantly less error in terms of periodicity and amplitude. An evaluation of the performance of the algorithm under nonlinear stiffness conditions with rapidly varying loads is included. Lastly, a proposal to an implementation system is presented. The main contributions in this chapter have been published in Engineering Computations (Algaard *et al.* 2001b).

4.1 INTRODUCTION

Several time stepping algorithms have been proposed for application in pseudodynamic testing (Bursi & Shing 1996), (Combescure & Pegon 1997), (Chang 1997). The majority of these are explicit due to the fact that the non-linear structural restoring forces at the end of any time step are unknown and displacement iterations in pseudodynamic tests are undesirable as these might result in partial unloading (Shing & Vannan 1990). Although implicit methods have the advantage of being unconditionally stable, the duration of the time steps still have to be limited for accuracy purposes, due to rapid changes in both loading and stiffness and linearisation errors in general. However, the so-called *integral form* of the Newmark Explicit method, proposed by Chang *et al.* (1998), relies on integrating the second-order equation of motion (4.1) with respect to time and it is argued that this method exhibits improved abilities to model rapidly varying loads and stiffness as well as improved error propagation characteristics.

$$M\Delta\frac{d^2x}{dt^2}+C\Delta\frac{dx}{dt}+\Delta r(x)=\Delta f \tag{4.1}$$

Implementation of the integral form into the pseudodynamic testing framework is however complicated by the fact that certain stiffness related terms become implicit. The possibility of enhancing Chang's formulation of the integral form by modifying some of its aspects is considered in this chapter.

4.2 NEWMARK EXPLICIT – INTEGRAL FORM

The integral form of the Newmark explicit method was initially suggested by Chang *et al.* in 1998, by integrating the equation of motion (4.1) in its incremental form once with respect to time. It is argued that such a form is better suited for systems with rapidly varying excitation force and rapidly varying level of non-linearity of the restoring force. Integrating equation (4.1) yields

$$M\Delta\frac{dx}{dt}+C\Delta x+\Delta\int r(x)dt=\Delta\int f dt \tag{4.2}$$

where Δ is the change during a time step. The improved accuracy originates from the fact that by performing the integration, equilibrium is satisfied over the duration of time step, rather than at its start or at its end. The solution involves utilisation of the time integral of the force for each step, which can be found reasonably accurately through some simple numerical integration and sub-stepping.

Linearisation of the time-force integral when sampling the excitation accelerogram at large time step intervals may lead to significant inaccuracies. While applied accelerograms typically have sampling points considerably closer spaced than integration steps, and restoring forces may be measured practically continuously during displacement of the specimen, there is clearly an opportunity to obtain accurate estimations of the time-force areas concerned. An example of how the variation in ground acceleration may be poorly linearised can be seen in figure 4.1 below. If the time step size employed is 0.2 seconds, and we consider the step between 5.0s and

5.2s, it is clear that the linearised acceleration over that step equals zero. The red line indicates the linearised accelerogram. On the other hand, a good estimate of the area can be found if a few sampling points are used in between and numerical integration like the trapezoidal rule or Simpson's rule.

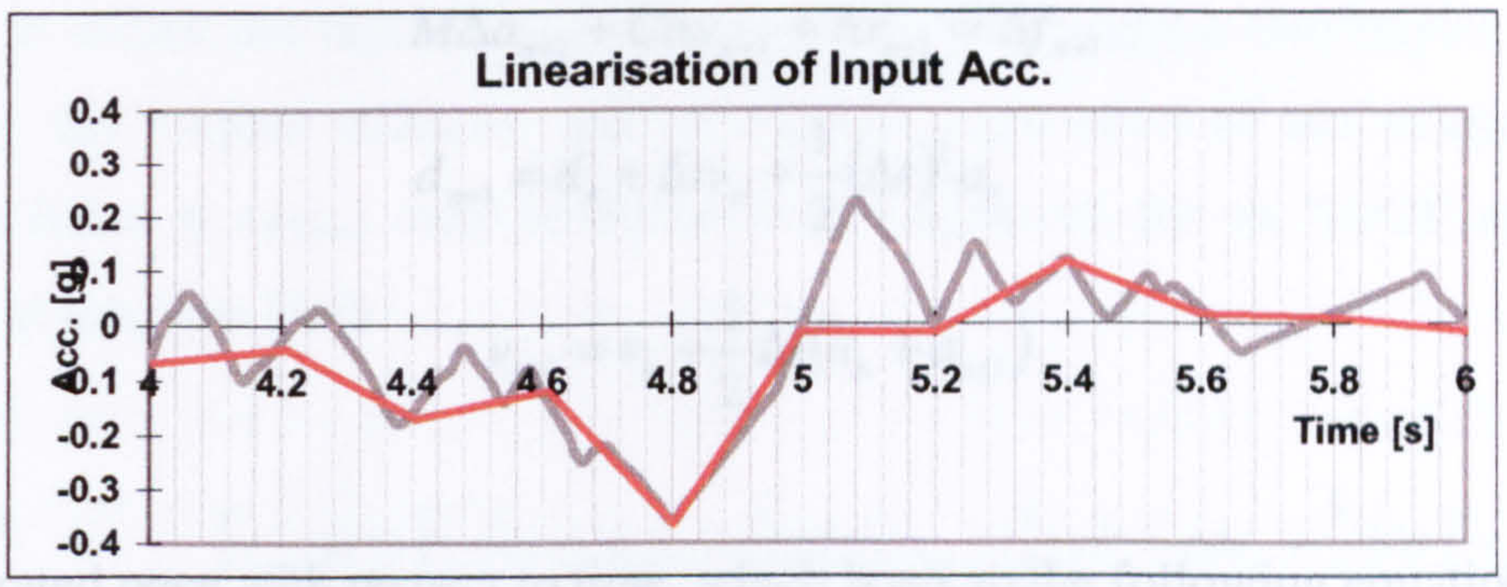


Fig. 4.1 Ground motion accelerogram

Similarly, for the restoring force, a linearised relationship between the start and the end values may very poorly represent the actual variation of the force over the time step, figure 4.2. Clearly, by linearising between the start and the end points underestimates the restoring force during loading and overestimates it during unloading – this evidently adds energy to the system and could potentially lead to an instability. While the error in the linearisation of the excitation force is of a random nature, it should be noted that the error associated with restoring force is systematic and cumulative.

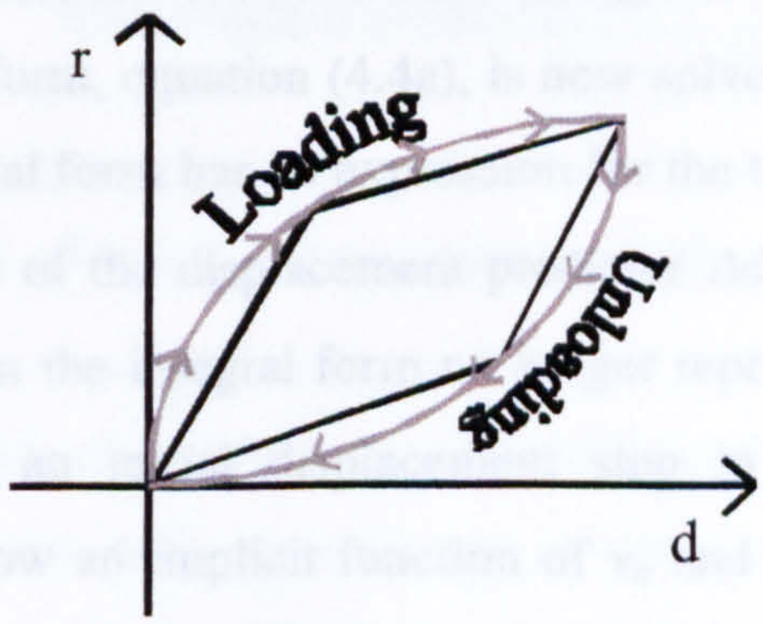


Fig. 4.2 Cumulative error as a result of linearisation of the restoring force

The applicability and the advantages of implementing the integral form method into pseudodynamic testing have been thoroughly investigated in Chang *et al.* (1998). As mentioned, the theory builds on integrating incremental equations in the explicit

format of the Newmark method ($\beta=0$ in the Newmark implicit relation $d_{n+1} = d_n + \Delta t v_n + \left(\frac{1}{2} - \beta\right)(\Delta t)^2 a_n + \beta(\Delta t)^2 a_{n+1}$, i.e. $d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}(\Delta t)^2 a_n$)

The Newmark Explicit equations in incremental form:

$$M\Delta a_{n+1} + C\Delta v_{n+1} + \Delta r_{n+1} = \Delta f_{n+1}$$

$$d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}(\Delta t)^2 a_n \tag{4.3}$$

$$v_{n+1} = v_n + \frac{1}{2}\Delta t(a_n + a_{n+1})$$

are integrated once with respect to time, which leads to the following equations:

$$M\Delta v_{n+1} + C\Delta d_{n+1} + \Delta\int r_{n+1}dt = \Delta\int f_{n+1}dt \tag{4.4a}$$

$$\int d_{n+1}dt = \int d_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n \tag{4.4b}$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t(v_n + v_{n+1}) \tag{4.4c}$$

where Δt is the time step duration, d and v the displacement and velocity, respectively, and Δ indicates the change over one time step. Whereas in the usual Newmark explicit format the equations (4.3) are solved for the change in acceleration, the equation of motion in the integral form, equation (4.4a), is now solved for the *change in velocity.* Additionally, the integral form has an expression for the time integral of displacement (equation 4.4b) instead of the displacement predictor Δd_{n+1} in the usual form. More importantly, the term in the integral form no longer represents an explicit prediction that may be used as an initial displacement step in pseudodynamic tests. The displacement step is now an implicit function of v_n and v_{n+1} and can be found from equation (4.4c), which in turn requires the solution of equation (4.4a) to obtain the velocity at the end of the time step, v_{n+1}. In effect, the action of integrating the set of equations has rendered the method *implicit* in the sense that the predictor displacement cannot be deduced directly any more. The integral form algorithm also requires an assessment of the integral of the restoring force before the displacement

predictor can be calculated. Such an estimate enables the solution for Δv_{n+1} to be found which in turn produces an explicit predictor for the displacement, equation (4.4c), which is needed for the pseudodynamic implementation. The restoring force, and its time integral are non-linear functions of displacement and can no longer be obtained directly, as no predictor displacement step exists to be imposed. In order to be able to utilise the algorithm, Chang *et al.* (1998) suggest multiplying equation (4.4b) by the tangent stiffness, and an explicit expression of the integral of the restoring force at $t=t_{n+1}$ may be found (here expressed for an SDOF system) as outlined in equation (4.5).

$$\int r_{n+1}dt = \int r_n dt + \Delta t k d_n + \frac{k}{2}(\Delta t)^2 v_n = \int r_n dt + \Delta t r_n + \frac{k}{2}(\Delta t)^2 v_n \qquad (4.5)$$

The physical interpretation of the above expression can be seen from the graph of the restoring force versus time, Fig 4.3, where $\int r_n dt$ indicates the restoring force-time area at a given time t, while the sum of the two remaining terms represents the projected trapezoidal area assuming that a constant velocity exists until the end of the step.

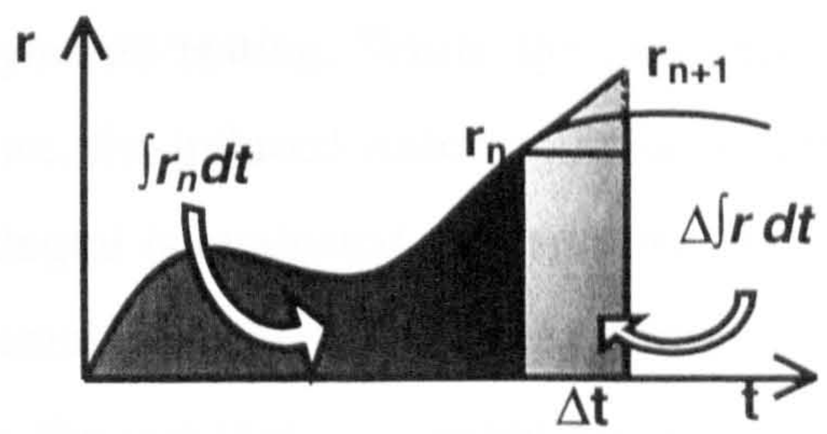


Fig 4.3. Approximation of $\Delta \int r_{n+1} dt$

Such a procedure tentatively assumes that the tangent stiffness is known or may be obtained somehow, which will normally not be the case in pseudodynamic testing (Chang *et al.* 1998). Only for the SDOF and for certain simple MDOF structures, may the stiffness matrix be computed from experimental data, and then only once a time step has been completed. To overcome the problem of the unknown tangent stiffness matrix, Chang *et al.* (1998) suggested replacing it with the initial stiffness term. The error involved is not large as the tangent stiffness is required only in the second order term on the right hand side of equation (4.5) (Chang *et al.* 1998). In any case, once an

expression for the integral of the restoring force exists, Chang *et al.* (1998) suggest a solution procedure where they solve for Δv_{n+1} from equation (4.4a) by substituting equation (4.4c) for d_{n+1}. When considering a SDOF system, the velocity change can be expressed as:

$$\Delta v_{n+1} = \left(m + \frac{\Delta t}{2} c \right)^{-1} \left(\Delta \int f_{n+1} dt - c \Delta t v_n - \Delta t r_n - \frac{k_0}{2} (\Delta t)^2 v_n \right) \tag{4.6}$$

where k_0 is the initial stiffness in place of the tangent stiffness term. By substituting this result back into equation (4.4c), a prediction for the change in displacement can finally be expressed as follows:

$$\Delta d_{n+1} = \Delta t v_n + \frac{\Delta t}{2} \left(m + \frac{\Delta t}{2} c \right)^{-1} \left(\Delta \int f_{n+1} dt - c \Delta t v_n - \Delta t r_n - \frac{k_0}{2} (\Delta t)^2 v_n \right) \tag{4.7}$$

which, when added onto the previous displacement value, furnishes an explicit displacement predictor to be applied in pseudodynamic tests similarly to any other displacement predictor. However, the method now departs from the traditional procedure in pseudodynamic testing. While the predictor displacement step is being imposed on the structure, the induced restoring force is continuously measured and its corresponding time integral is evaluated numerically. At the end of the step, this will represent an experimentally evaluated change in the time integral of the restoring force, the same term as the one that was earlier estimated in equation (4.5). In general, due to the material non-linearity, the restoring force will not follow the linear extrapolation as predicted, and the change in the time integral of this restoring force will in reality typically be smaller than estimated. For this reason, Chang's algorithm then recalculates Δv_{n+1}, based on the measured time integral of the restoring force. At this stage, Δd_{n+1} may or may not be recalculated, based on the updated Δv_{n+1} in equation (4.4c). Irrespective of whether the displacement increment Δd_{n+1} is then recalculated, there is a numerical damping present, which may be negative or positive depending on which reference values for the restoring force are used at the beginning of the displacement increment. The effect will be present irrespective of whether the tangential or initial stiffness term is used in equation (4.5).

4.3 NEWMARK IMPLICIT – INTEGRAL FORM

An inconsistency exists in the integral form of the Newmark Explicit algorithm, when recalculating Δv_{n+1} is based on the updated $\Delta\int r_{n+1}dt$ (Algaard *et al.* 2000). Δv_{n+1} indeed has to be recalculated; otherwise the information about the experimentally measured restoring forces is never taken into account. The predictor Δd_{n+1} is based on an estimation of Δv_{n+1}, which in turn builds on an approximation of the restoring force as expressed in equation (4.5). Once the integral of the restoring force has been obtained, the integrated equation of motion, equation (4.4a), can be applied.

Expanding equation (4.4c) yields

$$d_{n+1} = d_n + \frac{1}{2}\Delta t(v_n + v_{n+1}) = d_n + \frac{1}{2}\Delta t(v_n + v_n + \Delta v_{n+1}) = d_n + \Delta t v_n + \frac{1}{2}\Delta t\Delta v_{n+1} \quad (4.8)$$

Furthermore, equation (4.4a) assumes the integral of the restoring force over that time step, $\Delta\int r_{n+1}dt$, to be determined by computing the time-force area numerically over the step, as indicated in Fig 4.3. Assuming constant stiffness for simplicity, the exact expression for $\Delta\int r_{n+1}dt$ will be as follows

$$\Delta\int r_{n+1}dt = \int r_{n+1}dt - \int r_n dt = \Delta t k\left(\frac{d_n + d_{n+1}}{2}\right) \quad (4.9)$$

Equation (4.8) can now be substituted for d_{n+1} in equation (4.9) to yield the following

$$\Delta\int r_{n+1}dt = \Delta t k\left(\frac{d_n + d_n + \Delta t v_n + 1/2\,\Delta t\Delta v_{n+1}}{2}\right) \quad (4.10)$$

which can be further manipulated into

$$\Delta\int r_{n+1}dt = \Delta t k d_n + \frac{1}{2}\Delta t^2 k v_n + \frac{1}{4}\Delta t^2 k\Delta v_{n+1} \quad (4.11)$$

When comparing equation (4.11) with equation (4.5) it is clear that equation (4.11) now contains one additional term, $1/4\,\Delta t^2 k \Delta v_{n+1}$. This term corresponds to the term that is omitted in the standard Implicit Newmark algorithm to render it explicit (i.e. $\beta=0$), and the omission of this term is the cause of the numerical damping invariably present in the integral form of the algorithm. However, omitting the equivalent term in the integral form of the method does not render the method explicit – in fact its omission has no bearing on the nature of the algorithm. The integral form of the Newmark algorithm has been made explicit through an estimation of the time integral of the restoring force, which enables the calculation of Δv_{n+1} followed by Δd_{n+1}. At this point it becomes clear that there is no reason why the seemingly implicit additional term in equation (4.11) cannot be included in the estimation of the time integral of restoring force, as the implicit variable is the actual unknown which the expression is trying to represent. The situation is clarified through the following argument, where the equation (4.11) has been substituted into equation (4.4a) and solved for Δv_{n+1} to yield an alternative expression for equation (4.6).

$$\Delta v_{n+1} = \left(m + \frac{\Delta t}{2} c\right)^{-1} \left(\Delta \int f_{n+1} dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n - \frac{1}{4}\Delta t^2 k_0 \Delta v_{n+1}\right) \quad (4.12)$$

The unknown, Δv_{n+1}, is present on both sides of the equation, but through further rearrangement:

$$\Delta v_{n+1} + \left(m + \frac{\Delta t}{2} c\right)^{-1} \frac{1}{4}\Delta t^2 k_0 \Delta v_{n+1} = \left(m + \frac{\Delta t}{2} c\right)^{-1} \left(\Delta \int f_{n+1} dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)$$

$$\left(1 + \left(m + \frac{\Delta t}{2} c\right)^{-1} \frac{1}{4}\Delta t^2 k_0\right) \Delta v_{n+1} = \left(m + \frac{\Delta t}{2} c\right)^{-1} \left(\Delta \int f_{n+1} dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)$$

an explicit expression for the velocity can finally be found.

$$\Delta v_{n+1} = \frac{\left(m+\frac{\Delta t}{2}c\right)^{-1}\left(\Delta\int f_{n+1}dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)}{1+\left(m+\frac{\Delta t}{2}c\right)^{-1}\frac{1}{4}\Delta t^2 k_0} \tag{4.13}$$

This equation may now be substituted directly into equation (4.8) and rearranged to obtain a new explicit expression for the displacement predictor Δd_{n+1}.

$$\Delta d_{n+1} = \Delta t v_n + \frac{\frac{\Delta t}{2}\left(m+\frac{\Delta t}{2}c\right)^{-1}\left(\Delta\int f_{n+1}dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)}{1+\left(m+\frac{\Delta t}{2}c\right)^{-1}\frac{1}{4}(\Delta t)^2 k_0} \tag{4.14}$$

By using equation (4.14) rather than equation (4.7) as the displacement predictor, the time stepping algorithm has become implicit. To be exact, the algorithm still requires a value of the tangent stiffness (k_0 terms in equation 4.14), however this is also the case with the earlier explicit version of the algorithm and other implicit algorithms (Combescure & Pegon 1997). The same term is now merely present in one additional place, and as with the explicit version, the initial stiffness, or an estimate of the tangent stiffness, can be used.

Principal differences between the two algorithms are summarised below.

	Chang's integral form Newmark explicit algorithm
predictor d_{n+1}	$\Delta t v_n + \frac{\Delta t}{2}\left(m+\frac{\Delta t}{2}c\right)^{-1}\left(\Delta\int f_{n+1}dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)$
$\int d_{n+1}dt$	$\int d_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$
	Proposed integral form Newmark implicit algorithm
predictor d_{n+1}	$\Delta t v_n + \frac{\frac{\Delta t}{2}\left(m+\frac{\Delta t}{2}c\right)^{-1}\left(\Delta\int f_{n+1}dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)}{1+\left(m+\frac{\Delta t}{2}c\right)^{-1}\frac{1}{4}(\Delta t)^2 k_0}$
$\int d_{n+1}dt$	$\int d_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n + \frac{1}{4}(\Delta t)^2 \Delta v_{n+1}$

4.4 STABILITY AND DISSIPATION PROPERTIES

The stability properties of an integration algorithm are studied by considering the spectral properties of its recursive amplification matrix. Considering the state vector of the system at time $t=t_n$, the integral form of the numerical time integrator algorithms yields

$$X_n = \begin{bmatrix} \int d_n dt \\ \Delta t d_n \\ \Delta t^2 v_n \end{bmatrix} \tag{4.15}$$

For stability purposes, one can ignore the external load vector and damping forces, thus the recurrent relationship between the state vector at $t=t_n$ and $t=t_{n+1}$ can be expressed as (Bathe & Wilson 1976):

$$X_{n+1} = [A] X_n \tag{4.16}$$

where $[A]$ is the recursive amplification matrix.

4.4.1 Stability of the Newmark Explicit – Integral Form algorithm

Considering first the explicit, integral form algorithm, equations (4.4) can for a SDOF system be expressed in terms of the variables of the state vector as

$$\int d_{n+1} dt = \int d_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t (v_n + v_{n+1}) \tag{4.17}$$

$$\Delta v_{n+1} = m^{-1}\left[-\int_n^{n+1} r_{n+1} dt \right]$$

Depending on the precise implementation of the algorithm, i.e. whether d_{n+1} is updated following the recalculation of v_{n+1} or not, the exact expressions for d_{n+1} and $\int r_{n+1} dt$ will differ.

Assuming initially that d_{n+1} is not recalculated, the predicted d_{n+1} remains and the term $\int r_{n+1}dt$ will be a function of the restoring force both at the start of and at the end of the predicted step, yielding the second of equations (4.18). d_{n+1} will thus no longer be represented by the implicit expression in equation (4.17), rather by a simplification of equation (4.7) containing only the terms relevant for stability analyses. $\int r_{n+1}dt$ may be defined in terms of d_{n+1} or by the expression for the prediction step, however this will in the end lead to the same amplification matrix. Assuming constant stiffness for simplicity, $\int r_{n+1}dt$ may be expressed as $\frac{\Delta tk}{2}(d_n + d_{n+1})$, yielding the third equation of equations (4.18).

$$\begin{aligned}
\int r_{n+1}dt &= \int r_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n \\
d_{n+1} &= d_n + \Delta t v_n + \frac{1}{2}\Delta t m^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right) \\
v_{n+1} &= v_n + m^{-1}\left(-\frac{\Delta tk}{2}(d_n + d_{n+1})\right)
\end{aligned} \tag{4.18}$$

Multiplying the second and third equation by Δt and $(\Delta t)^2$, respectively, and substituting Ω^2 for $(\Delta t^2)k/m$, yields equations (4.19).

$$\begin{aligned}
\int r_{n+1}dt &= \int r_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n \\
\Delta t d_{n+1} &= \Delta t d_n + (\Delta t)^2 v_n - \frac{1}{2}\Omega^2 \Delta t d_n - \frac{1}{4}\Omega^2 (\Delta t)^2 v_n \\
(\Delta t)^2 v_{n+1} &= (\Delta t)^2 v_n - \Omega^2 \frac{\Delta t d_n}{2} - \Omega^2 \frac{\Delta t d_{n+1}}{2}
\end{aligned} \tag{4.19}$$

After sorting terms at $t=t_{n+1}$ and $t=t_n$ and expressing them in matrix form (Géradin & Rixen 1994), the amplification matrix is obtained as

$$A = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2} & 1-\frac{\Omega^2}{4} \\ 0 & -\Omega^2 + \frac{\Omega^4}{4} & 1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8} \end{bmatrix} \tag{4.20}$$

which clearly differs from the normal Newmark Explicit matrix (Shing & Mahin 1987). Stability of an algorithm is ensured when the spectral radius (modulus of highest eigenvalue) of the amplification matrix does not exceed unity (Golley & Amer 1999). In the above matrix, one eigenvalue will be equal to unity, while the other two will form a pair of complex conjugates. Corresponding moduli have been plotted below as a function of Ω in Fig 4.4. Both the expected stability limit of 2.0 and the existence of noticeable numerical damping can be seen for the values of Ω exceeding 0.5. The full derivation of the amplification matrix and analysis of the eigenvalues for this and with the alternative representation of $\int r_{n+1} dt$ in the Newmark Explicit – Integral Form algorithm, can be found in Appendices A1 and A2, respectively.

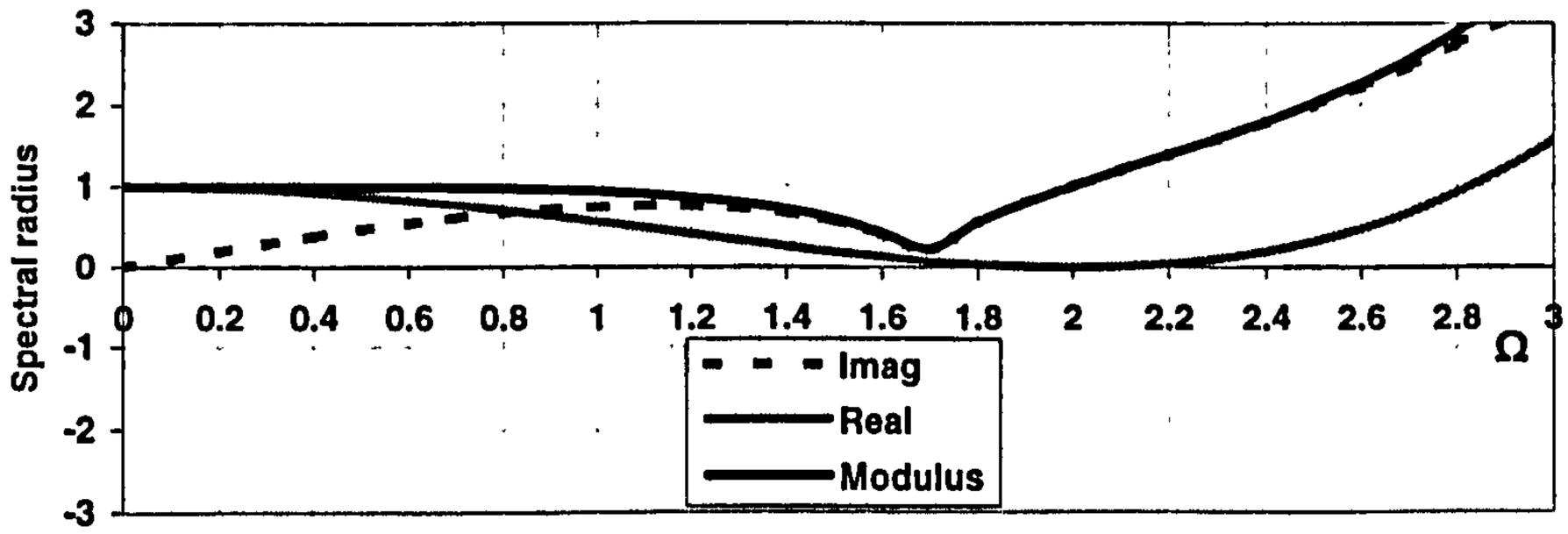


Fig. 4.4 Spectral radius vs. Ω of Newmark Explicit – integral form, without recalculation of d_{n+1}

If however the d_{n+1} is recalculated once the corrected v_{n+1} has been found, the situation is somewhat different. Equations (4.18) will be altered such that the displacement predictor is no longer present in the definition of d_{n+1}, but remains in the expression for v_{n+1}. This yields equations (4.21), which can be represented by the amplification matrix shown in equation (4.22), obtained the same way as equation (4.20).

$$\int r_{n+1} dt = \int r_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t v_n + \frac{1}{2}\Delta t v_{n+1} \tag{4.21}$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta t k}{2}\left(2d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)\right)\right)$$

$$A=\begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8} & 1-\frac{\Omega^2}{4}+\frac{\Omega^4}{16} \\ 0 & -\Omega^2+\frac{\Omega^4}{4} & 1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8} \end{bmatrix} \tag{4.22}$$

This algorithm exhibits the same stability and damping characteristics as the standard Newmark Explicit; perfect energy conservation up to the stability limit of 2.0. However, as the algorithm stands, it cannot be directly implemented into a pseudodynamic test. This is because r_n is in fact unknown at the start of the time step. The reason for this is that d_n was recalculated after the completion of the previous step, and the restoring force caused by it is thus unknown. The correct procedure would require the recalculated d_{n+1} to be imposed separately and the corresponding restoring force re-measured. Such a procedure would lead to a double step implementation, but employing iterations in an algorithm that is still only conditionally stable seems ineffective. The method can instead be implemented using the restoring force as measured at the end of the predictor step, before any update. Using this alternative method results in numerical damping and a reduced stability limit, but the appropriate amplification matrix for this algorithm cannot be formed. This is because a recurrent relationship between the state vector, equation (4.15), at time $t=t_n$ and $t=t_{n+1}$ is insufficient in describing a two-step method. It also requires representation of some variables at previous time steps, as indicated in Appendix B.

4.4.2 Stability of the Newmark Implicit – Integral Form algorithm

The effects of using the implicit version of the algorithm with the modified displacement predictor are substantial. Not only does the method avoid the numerical damping associated with the Newmark Explicit – integral form algorithm, but owing to the fact that the algorithm is now genuinely implicit, it also becomes unconditionally stable. This was initially noted through numerical experiments, but can also be confirmed analytically. By considering the expression for the time integral of displacement, the displacement and the velocity and using a similar procedure as with equation (4.19), it leads to the following equations:

$$\int d_{n+1}dt = \int d_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n - \beta \frac{\Omega^2}{1+\Omega^2/4}\Delta t d_n - \frac{1}{2}\beta \frac{\Omega^2}{1+\Omega^2/4}(\Delta t)^2 v_n$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n - \gamma \frac{\Omega^2}{1+\Omega^2/4}\Delta t d_n - \frac{1}{2}\gamma \frac{\Omega^2}{1+\Omega^2/4}(\Delta t)^2 v_n \qquad (4.23)$$

$$\Omega^2 \int d_{n+1}dt + (\Delta t)^2 v_{n+1} = \Omega^2 \int d_n dt + (\Delta t)^2 v_n$$

where β and γ are the parameters normally present in the Newmark algorithms related to the numerical integration of the remainders in the Taylor expansion. These typically take on the values of 0.25 and 0.5, respectively (Géradin & Rixen 1994). Expressing the above equations in a matrix form yields again the recursive amplification matrix of the integration operator

$$A = \begin{bmatrix} 1 & 1-\beta\frac{\Omega^2}{1+\Omega^2/4} & \frac{1}{2}-\frac{1}{2}\beta\frac{\Omega^2}{1+\Omega^2/4} \\ 0 & 1-\gamma\frac{\Omega^2}{1+\Omega^2/4} & 1-\frac{1}{2}\gamma\frac{\Omega^2}{1+\Omega^2/4} \\ 0 & -\Omega^2\left(1-\beta\frac{\Omega^2}{1+\Omega^2/4}\right) & 1-\Omega^2\left(\frac{1}{2}-\frac{1}{2}\beta\frac{\Omega^2}{1+\Omega^2/4}\right) \end{bmatrix} \qquad (4.24)$$

The complex expression for the eigenvalues of [A] can be simplified to

$$(1-\lambda)\left(\lambda^2 + \left\{\left(-2+\frac{1}{2}\Omega^2\right)\left(1-\frac{1}{4}\xi^2\right)\right\}\lambda + 1 + \frac{\Omega^2}{2} - \frac{\Omega^2\xi^2}{8} - \frac{1}{2}\xi^2\right) = 0 \qquad (4.25)$$

where λ are the eigenvalues and ξ^2 is expressed as

$$\xi^2 = \frac{\Omega^2}{1+\frac{1}{4}\Omega^2} \qquad (4.26)$$

Eliminating $\lambda_1 = 1$ leaves the remaining second order equation:

$$\lambda_{2,3} = \frac{\left(-2+\frac{\xi^2}{2}+\frac{\Omega^2}{2}-\frac{\Omega^2\xi^2}{8}\right) \pm \sqrt{-4\Omega^2+\frac{1}{4}\xi^4+\frac{1}{4}\Omega^4+\frac{3\xi^2\Omega^2}{2}-\frac{\Omega^4\xi^2}{8}-\frac{\Omega^2\xi^4}{8}+\frac{\Omega^4\xi^4}{64}}}{2}$$

By plotting the real and imaginary parts of the solution and computing the moduli, it can be seen that the moduli for $\lambda_{2,3}$ also equal unity for all Ω, Fig 4.5. This clearly yields the overall solution that $\rho(A)=1$ for all Δt, which implies unconditional stability and perfect energy conservation. The rather lengthy derivation of the amplification matrix and eigenvalues analysis of this method can be found in Appendix C.

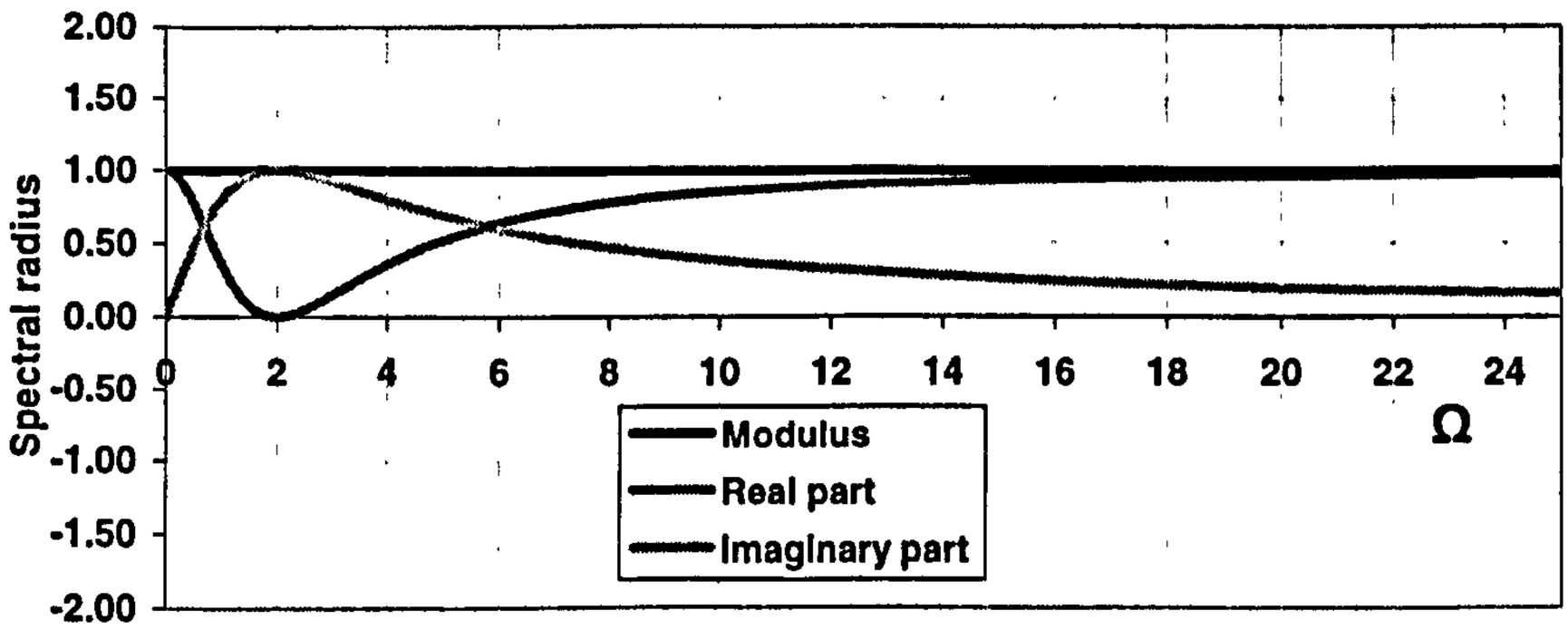


Fig. 4.5 Spectral radius vs. Ω of Newmark Implicit – Integral Form

The modifications carried out on the Newmark Explicit – Integral Form have been shown to eliminate the amplitude error of the algorithm and also improve the stability properties such that it is now unconditionally stable. As the algorithm is now implicit, the name Newmark Implicit – Integral Form seems appropriate.

4.5 IMPLEMENTATION

Most structures tested pseudodynamically will display considerable non-linear behaviour, typically of a strain softening nature. As mentioned earlier, and discussed in Chang *et al.* (1998), calculating the integral of the restoring force over each time step, rather than linearising between the start and end, should better capture the non-linearities in the stiffness. In order to calculate this integral numerically, the restoring forces generated should be measured a substantial number of times per time step.

Ideally, such measurements and subsequent numerical integration should be incorporated into the control algorithm to ensure optimal performance. A suggestion to how this can be implemented is shown in the semi-continuous implementation system developed, see chapters V and VI. The method relies on integration with an integral form algorithm and high-speed computations. The latter is ensured through a fully integrated control-time integration scheme that performs control iterations at a frequency of the order of 1kHz for a SDOF system. Assuming a 1 second control time for each step, 1000 force points are available for the restoring force integration. Such a number of sampling points will be more than sufficient to obtain an accurate computation of the time integral of the restoring force. The sampling points will however not be equally spaced, as the controller does not generate a linear time-displacement curve. On the other hand, as the time-integration algorithm does assume constant average velocity, the real-time points clearly cannot be used. These would in any case have to be scaled, as the time-integration time steps are typically an order of magnitude smaller than the real-time step. In order to obtain a correct force-time area, each of the, say 1000 displacement points must be given a corresponding time point. These points can be defined by examining the proportion of the completed displacement step, and by assuming constant velocity this will correspond to the proportion of the completed time, as demonstrated in figure 4.7.

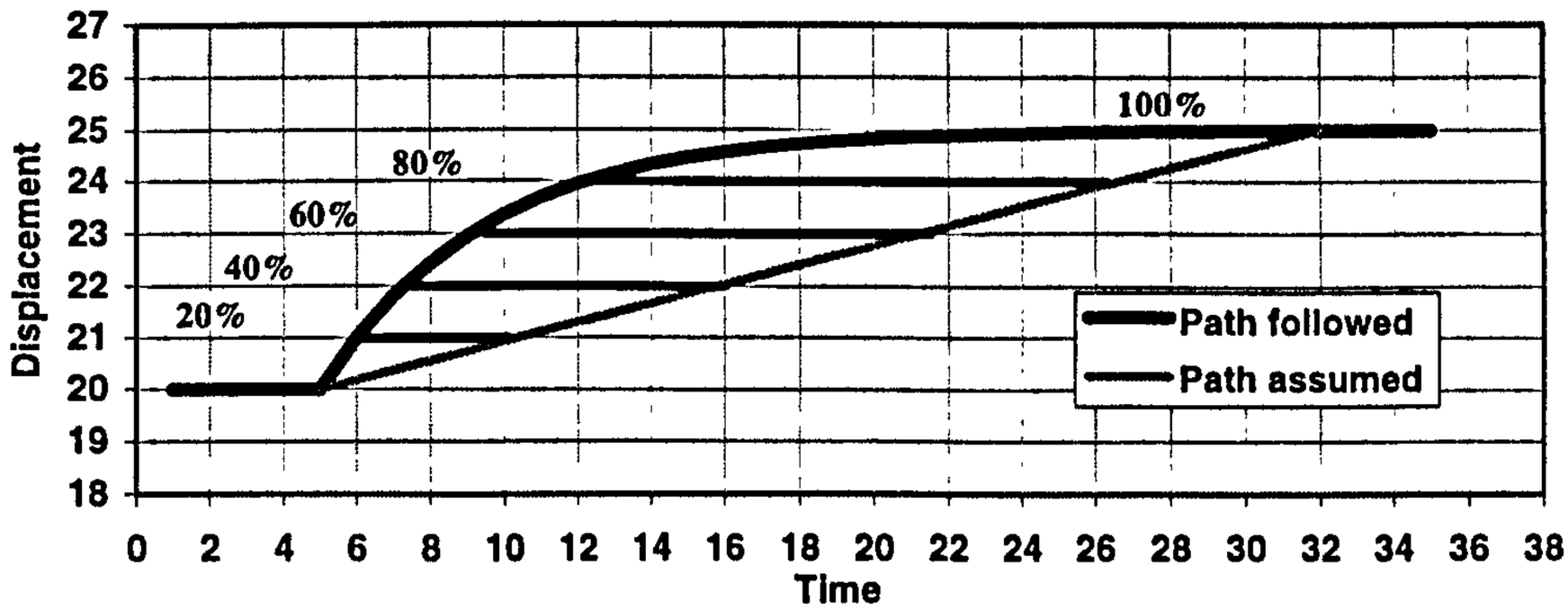


Fig. 4.7 Equivalent time points using proportion of completed step.

The controller will typically create a velocity profile that is initially fairly steep to attempt rapid completion of the displacement step, but flattens out when approaching the target displacement to ensure minimisation of the overshoot. This implies that, for example, 80% of the displacement step is completed in just 7 of the total 27 time units, or 26% of the time. When computing the equivalent time point assuming

constant velocity, 80% of the time should have passed at 80% completion of the displacement, or in other words around 22 time units. Thus, for each control iteration, a time value is assigned corresponding to the proportion of the displacement step completed and the time integration step size.

The method initialises as normal in pseudodynamics by computing an explicit displacement step that is then passed to the controller. This then directs the hydraulic actuator such that it displaces the structure to the commanded position. However, during the actuator motion, restoring force measurements and integration of this takes place. This will be completed simultaneously with the controller part of the algorithm to ensure that the next displacement step can be calculated with a minimum delay. Effectively, the force measurements are finished by the time the actuator reaches the desired point, and calculation of the next displacement step can take place immediately.

The method also shows improved handling of general control errors. As force computations are completed on reaching the target displacement, any further actuator motion does not affect the calculations. Damage from a potential overshoot will be highly limited as load reversal is avoided. This can be done because the restoring force measurements are finished before the overshoot takes place, and load reversal is not required to regain correct displacements. Initial experimental tests suggest that errors caused by load reversal are unacceptable.

4.6 ACCURACY PROPERTIES

The Newmark family is, in general, exact for linear systems. This applies to both the standard explicit and implicit versions, which display no amplitude error for any time step size as long as stability is ensured. Period distortion will however be present, and of a magnitude increasing with increasing time step-vibration period ratio. Typically, for a $\Delta t/T$ ratio of 0.10, period elongation of around 3% should be anticipated (Wilson & Bathe 1976).

If the stiffness term in the dynamic system is of a non-linear nature, this will affect both stability and accuracy properties of the schemes. Analytical solutions can no longer be found, and stability cannot in principle be proven although unconditional stability may in practise be preserved. Furthermore, implementation of implicit and integral form schemes requires either iterations or alternative approximation methods.

As discussed previously in this chapter, both Chang's explicit integral form method and the author's implicit method rely on using an approximation of the tangent stiffness term in the displacement predictor. This approximation normally extends to using the initial stiffness, measured from an undamaged specimen. *Initial stiffness* will throughout this chapter refer to the approximation of the tangent stiffness required by the integral form algorithms, i.e. the k_o term in equation (4.14). In a SDOF system, the initial stiffness can be obtained simply by displacing the specimen within the linear range and measuring the restoring force this creates. This evaluation will be carried out prior to commencement of the actual test. In MDOF systems, implementation of the integral form algorithms becomes significantly more complex as computation of the displacement predictor requires solution of equation (4.14) on a matrix level. However, the principles remain the same. The initial stiffness must now represent the total stiffness acting on each degree of freedom, which may be contributed to by a number of discrete springs. This stiffness can be obtained by displacing one degree of freedom in turn while keeping the others fixed.

As long as the initial stiffness is the same or higher than the tangent stiffness, stability should in most cases be ensured. However, the choice of this term may affect the accuracy of a scheme. An evaluation of the effect of this term, and the accuracy properties of the integral form algorithms in general is therefore included.

4.6.1 Evaluation system

In order to investigate the sensitivity of the integral form methods to the choice of the initial stiffness, an evaluation scheme was devised. This would try and evaluate the importance of both periodicity and amplitude errors introduced with Chang's explicit and the author's implicit integral form methods.

Initial experimentation suggested that the stiffness of an undamaged reinforced concrete specimen was of the order of three times higher than that of a slightly damaged specimen. To ensure that the entire potential spectrum of stiffness degradation had been considered, it was decided to evaluate the effect of using ratios of initial stiffness to actual stiffness ranging from 1 to 10, where 1 implies the specimen retains full stiffness and the initial stiffness is a perfect approximation.

The sensitivity of the algorithms to the approximation of the tangent stiffness was evaluated in numerical simulations of a SDOF dynamic system comprising a mass of 48600kg and a stiffness of 1000kN/m, yielding a natural frequency with an oscillation period of approximately 1.4s. A constant actual stiffness was used, so the ratio of initial to actual stiffness was manipulated by varying the initial stiffness. The effect was investigated using four different time step sizes with both the explicit and implicit integral form algorithms. The time step sizes ranged from 0.004s to 0.16s, where for 0.004s the effect was expected to be practically negligible. A 10s response, following an initial displacement of 1m, was modelled in all cases.

4.6.2 Results

To exhibit the effect of using the initial stiffness as a crude approximation of the tangent stiffness, a series of displays have been opted for. These include representation of the responses obtained during variations of the time step size and stiffness ratio. To trace the effect of the variation in stiffness ratio on the periodicity error and amplitude decay more directly, these have been plotted for the relevant time step sizes. The results from simulations using the explicit and implicit algorithms, respectively, are presented below.

4.6.2.1 Newmark Explicit – Integral Form

As argued in the analytical assessment of the integral form algorithms, Chang's procedure can be interpreted in two different ways: with or without recalculation of d_{n+1}. It will be assumed in this section that d_{n+1} is not recalculated, as only this method of implementation can be directly applied to pseudodynamics in a single step manner. This procedure creates an amplification matrix whose spectral radius is less

than unity up to the stability limit of $\omega_n \Delta t$=2, as seen in figure 4.4. This implies numerical dissipation of energy, and the effect is most pronounced for $\omega_n \Delta t$ values between 1.0 and 1.7, i.e. for fairly large time steps. A somewhat damped response should therefore be expected even when using a prediction of the initial stiffness that equals the actual tangent stiffness of the system. Considering the response obtained with such stiffness, figure 4.8, the largest time step size, Δt=0.16s clearly produced significant dissipation. This step size corresponds to an $\omega_n \Delta t$ value of 0.72. When inspecting figure 4.4, it can be seen that this is only the onset of the damping, which increases considerably up to $\omega_n \Delta t$=1.7. Time step sizes any larger than this will evidently produce substantial damping, which will greatly affect the response. On the other hand, it is seen that smaller time steps, even Δt=0.08s, produce only negligible damping.

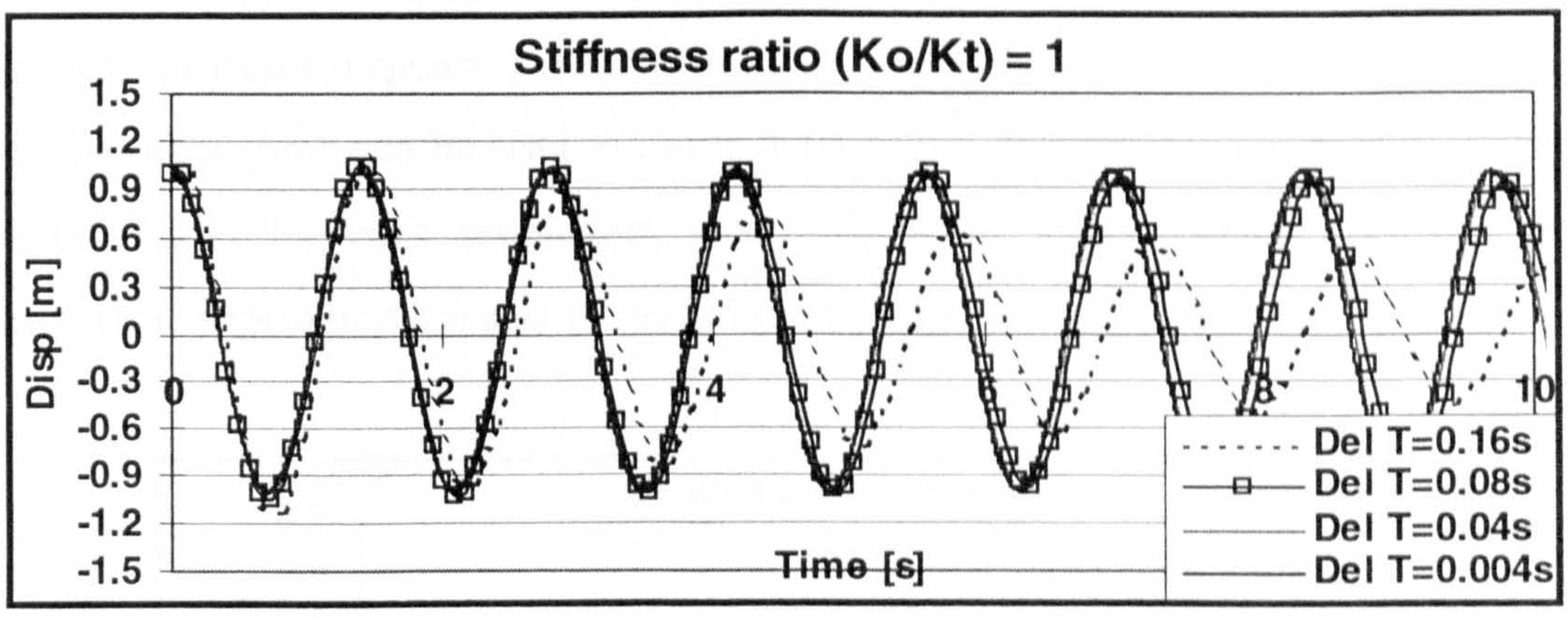


Fig. 4.8 Response obtained using explicit method with k_o/k_t ratio of 1

Considering now the response obtained using a large discrepancy between the assigned and actual stiffness, a ratio of 10, it is apparent that overall dissipation is higher, see figure 4.9. Another immediate observation is that with the largest time steps, Δt=0.16s, the solution is unstable even though the $\omega_n \Delta t$ value is well within the stability limit of 2. This demonstrates that stability is not necessarily ensured with initial stiffness (k_o) values equal to or greater than the tangent stiffness and may also indicate that the stability limit analytically proven under linear conditions does not necessarily apply to non-linear systems. In terms of damping, the response contains noticeable amplitude decay for the Δt=0.04s step size and significant for Δt=0.08s.

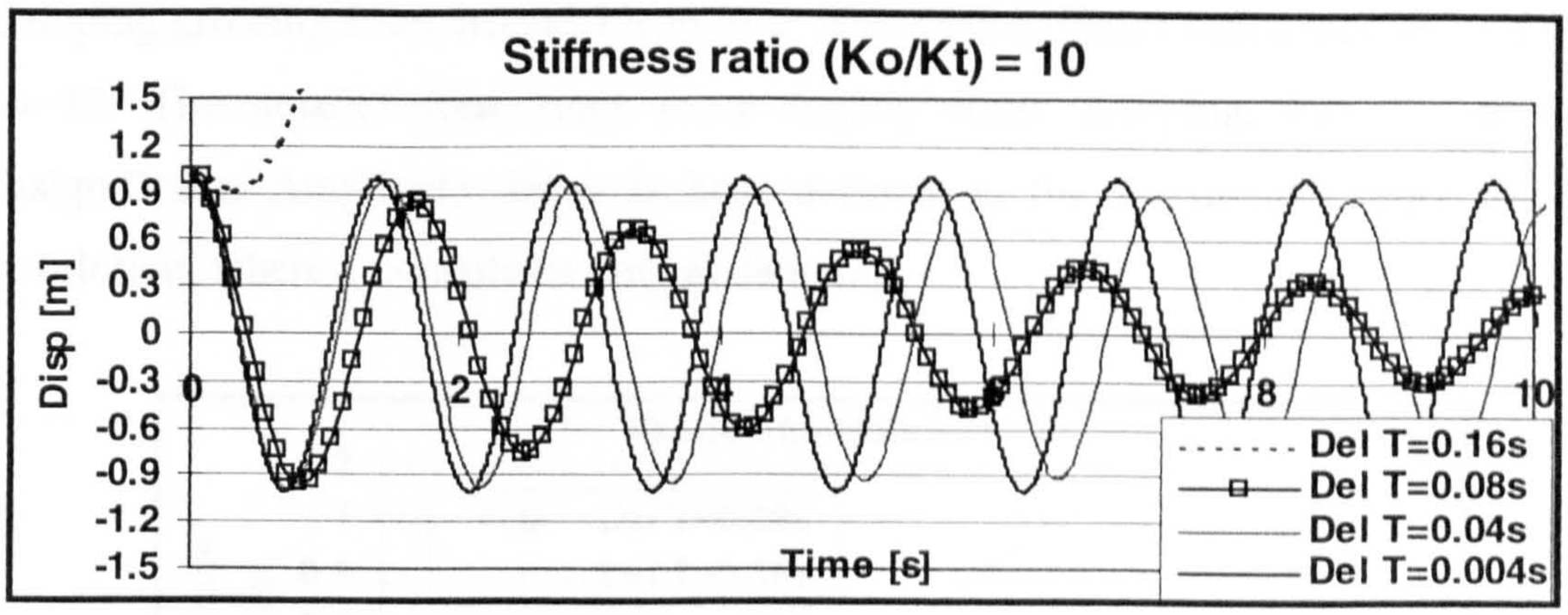


Fig. 4.9 Response obtained using explicit method with k_o/k_t ratio of 10

Simulations were carried out using initial stiffnesses of 1000, 2000, 3000, 5000, 7000 and 10 000kN/m (providing stiffness ratios of 1, 2, 3, 5, 7 and 10) and time step sizes of 0.004s, 0.04s, 0.08s and 0.16s. The smallest time step size, 0.004s, was assumed to provide an exact response for all stiffnesses, while the effect on the vibration period for the other three can be seen in figure 4.10 below. Period elongation is here defined as $(T_a\text{-}T_n)/T_n$ for each revolution, where T_a is the actual period observed in the numerical approximation and T_n the natural period.

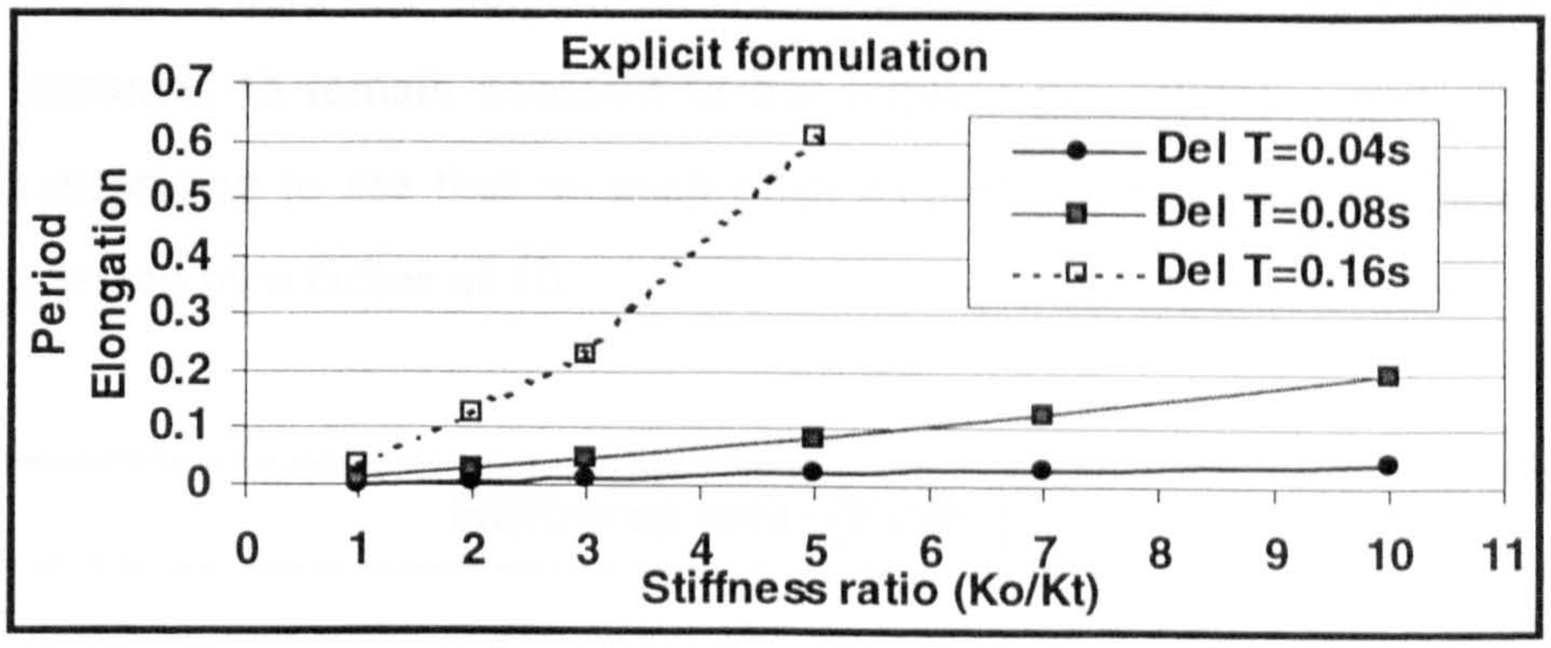


Fig. 4.10 Period elongation in explicit method as function of stiffness ratio

As substantial damping can in some cases be observed even with exact stiffness prediction; a plot of the energy dissipated in each revolution as a function of time step size and stiffness ratio has been provided as shown in figure 4.11. The largest time step size displays almost 10% damping with no stiffness discrepancy, while critical damping results from a stiffness ratio of 7. The 0.08s time step shows moderate

damping growing from around 1% to over 10% with stiffness ratios increasing from 1 to 10. The smaller time steps sizes display some damping, but this is fairly insignificant. Amplitude decay is here defined as the fractional energy loss per revolution, where 1 constitutes critical damping.

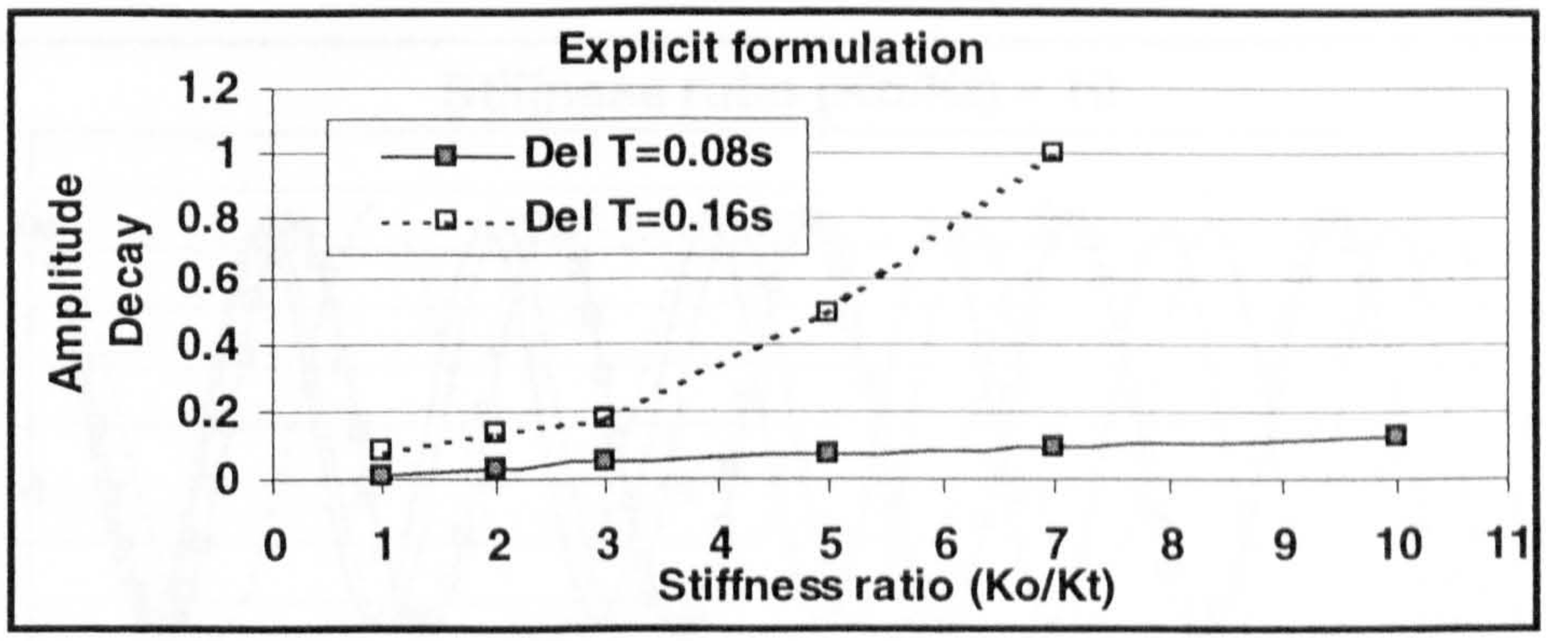


Fig. 4.11 Amplitude decay for explicit method as function of stiffness ratio

4.6.2.2 Newmark Implicit – Integral Form

As shown in the analytical assessment of the scheme, no amplitude error exists when applied to linear systems. It was however expected to display some error in the non-linear systems, where the tangent stiffness diverges significantly from the initial stiffness assumed to remain constant in the implicit part of the expressions. It was therefore surprising to see that no such error existed, even when the initial stiffness was exaggerated by a factor of 10.

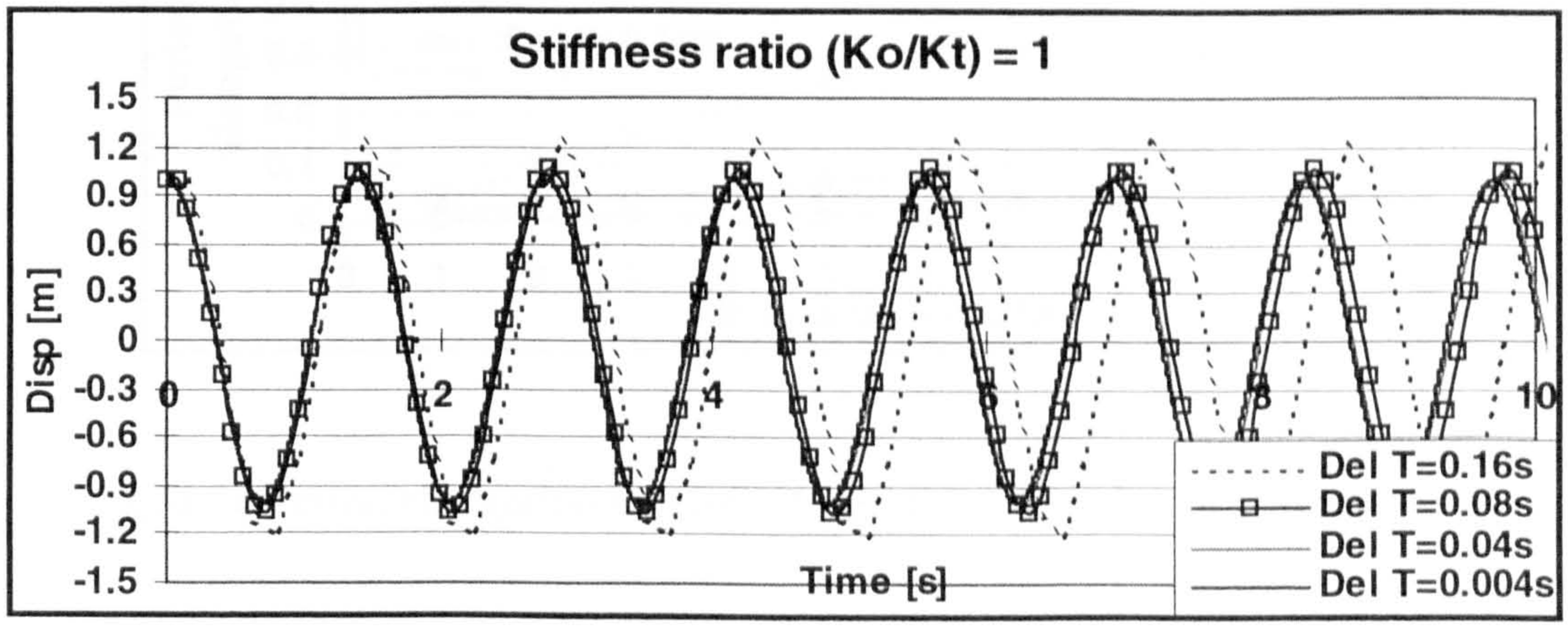


Fig. 4.12 Response obtained using implicit method with k_o/k_t ratio of 1

As seen in figures 4.12 and 4.13, the method remains energy neutral regardless of the time step size and stiffness ratio, although the largest time step size produces a somewhat larger amplitude response. The oscillation period is on the other hand however affected notably.

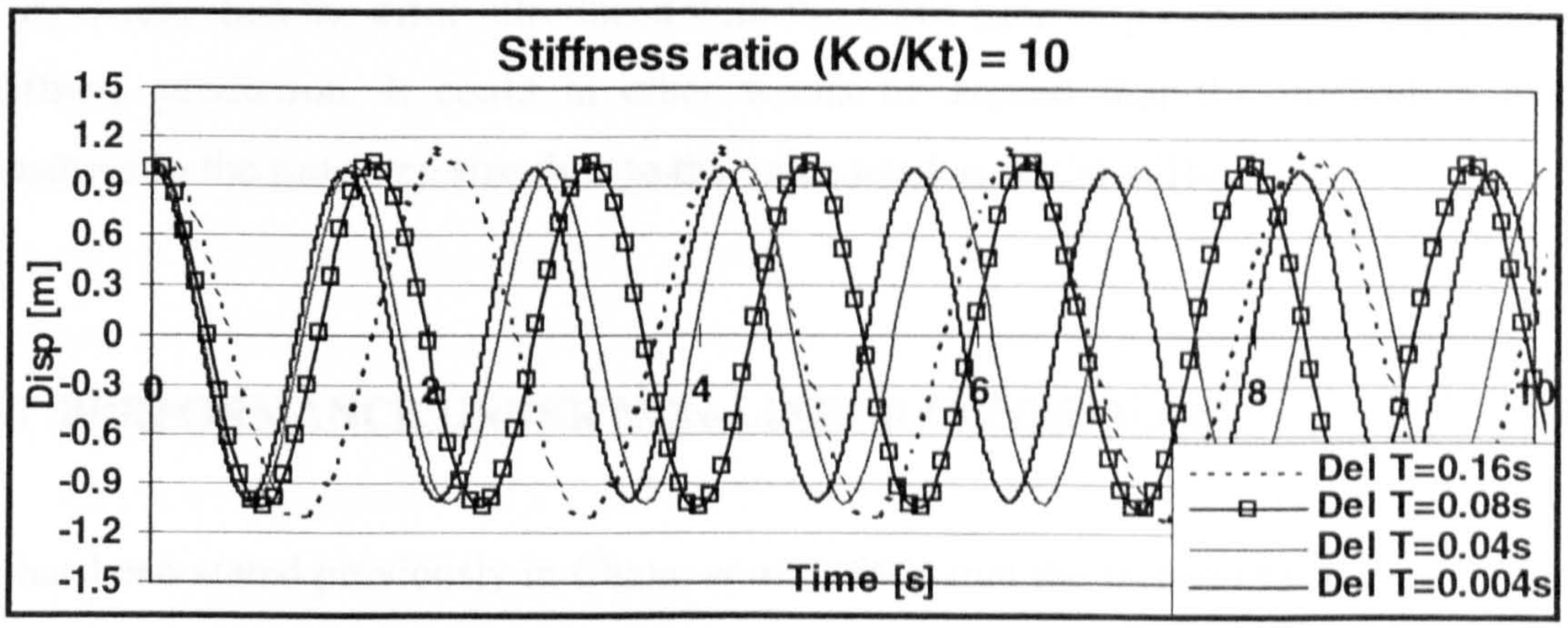


Fig. 4.13 Response obtained using implicit method with k_o/k_t ratio of 10

Simulations were again carried out using initial stiffnesses of 1000, 2000, 3000, 5000, 7000 and 10 000kN/m and time step sizes of 0.004s, 0.04s, 0.08s and 0.16s. The smallest time step size, 0.004s, was assumed to remain exact for all stiffnesses, while the effect on the vibration period for the other three can be seen in figure 4.14.

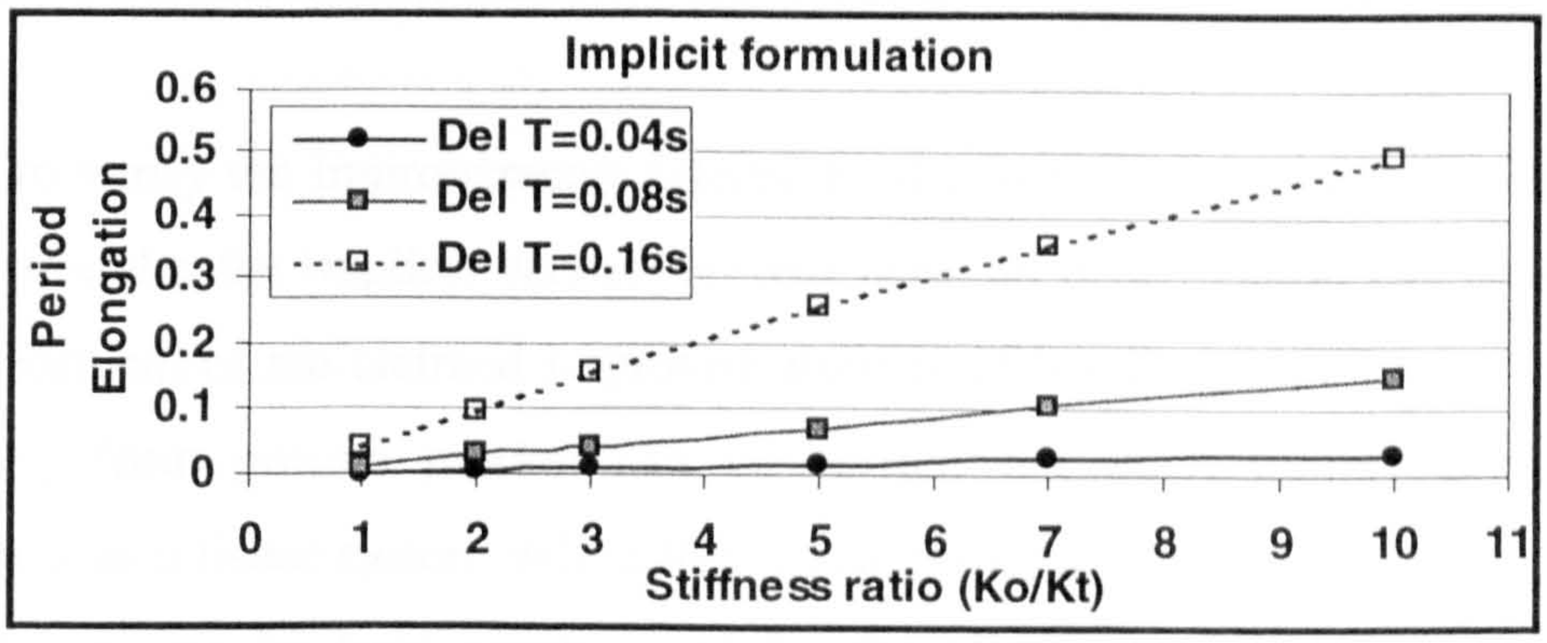


Fig. 4.14 Period elongation in implicit method as function of stiffness ratio

Clearly, for the largest time step size, Δt=0.16s, the period distortion far exceeds acceptable levels for stiffness ratios approaching 10. Even for a moderate stiffness

discrepancy, *Ko/Kt*=2, the period distortion already constitutes 10% elongation. For a somewhat smaller time step size of 0.08s, providing around 17 steps per revolution, the errors are modest. A stiffness ratio of almost 7 is required here to create the same period error. Lastly, with a reasonable time step of 0.04s, providing 35 steps per revolution, the period error remains low (less than 3%) even for stiffness ratios of 10. This is less than the error introduced with the 0.16s time step even when using exact stiffness prediction. It could in other words be argued that the method is more sensitive to the time step size than to the value used as initial stiffness.

4.7 PERFORMANCE UNDER NON-LINEAR CONDITIONS

It has been stated previously in Chang *et al.* (1998) that the integral form method they propose exhibits improved abilities in handling rapidly varying loads and stiffnesses during numerical time integration of dynamic systems. This is conceptually sound, supported by the fact that the methods effectively use a number of sampling points during each time step, which clearly should reduce linearisation errors. Chang's paper does however not convincingly convey that this entails improvements for practical systems on a general level. Although some examples are provided offering evidence of improvements within the two aspects mentioned above, this is not conclusive. This is partly due to incomplete and erroneous documentation of system properties and partly due to the fact that the causes of difference in the response cannot be isolated.

In order to verify the improvements offered by the integral form methods in general, and in particular the implicit version present here, an investigation has been carried out. Assessment of the claimed improved abilities of handling rapidly varying forces originating from ground motion can be carried out fairly simply in numerical simulations as a linear system will suffice in capturing any effects. The time step size and stiffness may be kept constant, whilst the difference between using a single acceleration value for each step, as employed by conventional explicit schemes, or a time integral of the force over the step computed using a number of sampling points, is investigated. This will then clearly yield any direct effects on the response, but not necessarily determine which is more accurate. A form of reference solution may be

obtained using smaller time steps, but this will again affect the response in other ways. Still, this method will provide an indication to the performance in this respect.

Whilst the effect of approximation errors in the truncation of the ground motion accelerogram can be tested on a linear system, linearisation effects in the restoring force term clearly require a non-linear system to be exposed. This makes investigation of such effects through numerical simulations more difficult. As an alternative, pseudodynamic tests may be carried out to capture the linearisation effects, but this introduces a whole range of potential sources of error and makes isolation of specific influences difficult. Repeatability and systematic tracking of specific effects may be problematic as, for example, natural variations within the specimen could influence the response to a similar extent. As the overall results from such a test series may be inconclusive, it would be advantageous to enable also non-linear tests in the numerical simulations. With the restoring force defined as a series of second-order polynomial functions of displacement, a dissipative system displaying some of the typical properties of say a damaged reinforced concrete member can be created. Both single point and integral form multipoint time integration schemes may then be employed to trace any effects using various time step sizes. For all methods, the response should converge to the exact solution with decreasing step sizes.

4.7.1 Ability to capture rapidly varying external forces

The natural phenomenon of earthquakes produces ground motion acceleration with a very wide range of frequencies. While a few display some sort of predominant frequency, others can range from having an effectively single impulse to appear to have an almost random distribution of intensities and frequencies. There is no way of predicting the frequency content accurately. During real earthquakes, the ground acceleration that takes place is recorded, and accelerograms are created. These are simply time histories of the acceleration. The sampling rates of the accelerograms also vary. While the 1940 El Centro quake has been recorded with acceleration values of every 0.02s, 1957 Port Hueneme accelerograms are available with a sampling interval of 0.004s. More recent records have been created with still higher sampling rates.
When testing a structure pseudodynamically, typical time step sizes would lie in the range between 0.01s and 0.1s, of course depending on the fundamental frequency of

the structure concerned and the time stepping scheme employed. In any case, it is clear that sampling intervals in accelerograms are normally significantly smaller than the time step size. Conventional time stepping schemes thus require some form of truncation technique, as only one force value is used per step. This could be done simply by picking up the value corresponding to the time point concerned, or by using some kind of averaging technique. Integral form algorithms on the other hand employ the time integral of the force over the time step, and these will normally utilise the full sampling rate to obtain this, thus taking all the information into account. The difference will of course only be appreciable if there are large variations in the acceleration within a time step. This may well be the case with many accelerograms, but the error involved is of a completely random nature.

To investigate any improvements offered by the integral form method in capturing effects of rapidly varying applied forces, tests were conducted using the N-S 1940 El Centro component. The first 20s of the accelerogram is displayed in figure 4.15, where sampling intervals of 0.02s are employed.

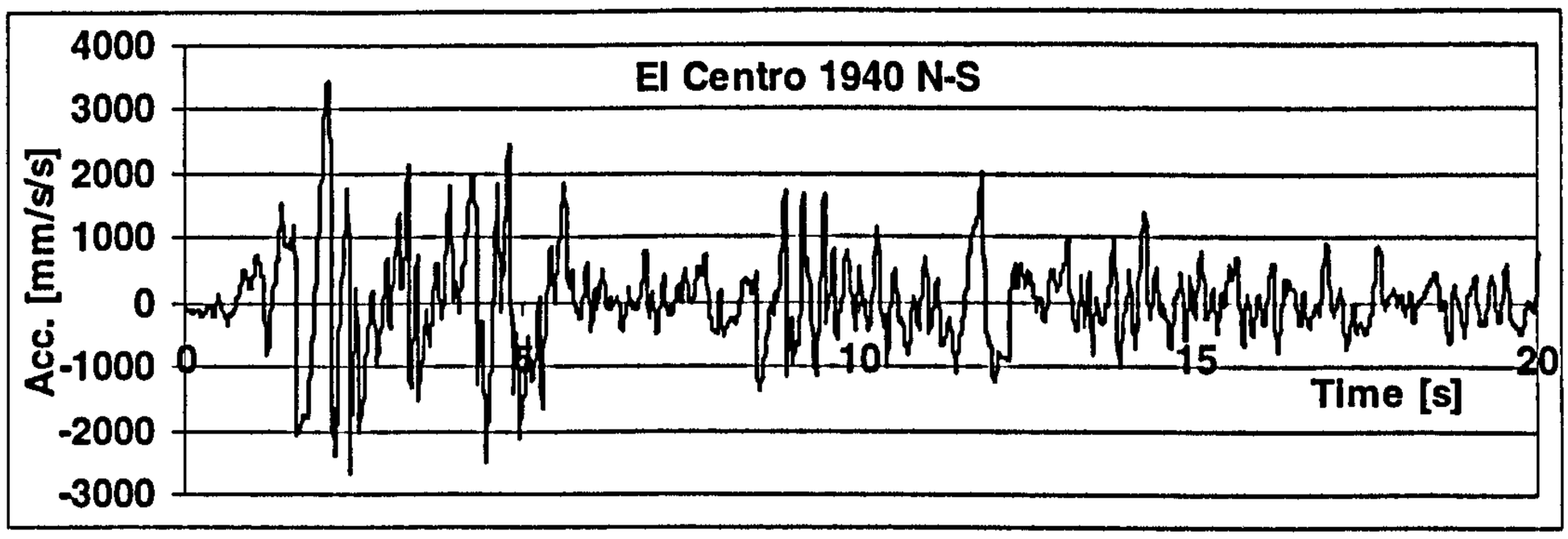


Fig. 4.15 North-South component of the 1940 El Centro earthquake

4.7.1.1 Results

Numerical simulations were carried out using the central difference and the Newmark Implicit – Integral Form methods on dynamic systems using a stiffness of 3000kN/m. Mass was varied between 5400kg and 54000kg to create natural frequencies of 3.75Hz and 1.2Hz. Various levels of viscous damping were also applied, and time step sizes of 0.02s and 0.08s were employed. In all cases, the first 10s of the response

was modelled. Using the 0.02s time step size should enable a form of reference solution, as both schemes would utilise the full sampling rate. This was first attempted using zero viscous damping, but the response obtained using the different schemes deviated far too much, as seen in figure 4.16. Such discrepancies would not be expected in a typical system, where some damping is always present. Prescribing 5% viscous damping should produce a more realistic response, and avoid the somewhat exaggerated response created by the central difference method. Figure 4.17 displays the 5% damped response, where *Del T* indicates the time step size and *C* the percentage of critical damping.

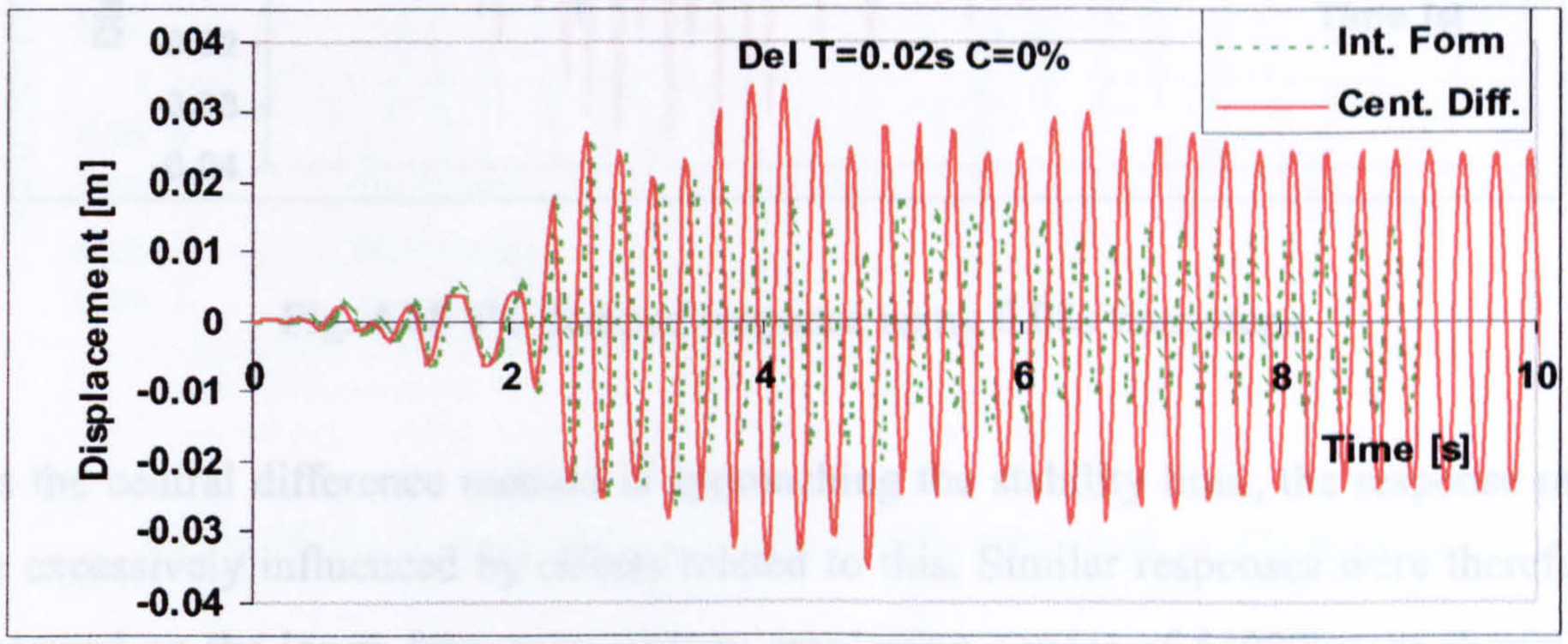


Fig. 4.16 Undamped response using 0.02s time steps

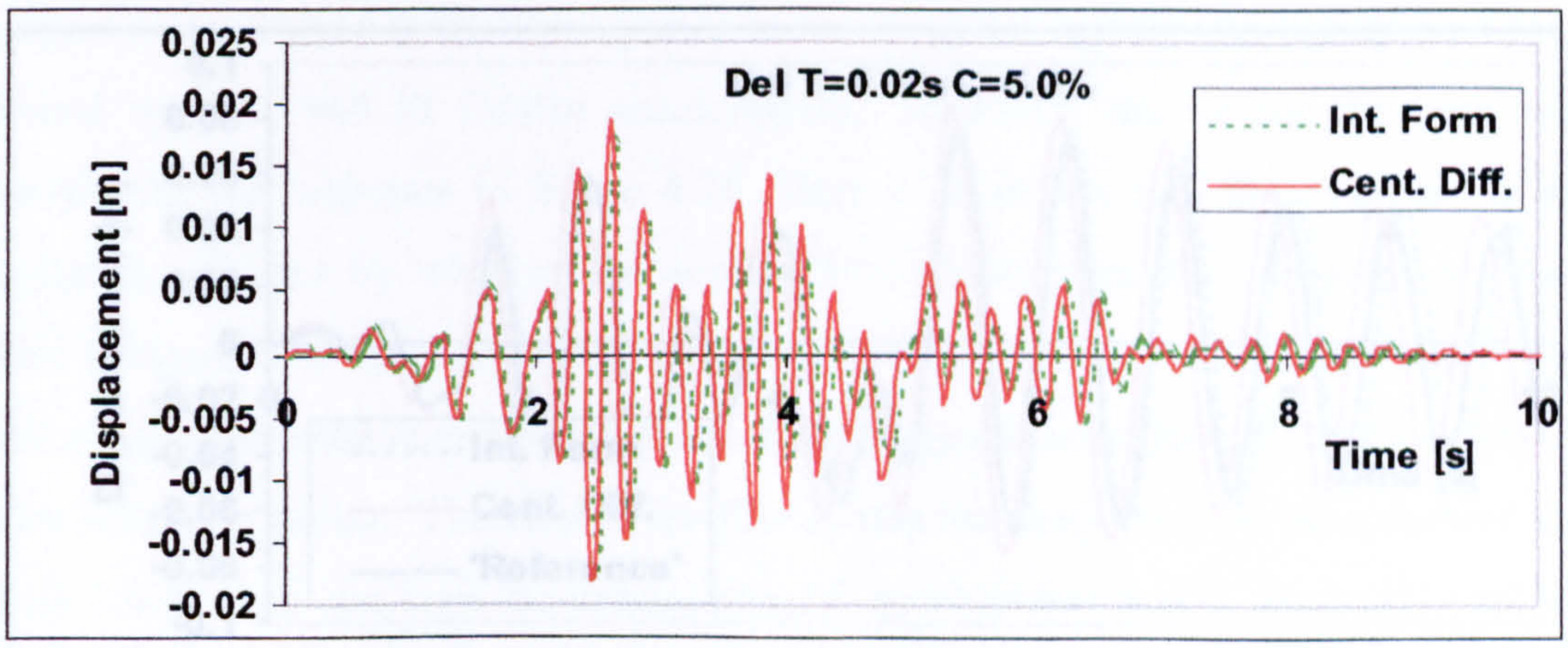


Fig. 4.17 5% damped response using 0.02s time steps

The difference in the response produced by the two algorithms can here be considered small enough to utilise either one of them as a reference solution. When using 0.08s

time steps, the central difference method simply uses every fourth acceleration value, while the integral form method calculates the time-force area using 5 points and Simpson's rule. The response obtained with 5% damping and the larger time step size can be seen in figure 4.18.

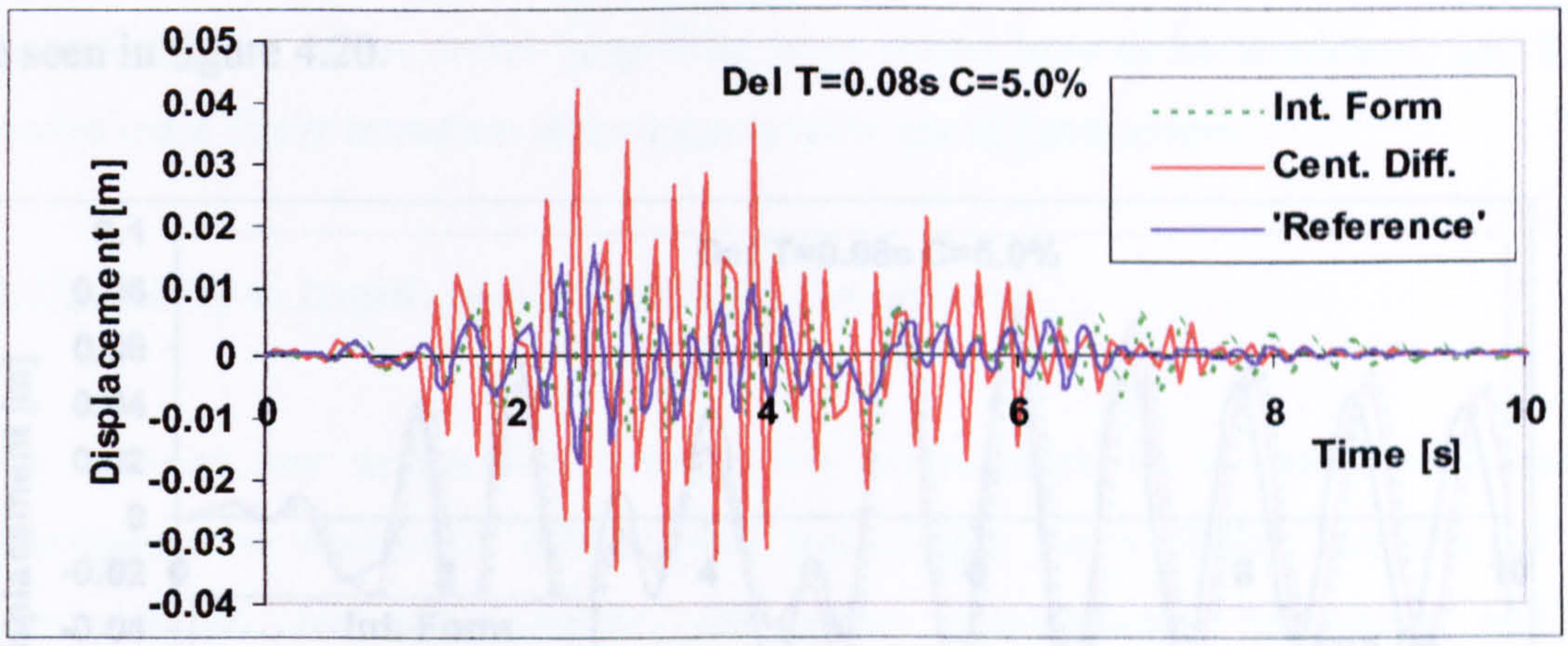


Fig. 4.18 5% damped response using 0.08s time steps

As the central difference method is approaching the stability limit, the response may be excessively influenced by effects related to this. Similar responses were therefore obtained on the lower frequency system; employing a mass of 54000kg, as shown in figure 4.19. The responses obtained with the two methods are now much more similar, but still diverge noticeably.

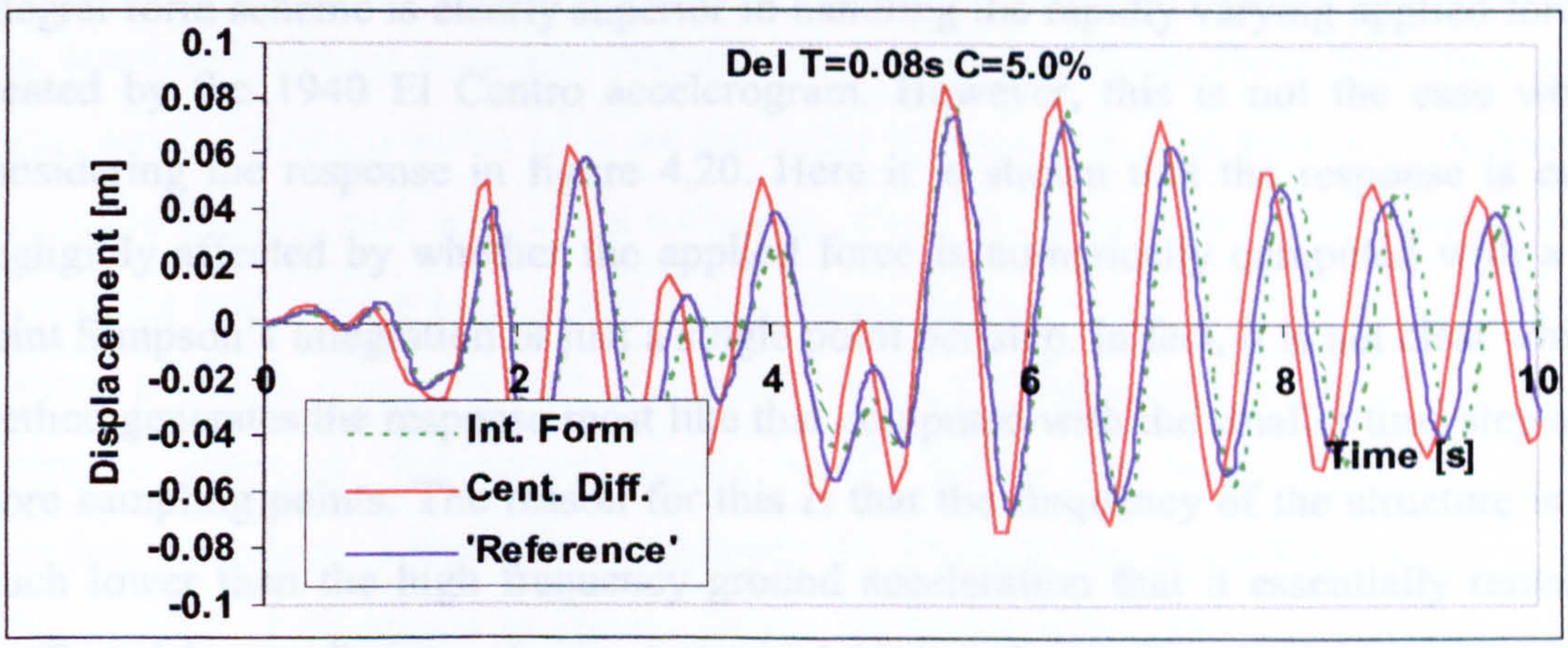


Fig. 4.19 5% damped response using 0.08s time steps and high mass

It is however not clear whether the differences in the response presented in figure 4.19 are due to linearisation effects in the applied force or other algorithmic differences. To

completely isolate the potential effects of linearising the applied force over the time step, the same algorithm has to be used to obtain both responses. In one case, the proper integral of the force is used ("Int. Form" in figure 4.20), while in the other, one assumes the force at the start of the step to act unchanged until the end ("Single Point" in figure 4.20). The response obtained for the same system as that in figure 4.19 can be seen in figure 4.20.

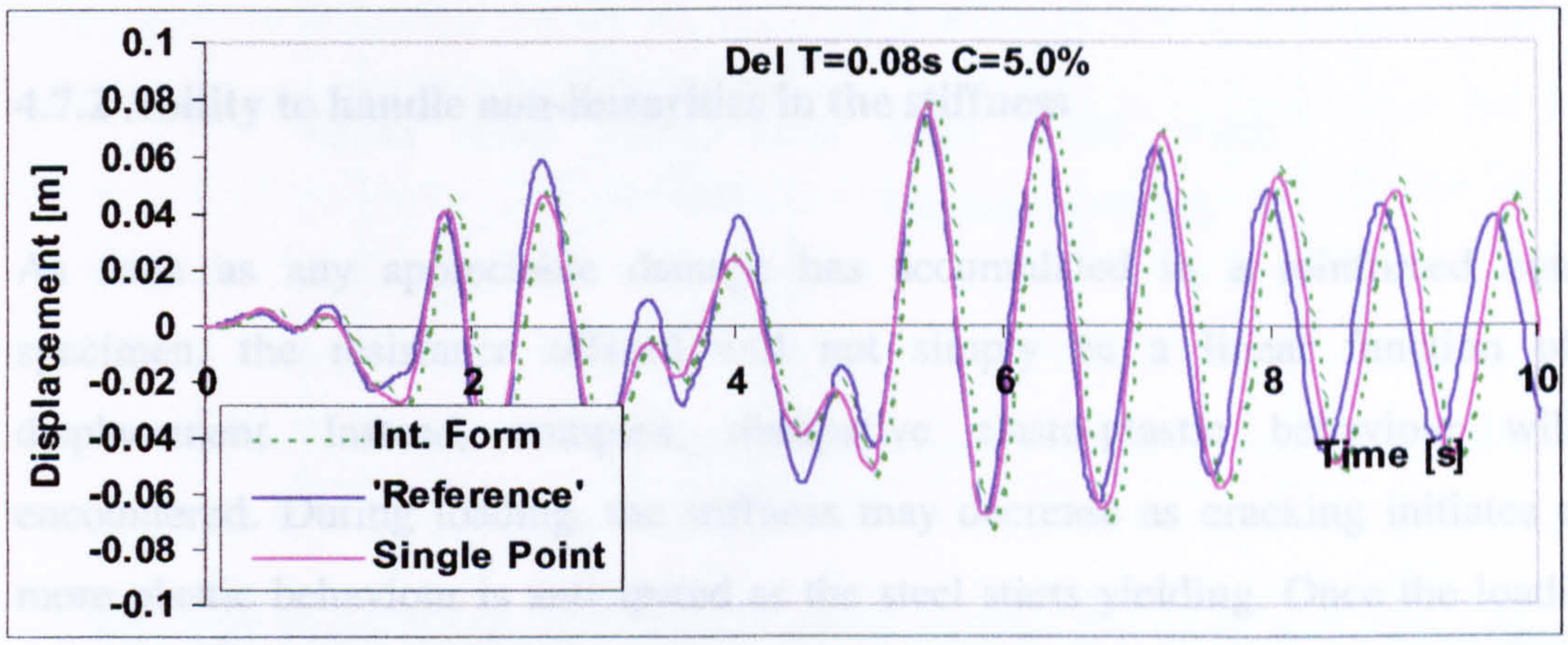


Fig. 4.20 5% damped response using 0.08s time steps and high mass

4.7.1.2 Evaluation

The immediate impression formed when inspecting figures 4.18 and 4.19 is that the integral form scheme is clearly superior in handling the rapidly varying applied forces created by the 1940 El Centro accelerogram. However, this is not the case when considering the response in figure 4.20. Here it is shown that the response is only negligibly affected by whether the applied force is numerically computed with a 5-point Simpson's integration or just a single point per step. In fact, it is not clear which method generates the response most like that computed with the smaller time step and more sampling points. The reason for this is that the frequency of the structure is so much lower than the high frequency ground acceleration that it essentially remains unaffected by small, intra-step variations. Additionally, as the changes during each step are completely random, there is no opportunity for a cumulative effect to develop. So although the response obtained with the integral form method in figures 4.18 and 4.19 is superior to that obtained with the central difference method, this is

due to other algorithmic advantages. It therefore remains apparent that under typical conditions, there is little immediate advantage in being able to capture the high frequency variations in the acceleration. Under particular conditions though, the advantage may be more visible. This could possibly include the situation where a very stiff structure was exposed to an essentially low frequency ground excitation. Still, for there to be any effect, rather large time steps would have to be employed, and these would most likely introduce other unacceptable algorithmic errors.

4.7.2 Ability to handle non-linearities in the stiffness

As soon as any appreciable damage has accumulated in a reinforced concrete specimen, the resistance offered will not simply be a linear function of the displacement. Instead, complex, dissipative elasto-plastic behaviour will be encountered. During loading, the stiffness may decrease as cracking initiates and a more plastic behaviour is anticipated as the steel starts yielding. Once the loading is reversed, a steep reduction in restoring force will take place before further displacements mobilise forces in the opposite direction. If the displacement continues growing, softening and plastic behaviour will again take place before load reversal reduces the restoring force sharply.

If a numerical simulation is used to evaluate the performance of a scheme, it is vital that the fundamentals of the non-linear behaviour are reproduced within the model. In the numerical system, nonlinear stiffness conditions resembling those described above were created using a series of 4 second-order polynomials. The first one represented the initial loading path with strain softening, while the second one corresponded to the unloading path with a high initial drop in force followed by a flattening towards the neutral position. Upon reversal of displacements, the same pattern was created in the opposite direction. Subsequent cycles would follow the same functions, but as amplitudes reduce, a larger jump between the loading and unloading curves occurs. The system of functions is shown below in figure 4.21. The loading function is defined as $R=3.0e6*X-2.0e8*X^2$, and unloading as $R=1.0e6*X+2.0e8*X^2$, while negative displacements are exactly opposite. Although not a particularly good approximation to the hysteresis one would expect from cyclic testing of a reinforced

concrete member, the system is considered well suited to evaluate any improvements offered by the integral form algorithms.

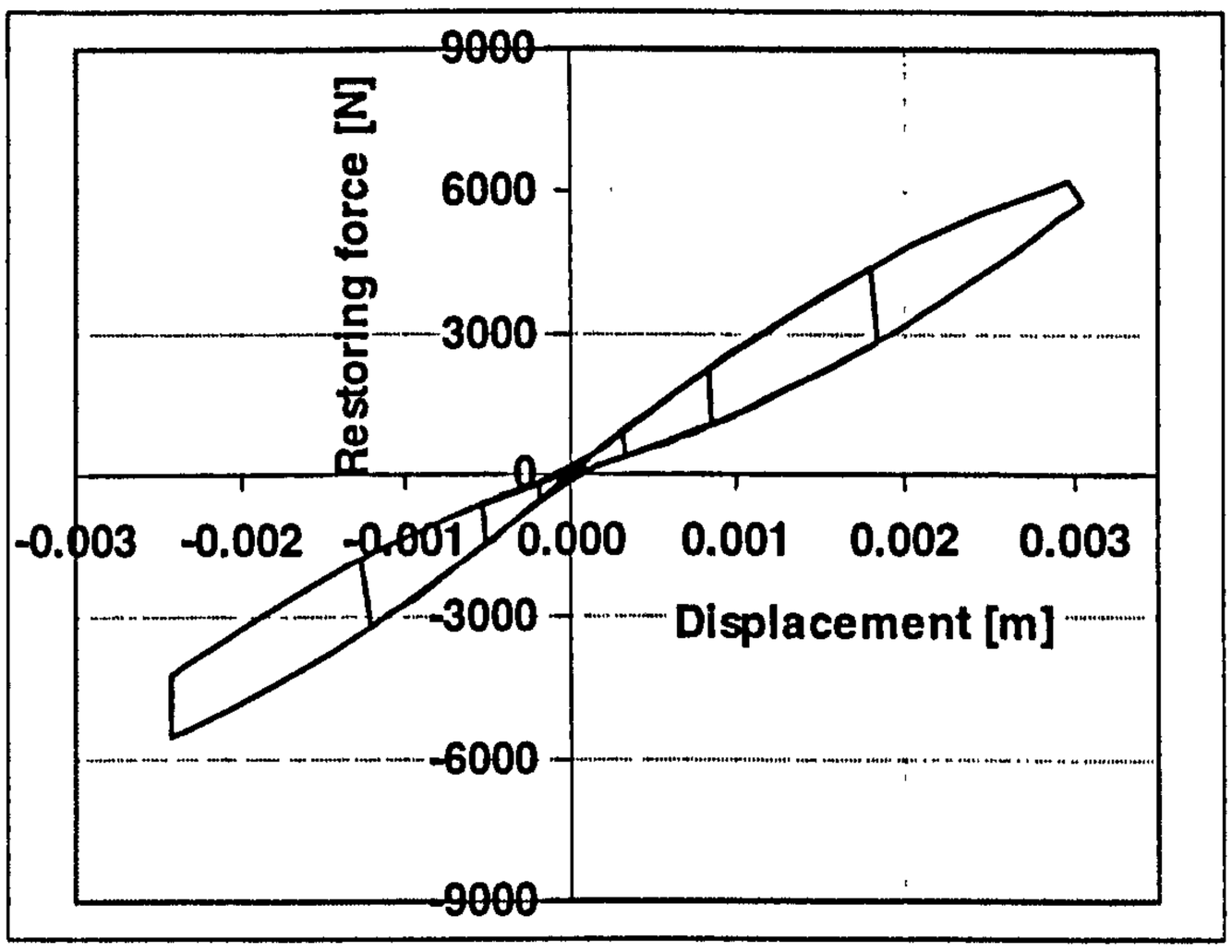


Fig. 4.21 Force-Displacement relationship employed in numerical simulations

The system properties comprised a mass of 5400kg and an initial stiffness of 3000kN/m, giving an undamaged natural frequency of 3.75Hz, which, with the non-linearities, was reduced to around 2.8Hz (of course depending on amplitude). Zero viscous damping applied throughout. The Newmark Implicit – Integral Form and the central difference methods were used to obtain the response with a range of time step sizes. The integral form method employed 5 sampling points per time step, and Simpson's rule (exact for second order polynomials) was used to compute the integral. Only negligible differences were found when employing more sampling points, as the integration scheme is approximate only when displacement steps cross the neutral position. Rather than exposing the structure to an external excitation, the free vibration response following an initial displacement was considered. Using a ground motion accelerogram would introduce further sources of discrepancies between time integration scheme and time step sizes. The schemes were coded in Microsoft Excel 2000. This environment provides a suitable coding language and exhibits the particular advantage that calculations for every time step are displayed simultaneously.

4.7.2.1 Results

The central difference method and the Newmark Implicit – Integral Form methods should, in principle, when employing a small enough time step, produce near identical responses to the initial displacement. This is because the periodicity error present in both cases reduces as smaller time steps are employed and soon becomes negligible for step sizes of a small, but practical size. Furthermore, any linearisation effects in the restoring force will reduce to an insignificant magnitude with equally small steps. In numerical simulations employing time steps of 0.005s, or approximately 70 steps per cycle, this was also found to be the case. The response obtained from this could thus be considered as a reference solution. As any superiority of the integral form can only be expected for sizeable time steps, results using step sizes of 0.01s, 0.05s and 0.075s, providing 33, 6.6 and 4.4 steps per oscillation, are included. The step sizes correspond to $\omega_n \Delta t$ values of 0.19, 0.95 and 1.43, respectively, so the theoretical stability limit for an equivalent linear system of the explicit central difference method is being approached. The response obtained using step sizes of 0.01s, 0.05s and 0.075s can be seen in figures 4.22, 4.23 and 4.24, respectively.

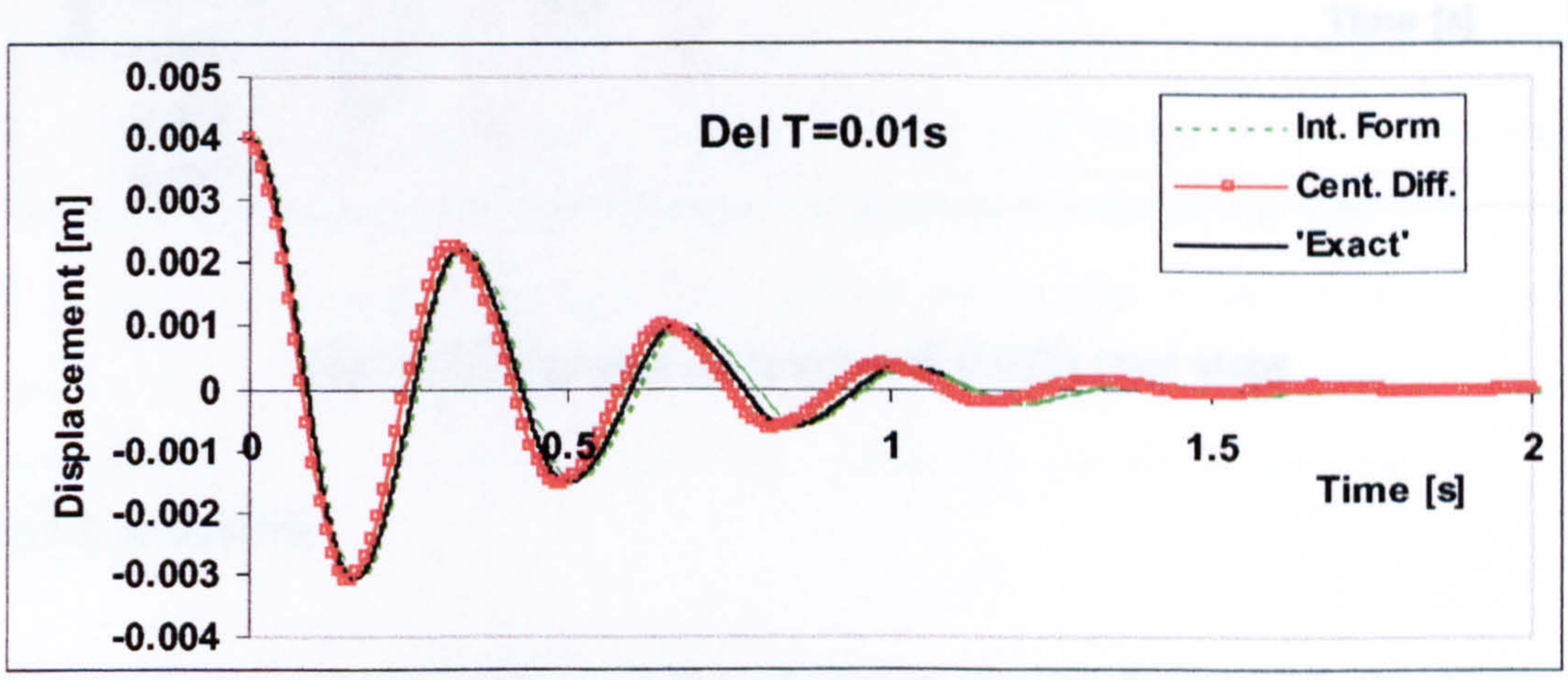


Fig. 4.22 Response obtained with 0.01s time steps

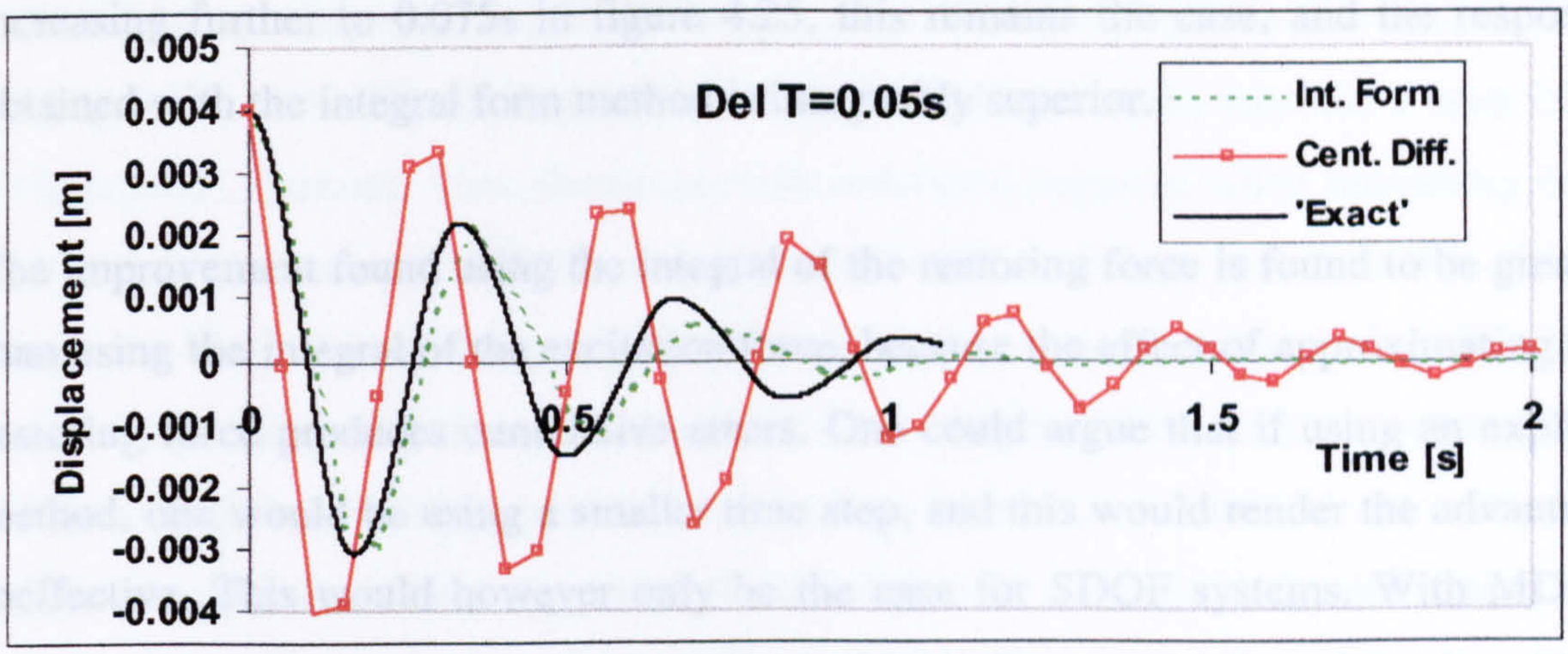


Fig. 4.23 Response obtained with 0.05s time steps

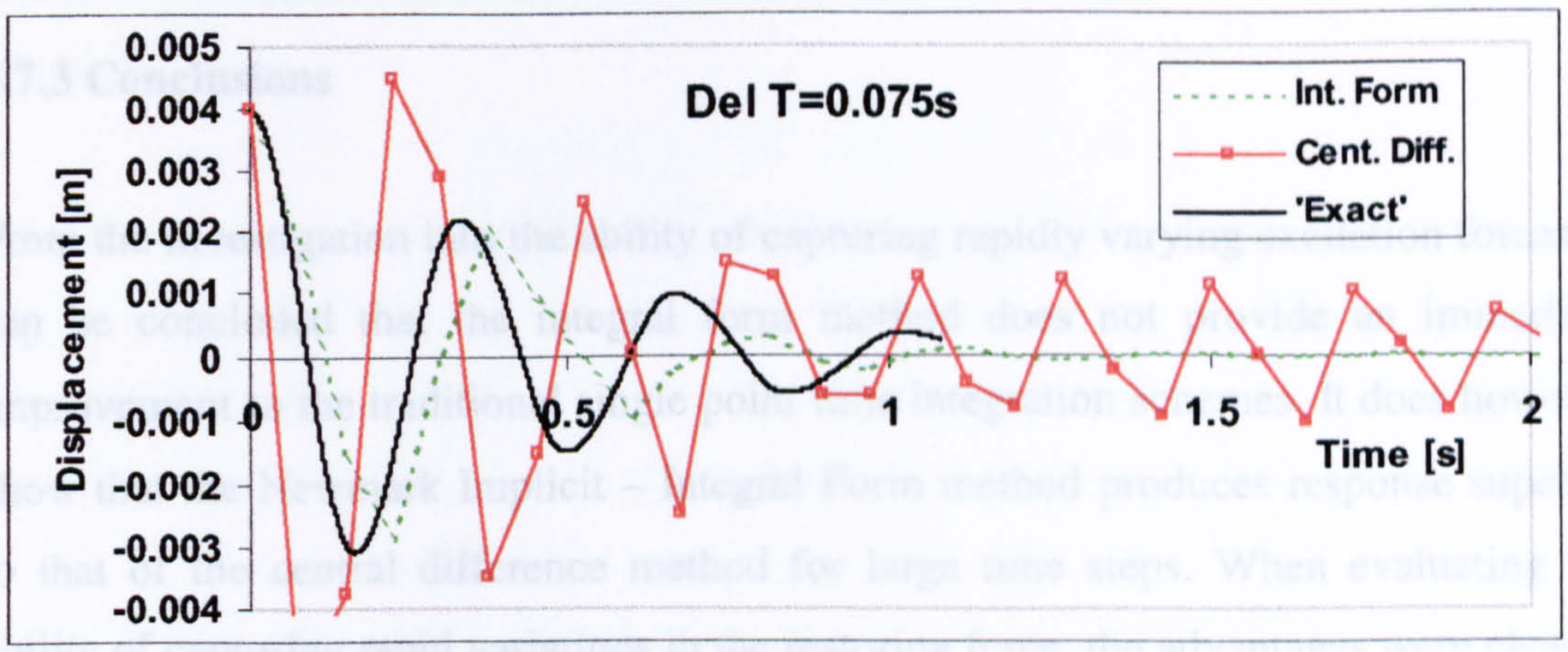


Fig. 4.24 Response obtained with 0.075s time steps

4.7.2.2 Evaluation

When inspecting figure 4.22, a slight advantage can be found with the Newmark Implicit – Integral Form method, but both this and the central difference methods produce satisfactory results. This is because the steps are still reasonably small, and the linearisation effects remain minor. When the step size is increased to 0.05s however, the differences become more apparent. In figure 4.23, it can be seen how the central difference method creates a highly exaggerated response, while the integral form method still remains close to the near exact solution obtained with the small time step. The integral form method displays less period and amplitude error, and clearly represents an improvement to the central difference method. With time step sizes

increasing further to 0.075s in figure 4.25, this remains the case, and the response obtained with the integral form method is inarguably superior.

The improvement found using the integral of the restoring force is found to be greater than using the integral of the excitation force, because the effect of approximating the restoring force produces cumulative errors. One could argue that if using an explicit method, one would be using a smaller time step, and this would render the advantage ineffective. This would however only be the case for SDOF systems. With MDOF systems, keeping a low $\omega_n \Delta t$ number with the higher frequencies would require a time step size too small for practical purposes. In this case, the integral form method remains superior in dealing with intra-step variations in stiffness.

4.7.3 Conclusions

From the investigation into the ability of capturing rapidly varying excitation forces, it can be concluded that the integral form method does not provide an immediate improvement to the traditional single point time integration schemes. It does however show that the Newmark Implicit – Integral Form method produces response superior to that of the central difference method for large time steps. When evaluating the ability of capturing rapid variations in the restoring force, the advantages were clearer. With large time steps in the non-linear system, the integral form method performed consistently better than the central difference method. This improvement is substantiated through the avoidance of cumulative errors in the restoring force component.

4.8 CONCLUSIONS

By including an additional term in the estimate for the restoring force in the Newmark Explicit – Integral Form time-integration algorithm, a new displacement predictor has been proposed, which is consistent with the constant average velocity the algorithm assumes. This addition eliminates numerical damping which is otherwise present. Furthermore, the modification renders the algorithm unconditionally stable. It now

successfully combines potential advantages of the integral form in handling rapidly varying loads and stiffness degradation with the unlimited time step sizes associated with implicit methods. This should provide enhanced capacity when modelling both real and sub-structured MDOF structures as well as being excellent for SDOF structures. Both the advantages have been displayed through numerical simulations and proven analytically through assessment of the amplification matrices representing the recurrent relationships between the state vectors. A method for effective implementation has been suggested in the form of a semi-continuous combined control-time integration algorithm. Following an analysis into the sensitivity of the magnitude adopted for the initial stiffness term, it has been found that the implicit method exhibits no amplitude error and less periodicity error than the explicit method, which also produces significant dissipation at large time steps.

Chapter V:

DEVELOPMENT OF THE CONTROLLER

This chapter deals with the theoretical and practical development of the controller for the pseudodynamic testing system described in this thesis. It provides a short discussion on the objectives of the controller and an introduction to some of the main theory behind the control in general. It then describes in more detail how this particular control system has been built up, elaborating furthermore on the signal generation and implementation of the control modes. Finally, a short, basic evaluation of the controller is included.

5.1 OBJECTIVE

The primary objective of the controller is to generate signals that manipulate the controllable variable, in this case the signal to the servo valves, such that the controlled variable, in this case displacement, achieves its target. This is done by continuously reading the controlled variable and adjusting the signal accordingly. Essentially, the controller, in pseudodynamic testing, is the device that ensures that the displacement target provided by the time stepping algorithm is imposed on the experimental structure by the hydraulic actuator.

The target should be reached directly, and preferably in a swift and smooth manner with minimal overshoot. In the context of pseudodynamics, this ensures that the representation of the elastic restoring force is created as accurately as possible. Any overshoot could cause erroneous plastic deformation of the specimen resulting in loss of strength and/or inaccurate restoring force measurements. It is also generally desirable to keep loading rates smooth and as high as possible to reduce the deviation from the rate of loading existing during, for example, strong ground motion.

A secondary objective of the controller is, in the case of the implementation with integral form algorithms (recall Chapter IV, section 4.5), to numerically obtain a measure of the time integral of the restoring force over each time step.

5.2 THEORETICAL CONTROL MODES

In control theory, three distinct modes of control exist, the *proportional*, *integral* and *derivative* modes. They all display individual properties and can be combined to form an optimised control system. Prior to elaborating on the specific control system developed for the pseudodynamic tests and the hydraulic system described in this thesis, a brief discussion on the control theory has been included. This will clarify some of the main theoretical aspects of control engineering, which will be relevant for the later discussion about the development of the pseudodynamic test set-up. The introduction to the basic control modes is based on Bateson (1996).

5.2.1 The proportional control mode

One of the elementary modes of control is the *proportional* mode. In this mode, the signal generated is proportional to the error. In the context of pseudodynamics, this implies that the signal sent to the servo valve is proportional to the difference between the measured and the targeted displacements. This requires that a reference signal, a zero signal, exists, onto which the proportional signal can be added.

The mode should ensure a velocity profile that starts off high while the displacement discrepancy is high and then slows down as the remaining discrepancy reduces. Furthermore, it inherently takes sign reversals into account so that overshoot should be stopped immediately. The mode does however display the weakness that the target positions are not necessarily always met. If an error remains and the signal generated based on this is not sufficient to initialise further movement, the system simply gets "stuck" and cannot proceed.

The proportional control mode can be employed independently, and many control systems employ only this mode. During the initial stages of the development of the controller described in this thesis, it was believed that the proportional control mode would function satisfactorily on its own for the pseudodynamic system concerned.

5.2.2 The integral control mode

The *integral* control mode generates signals proportional to the integral of the error. In the pseudodynamic setting, this means that the integral control signal for the servo valve will be generated proportionally to the integral of the displacement error with respect to time. In practise, this means that if, for any reason, an error remains in the displacement, the valve control signal will continue growing until further motion is initiated.

The integral control mode ensures that the target displacement is always finally met regardless of the resistance in the system, but does on the other hand not provide the ability of controlling well the rate at which the target is approached. The mode does take error reversals into account, but does not provide an immediate, strong correction to the signal. Although the mode can be employed independently, it is more often combined with the proportional mode. This should theoretically enable a fast, responsive controller which always ensures the target position is met.

5.2.3 The derivative control mode

The last, and possibly the least important of the three control modes is the *derivative* mode. As the name suggests, this mode generates signals proportional to the derivative of the error, or in other words the rate of change of the error. In the pseudodynamic framework, this will mean that the valve signal is generated proportionally to the time derivative of the displacement error, or more specifically, the velocity of the actuator.

The mode is effectively able to control the speed of the actuator, but has no way of ensuring that the target is met. It does take sign reversals into account, but can only reverse the signal based on this; the rate remains the same. The mode is therefore not employed independently, but can successfully be combined with the proportional or proportional-integral modes. This may improve the control of the rate of change of displacement, or in pseudodynamics, the velocity profile.

5.3 THE CONTROL LOOP

Regardless of whether the controller is created as a piece of software or hardware, it is clear that it functions as a loop structure. For every cycle of the loop, the input signals are read and a series of calculations are carried out before the output is computed and finally written as a signal. This process runs continuously both when moving the actuator and when keeping it in position. The frequency of controllers varies considerably, and can be as high as in the mega-Hz rage. For pseudodynamic tests, it is typically required to have a rate of at least 100Hz, or 100 iterations per second, although substantially higher rates are beneficial.

In order to implement the control modes discussed in the previous section, the signal contribution from each mode is computed in each iteration based on the positional error, the integral of the positional error or the derivative of the positional error. The error calculations are based on the input read in the same cycle, while the adjusted output signal is compiled from the contributory modes.

The following sections contain a discussion on the processes that take place within the control. loop. It elaborates on the input and output, ramp generation, dead-zone compensation and signal generation as well as explaining how the processes are tied together within the pseudodynamic set-up.

5.3.1 Input and output

Implementation of a SDOF pseudodynamic test requires a minimum of three communication channels: displacement and force measurements and the servo valve signal. For control purposes on the other hand, the force signal is not required, leaving the displacement and the valve signal as the input and output channels, respectively.

Reading the input is one of the first tasks to be carried out by the controller. This enables the "error" to be calculated, which forms the basis of most of the other processes within the controller. The error in this set-up and in general in a pseudodynamic framework, is the discrepancy between the target displacement

computed by the time integration algorithm and the current displacement as measured by the displacement transducer and as read at the start of the control loop.

The valve signal is generated through a series of processes within the controller, which will be discussed in the next sections. Generally speaking, this signal will consist of a smoothly changing voltage, except when keeping the actuator stationary, in which case it will be rapidly oscillating over the *dead-zone* as discussed in section 5.3.3.

Both the input and output signals are analogue and thus simply consist of voltage levels within a certain range. The controller must relate this level to a displacement value in millimetres. A discussion on how this is carried out is included in section 5.3.6. The output signal on the other hand has got no other meaning than that interpreted by the servo valve and cannot be related to any physical quantity. The voltage range applicable to the valve is very small, as it in reality operates on current rather than voltage. Care must therefore be taken to ensure that excessive signals, which could damage the valve, are not created. This has been done by including a series of filters in the controller that limit the maximum signal sent to the valve.

5.3.2 Ramp generation

As the pseudodynamic test method progresses in a step-by-step method, the time stepping algorithm supplies the controller with discrete displacement positions but says nothing about the displacement history in between. It is therefore the task of the controller to create a displacement-time curve for each displacement step.

Some pseudodynamic implementation systems employ a linear displacement profile, e.g. Combescure & Pegon (1998). This can be achieved with a high specification controller employing a ramp generator, and ensures that the velocity profile corresponds to that assumed in time stepping algorithms (e.g. constant). The ramp generator computes the desired displacement path based on the starting position, the target position and desired implementation time, as indicated in figure 5.1, where d_n and d_{n+1} are the target displacements at the start and end of the step, respectively.

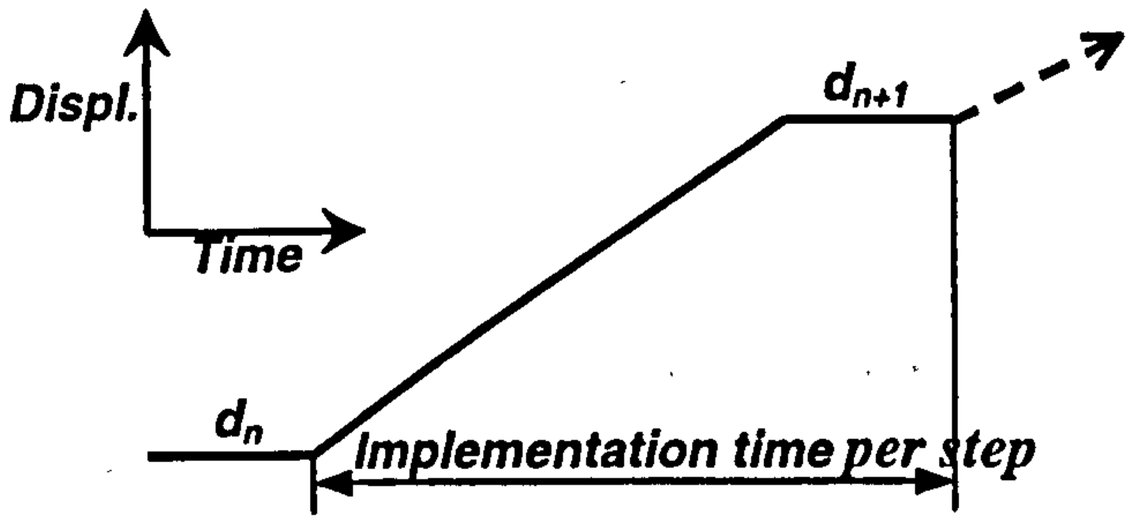


Fig. 5.1 Displacement ramp

The control system developed by the author does not employ a ramp generator as such. This is mainly because such an implementation system is fairly resource consuming and requires the controller to run at a high frequency. In order to ensure that not only the final target position is reached accurately, but also that the whole displacement path closely follows the predefined one, more intermediate signal adjustments are required than when simply attempting to reach the target.

In pseudodynamic tests, it is not imperative that the velocity during the displacement step remains constant, as long as it is ensured that the final displacement target is reached directly, without any overshoot. In fact, it may be beneficial to employ a highly non-linear displacement profile if high-speed implementation is desired. This can enable a high initial velocity to attempt to impose the increment as fast as possible, and then slowing down the actuator when approaching the target to ensure high accuracy implementation. Here, the developed controller tends to generate such a velocity profile, but this depends on the relative influence of the proportional and integral control modes as discussed in section 5.3.4.

5.3.3 Dead-zone compensation

When controlling any form of system, it is often the case that a so-called *dead-zone* is present. A dead-zone in control engineering is defined as the situation when a change in the input does not result in a change in the output (Bateson 1996). During the initial stages of the development of the controller, the existence of a significant dead-zone compromised the control considerably, and required remedial action. The dead-zone

in the system concerned was probably a combined consequence of the servo valve itself and the frictional characteristics of the hydraulic actuator.

The effects of an uncompensated dead-zone were such that close control was impossible even for an unloaded system. This was essentially due to the fact that the signals required to initiate and terminate the actuator motion were not the same. As an example, consider the situation where the actuator is moving towards a target position under proportional and/or integral control. Once the actuator reaches its target, the algorithm will create a negative signal change, but this does not suffice to arrest the actuator. This is due to the existence of a dead-zone, which requires that a much stronger signal has to be reached before the actuator will finally stop or reverse. By this time, of course, the actuator will have overshot or even reached the end of its stroke. This situation is clarified in figure 5.2 below. In the figure, the displacement measurements and the signal levels are superimposed. The blue line displays the displacement and corresponds to the vertical axis on the left hand side while the violet line represents the valve signal corresponding to the right hand vertical axis. The horizontal, black lines indicate the limits of the dead-zone in terms of the signal, and take on values of just over +0.1V and almost –0.9V.

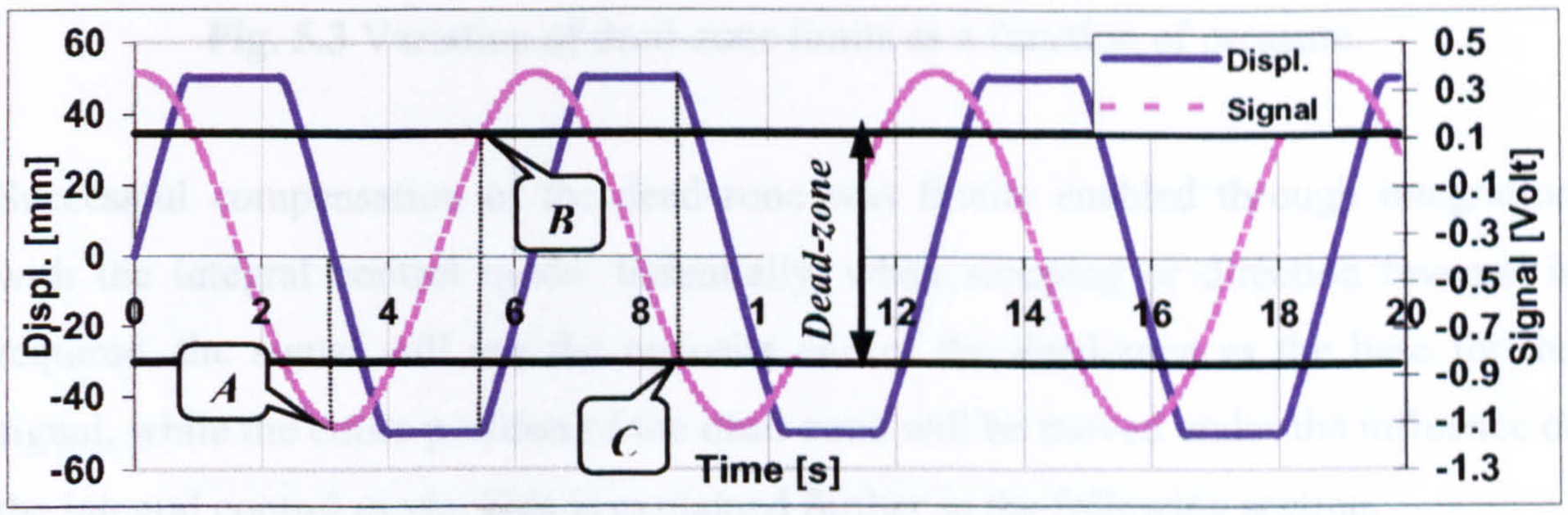


Fig. 5.2 Effect of an uncompensated dead-zone.

When the displacement reaches the commanded position, in this case 0mm, the signal turns (Point A) but the actuator continues moving past the target until saturation. It is not until the signal reaches Point B the velocity is finally reversed. Following another interception with the target position, it again continues to move past it, until the signal reaches point C. The dead zone can thus be defined as the range existing between the two horizontal lines passing through points B and C.

The dead-zone creates further complications, as its position is not constant. The voltage range shown in figure 5.2 above only applies to unloaded conditions. If high pressures are developed within the system, as will of course be the case during real tests, the position of the zone will shift. A trace of the effective dead-zone as a function of the pressure was obtained experimentally, and can be seen in figure 5.3. It should be noted that the dead-zone is not a simple function of the pressure and that for example zero pressure does not correspond to the centre of the dead-zone being at zero volts. Any compensatory system clearly needs to also take the position of the dead-zone into account.

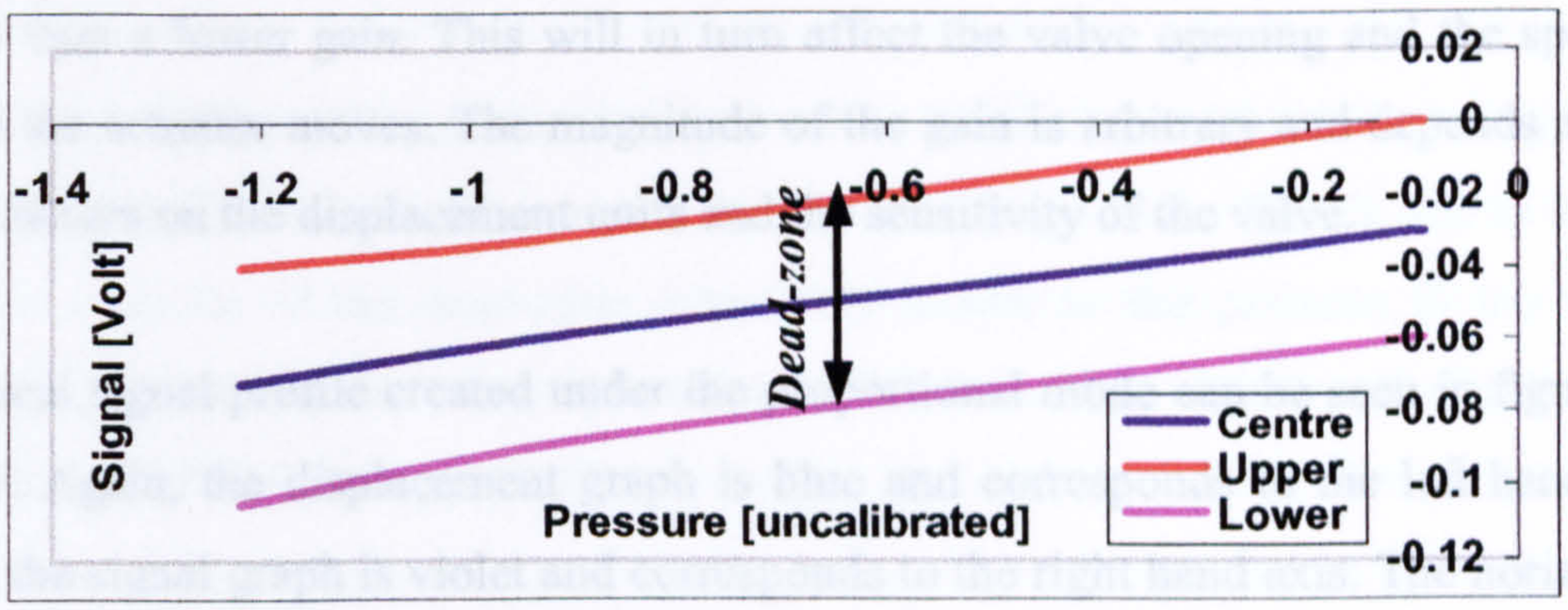


Fig. 5.3 Variation of dead-zone limits as a function of pressure.

Successful compensation of the dead-zone was finally enabled through integration with the integral control mode. Essentially, when stopping or direction reversal is required, the signal will use the opposite end of the dead-zone as the base for the signal, while the entire position of the dead-zone will be moved under the influence of the integral control mode. This is explained further in the following section.

5.3.4 Implementation of the control modes

Two of the theoretical control modes described in section 5.2 were incorporated into the controller to provide a fast, responsive and rigorous control system. The proportional mode is possibly the most direct and easily understood mode, and also provides a favourable velocity profile. However, to ensure that all displacement

targets are actually met, a form of integral control was also included. The derivative mode has not been used, partly because it was not needed and partly because it would be difficult to apply with a low frequency controller.

The proportional control mode uses the limits of the dead-zone as a reference signal. Depending on the direction of the displacement step, one of the limits of the zone will be chosen to represent the zero signal. The proportional component will be added on top of this, and is based on the displacement error currently existing in the cycle concerned. The error is then multiplied by the proportional *gain* to form the actual signal component. This gain thus forms the relationship between the error and the proportional signal component. A high gain will, for a certain error, create a stronger signal than a lower gain. This will in turn affect the valve opening and the speed at which the actuator moves. The magnitude of the gain is arbitrary and depends among other factors on the displacement units and the sensitivity of the valve.

A typical signal profile created under the proportional mode can be seen in figure 5.4 below. Again, the displacement graph is blue and corresponds to the left hand axis while the signal graph is violet and corresponds to the right hand axis. The horizontal, black lines are the dead-zone limits, which correspond to the right hand axis.

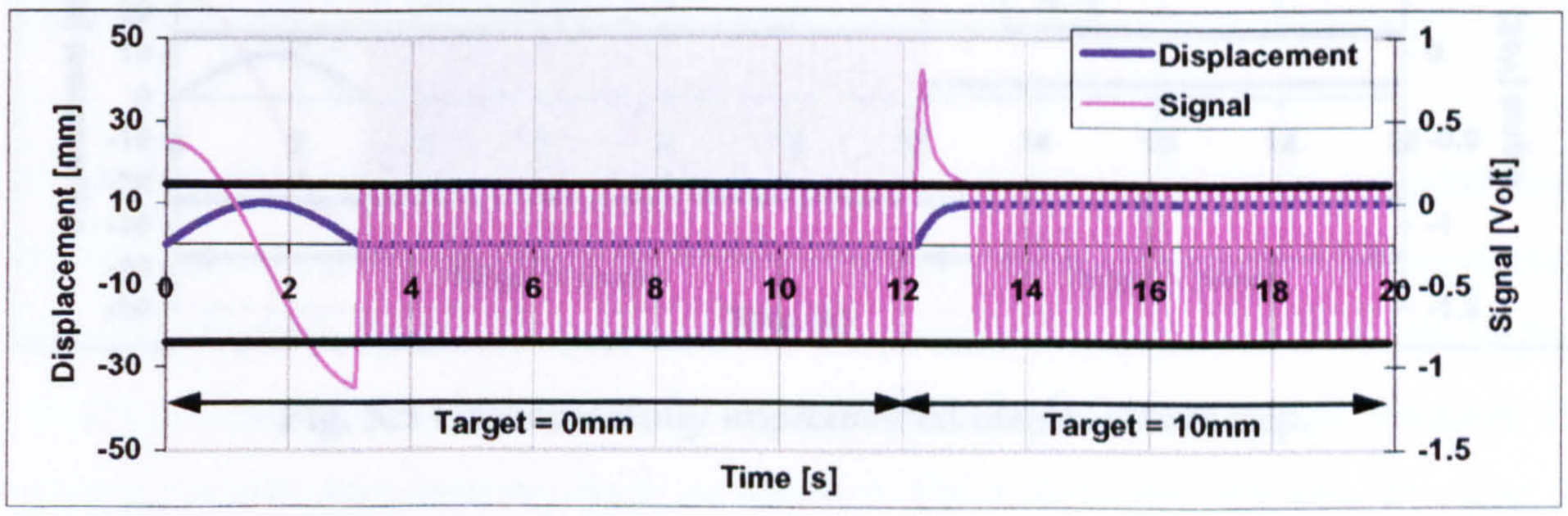


Fig. 5.4 Signal characteristics with the proportional mode.

In this example, the target position is 0mm until 12s into the test when the new target of 10mm is set. While keeping the actuator stationary, the signal constantly jumps between the two dead-zone limits to adjust the exact position. When the new target is set, a large error is effectively created. This results in a large signal being generated by the proportional mode, placed on the upper limit of the dead-zone as that is the

direction of the displacement step. The signal will immediately initiate movement, and as the displacement increases and the error reduces, the signal drops off slowing down the actuator. When the target is reached, the signal contribution from the proportional mode reduces to zero and the signal returns to oscillating between the limits, keeping the actuator in position.

The proportional control mode described above ensures an initially high signal, giving rise to high actuator speeds. This enables fast implementation of pseudodynamic tests, which can reduce potential rate effects. The velocity is on the other hand minimal as the target is reached, reducing the potential for overshoot and providing high accuracy in final force readings.

While the signal profile generated above would be ideal for unloaded or very lightly loaded systems, it would prove unable to deal with large loads. This is due to the fact that the position of the dead-zone effectively moves as the pressure in the system changes. Unless this is taken into account, the resulting signal and displacement profiles would be as in figure 5.5.

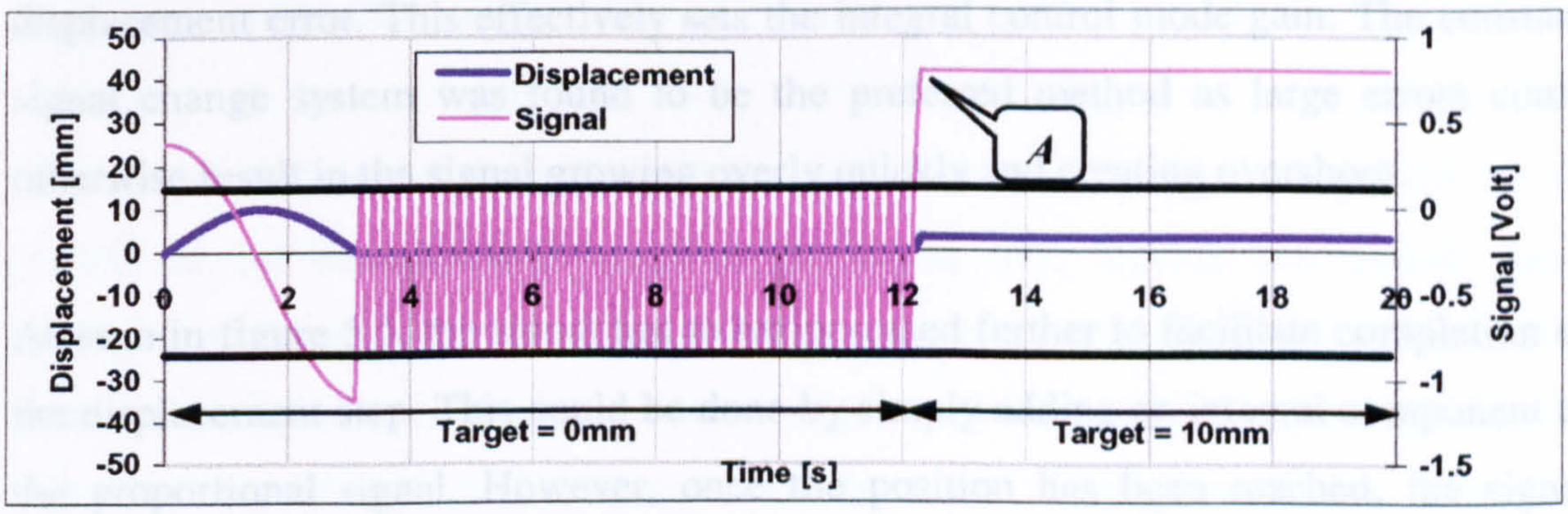


Fig. 5.5 Unsuccessfully implemented displacement step.

As in figure 5.4, a new displacement target of 10mm is set at 12s. The signal immediately grows as a result of the large error, and displacement is initiated. However, after only part completion of the step, the actuator stops. The structure offers so much resistance that the signal generated by the proportional component does not suffice in supplying enough pressure to the actuator to enable further displacements. The load is too high.

The integral control mode offers ideal features to deal with the problem above. This is because, in principle, it generates a signal component according to the integral of the error, and the remaining part of the displacement step in the example above will produce such an error when integrated with respect to time. As long as an error exists, the time integral of it will continue changing, altering the signal such that the error is eventually eliminated. In the example above, the integral mode would continue increasing the signal beyond point A indicated in figure 5.5.

The precise method of implementing the integral mode can vary. The signal change may be applied according to the exact integral based on the real-time or on a more approximate integral based on the iteration rate of the controller. Furthermore, the actual displacement error, or simply the direction of this, may be used when computing the integral and generating the signal. In the controller described in this chapter, a system of applying a certain voltage change for each control iteration has been adopted. In other words, this does not take the duration of the control iteration into account. As this duration remains fairly constant, no advantage of using the more complex real-time system could be envisaged. The voltage change for each iteration is set manually by the operator and is not a function of the magnitude of the displacement error. This effectively sets the integral control mode gain. The constant signal change system was found to be the preferred method as large errors could otherwise result in the signal growing overly quickly and creating overshoot.

As seen in figure 5.5, the signal has to be increased further to facilitate completion of the displacement step. This could be done by simply adding an integral component to the proportional signal. However, once the position has been reached, the signal should not drop back to the other end of the dead-zone. If this were to be the case, the actuator would start moving back as soon as the target was reached. In order to maintain stable and accurate control, it is not sufficient simply to add an integral signal component. Instead, a highly effective system was created by shifting the base signal under the integral mode. This allowed a gradual increase in the overall signal as errors remained and formed a new base for further steps. To avoid displacements dropping off after reaching the target, also the other end of the dead zone was shifted, effectively following the curves in figure 5.3. The displacement and signal profiles for

the example in figure 5.5 would with this implementation of the integral mode look like the one in figure 5.6 below.

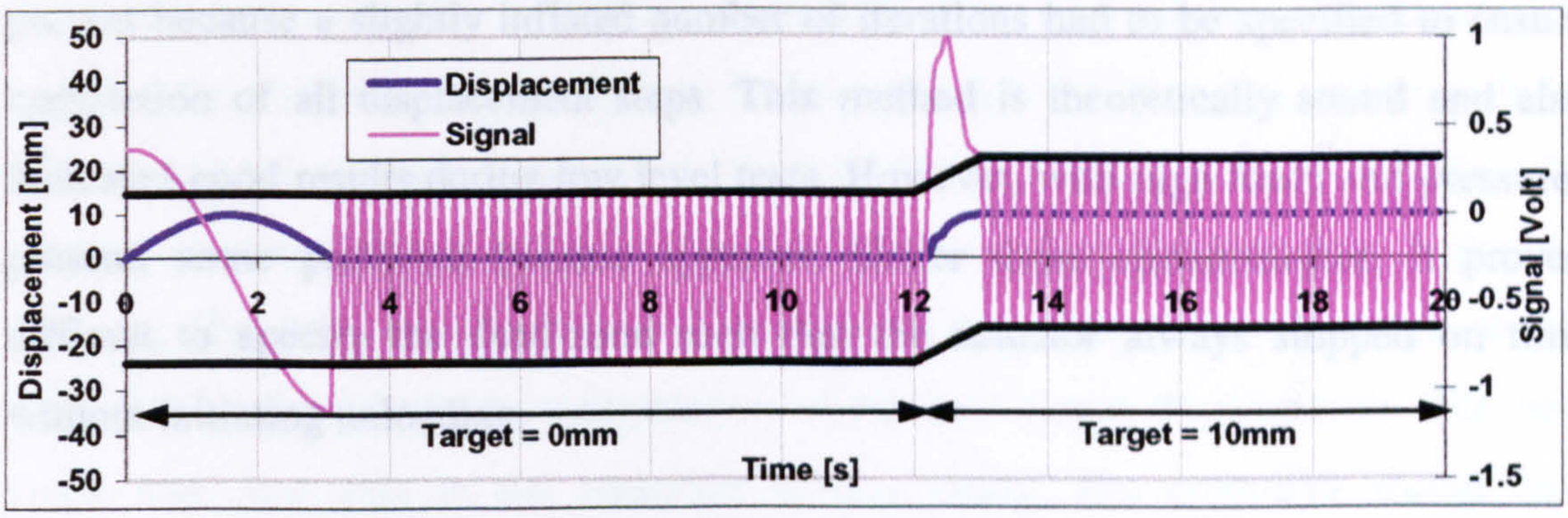


Fig. 5.6 Combined proportional-integral control.

5.3.5 Step continuity

While the previous sections all discuss the implementation of a single displacement step, this section elaborates on what happens between the steps and the displacement and signal profile created over the step change. During the development of the controller, it was found that the results could be particularly sensitive to details within this part on the implementation.

The so-called "classic" pseudodynamic implementation assigns a specific "hold period" at the completion of each displacement step. During this period, force measurements are taken and communication with the hardware carried out to enable computation of the next displacement step. Only once the time integrator has computed the next step, and communicated this to the ramp generator and controller, will the actuator again be set in motion after a stationary period of the order of 1s. Although the implementation system developed by the author, as described in Chapter VI, differs substantially from the classic one (e.g. Negro 1997), a similar discontinuous implementation system was first adopted.

The earliest versions of the control system allowed a certain number of iterations per displacement step. This was one of the criteria considered for the completion of a displacement step, with the other two tested procedures corresponding to the completion with displacement within tolerance of target and the completion with displacement equalling or passing the target.

Allowing a certain number of iterations per step effectively provided a hold period within which the actuator did not move appreciably. The hold period was effectively present because a slightly inflated number of iterations had to be specified to ensure completion of all displacement steps. This method is theoretically sound and also indicated good results during low level tests. However, with high loads and pressures present, some problems became apparent. Under these circumstances, it proved difficult to specify the dead-zone such that the actuator always stopped on time without initiating unloading.

For the tests on the reinforced concrete structure, as described in section 3.3.3.1, the force measurements had to be taken using an internal pressure transducer, as explained in section 3.4.1.1. This system displays the unfortunate effect that minor unloading during negligible displacement changes results in substantial changes in the force readings. As the implementation system described above could not avoid slight unloading, the force measurements could not be obtained with sufficient accuracy. This, in turn, resulted in an unrealistic response and required improvement.

The other two methods of completing a displacement step both limit the duration of the hold period substantially. These complete the step either: (a) once the displacement is within a certain tolerance of the target or (b) when the target is actually reached or passed. The number of iterations thus varies from step to step, and the actuator is not kept stationary for longer than necessary. As soon as the tolerance or target is reached, the force measurement is taken and passed to the time integration algorithm.

Method (a), using the displacement tolerance, was quickly abandoned simply because the displacements corresponding to the final force measurements were consistently somewhat smaller than the target. Method (b), requiring the displacement to actually be met or passed, turned out to provide superior final displacement accuracy. However, this did not solve the problem with unloading and inaccuracy in force measurements. When the target was met, the signal would still jump to the opposite limit of the dead-zone to avoid overshoot.

In order to improve the force measurements, using the force reading in the iteration prior to reaching the target was considered. Although theoretically sound, a better solution was found following a substantial redesign of the controller. It was now set up such that the controller looked into the details of the next displacement step before applying firm signals. This could only be done through the close integration with the time integrator as will be detailed later in Chapter VI. On completion of the displacement step, the controller now took the force readings but no action to stop the actuator. The force measurement was sent to the time integration algorithm, which immediately carried out the computation of the next target displacement. The new target was then sent to the controller without delay. The controller would now consider the next target in relation to the present position of the actuator. If the target were farther ahead, the signal would be altered only such that the actuator moves faster. In practice, this enables continuous actuator motion, with no stopping or unloading; the actuator only slows down when passing through the target. If, on the other hand, the next target turned out to be in the opposite direction, the controller would immediately initiate a signal jump to the opposite side of the dead-zone. This would stop the actuator and commence movement in the opposite direction.

However, such a semi-continuous method of implementation displays disadvantages as well as the advantages mentioned above. When the next displacement step is in the opposite direction of the one the actuator is currently moving in, the controller has to wait for the new displacement target before the actuator can be stopped. This may potentially lead to a displacement overshoot. The force measurement has already been taken, so it will not suffer from the overshoot. However, the continued motion may lead to some further damage taking place in the specimen. It is believed though, that the delay involved in allowing an extra iteration to take place will be so minor that this will not adversely affect the accuracy of the test. The integration between the controller and the time stepping algorithm is discussed in detail in the following chapter.

5.3.6 Transducer compensation

Displacement measurements are taken through an internal, analogue displacement transducer, an LVDT, as described in section 3.4.1.2 The total stroke of the LVDT is 120mm, indicating that the required ±50mm should be within a reasonably linear

region. When calibrating the device however, errors of up to 2.5mm remained after optimal adjustments. Such errors are clearly unacceptable, as displacement accuracies of the order of 10-100μm are typically required for reliable pseudodynamic tests.

In order to reduce the error stemming from the non-linearity of the LVDT, software compensation could be employed. By tracing the difference between the displayed and the actual displacement every 10mm of the range, a curve of the absolute error was plotted. A sixth order polynomial was fitted as close as possible to all the points, giving the function displayed in equation (5.1).

$$Y = -1.31224E\text{-}06x^6+9.19900E\text{-}05x^5-3.75415E\text{-}05x^4 -1.45298E\text{-}03x^3-6.69400E\text{-}04x^2-2.96456E\text{-}01x+1.17532 \quad (5.1)$$

The trace of the error and compensation function can be seen in figure 5.7 below. When converting the electrical signal to a displacement, this compensatory function could simply be added to the otherwise linear relationship between voltage and millimetres.

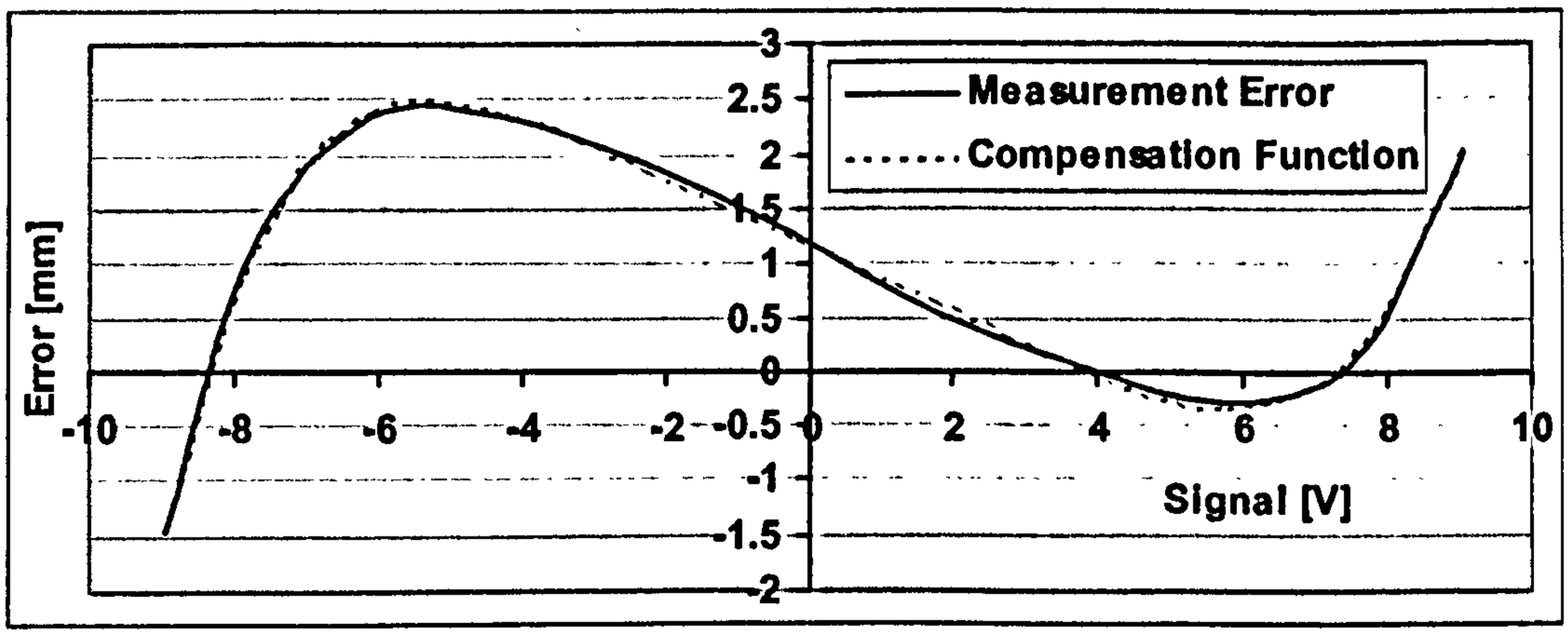


Fig. 5.7 LVDT error and compensatory function

The compensation function was incorporated into the controller itself, and adjusted the displacement signal as soon as it was read. This reduced the non-linearity error from around 2.5mm to a maximum of around 100μm.

5.4 INTEGRAL FORM FORCE MEASUREMENTS

Both the explicit and implicit formulations of the integral form of the Newmark time stepping scheme require a measure of the time integral of the restoring force over each time step. This variable forms part of the explicit displacement predictor, as detailed in Chapter IV. In this section however, only the method of obtaining the force integral will be discussed.

In order to compute the time integral of the restoring force over each time step, two main methods were envisaged. A numerical integration scheme that operates at the iteration rate of the controller (the multipoint method) could clearly be used, as a continuous stream of force readings is available. Alternatively, a similar scheme using only the force readings at the start and end of the time step (the two point method) could be implemented with significantly less complexity. However, this would render the main features of the integral form methods redundant.

During the initial development of the implementation system, the method of using only the first and last force readings in a step (the two-point method) was used. This system allowed the integral to be computed upon completion of the step with the simple trapezoidal rule expression in equations (5.2), where Δt is the duration of the time step and r_n and r_{n+1} are the restoring force values as the start and end of the step, respectively. The two-point method is not required to be integrated with the controller, and may therefore be coded more efficiently elsewhere in the implementation system, as explained later in Chapter VI.

$$\int_{\Delta t} r dt = \Delta t \frac{(r_n + r_{n+1})}{2} \tag{5.2}$$

As only two force points are used, the method assumes linear variation of the restoring force over the duration of the time step. When employing large steps during tests on highly non-linear structures, this may introduce considerable error as detailed in Chapter IV. To avoid this approximation, the multipoint integral computation system may be used.

The multipoint method of computing the time integral of the restoring force operates at the iteration rate of the controller. Essentially, it obtains a measure of the restoring force during each iteration, and computes the integral based on this, again using the trapezoidal rule. In order to carry out the integration and provide the variable without delay upon completion of the displacement step, the multipoint integrator is coded directly into the controller. The exact details of this are documented later in section 6.2.2.4.4.4.

In principle, the method computes a portion of the full integral in each iteration in a sub-stepping routine. It uses the restoring force values from two successive iterations and the time between them to compute the portion of the integral; these portions are then summed up for all the iterations to yield the full time integral of the force over the step. The variable is thus computed as in equation (5.3) where r_m and r_{m+1} are the restoring forces at two successive sub-steps and t_m and t_{m+1} are the corresponding points in time.

$$\int_{\Delta t} r dt = \sum_{m=0}^{\Delta t} \frac{(r_m + r_{m+1})(t_{m+1} - t_m)}{2} \tag{5.3}$$

While the assessment of the restoring force variables is straightforward as they are simply the measured quantities, the definition of the time variables is somewhat more complex. Real-time clock measurements cannot be used for two main reasons: firstly the displacements are generally not imposed in real-time as discussed in section 2.3 and secondly the velocity profile generated is not generally linear, as discussed in section 5.3.4. The effect of these two facts is that the measured time bears no correspondence to the theoretical time position of the sub-steps.

Equivalent time points must be obtained for each sub-step. These points can be defined by examining the proportion of the completed displacement step and, by assuming constant velocity, this will correspond to the proportion of the completed time, as demonstrated in figure 5.6.

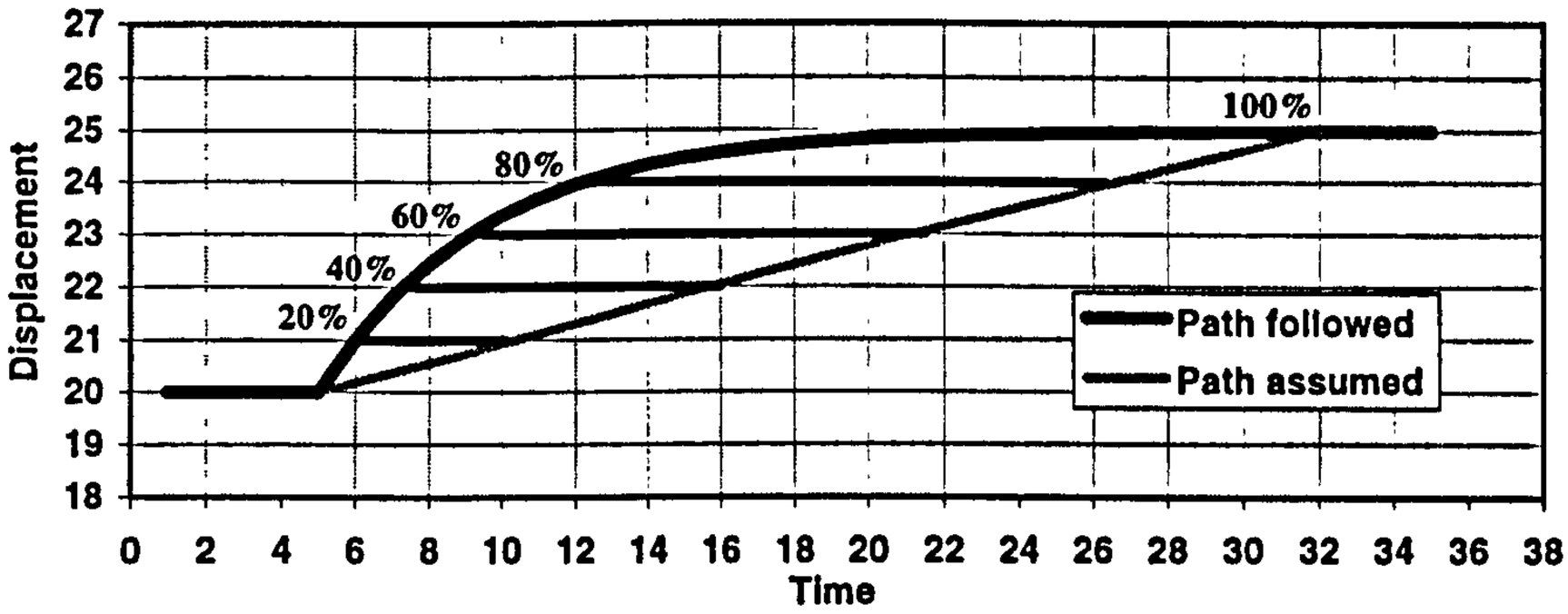


Fig. 5.6 Equivalent time points using proportion of completed step

The controller will typically create a velocity profile that is initially fairly steep to attempt rapid completion of the displacement step, but flattens out when approaching the target displacement to ensure minimisation of the overshoot. This implies that, for example, 80% of the displacement step is completed in just 7 of the in total 27 time units, or 26% of the time. When computing the equivalent time point assuming constant velocity, 80% of the time should have passed at 80% completion of the displacement, or in other words around 22 time units. Thus, for each control iteration, a time value is assigned corresponding to the proportion of the displacement step completed and the time integration step size.

The time position of each sub-step has to be computed based on the fraction of the completed time step. As most time stepping schemes assume constant velocity over a time step, this criterion may also be applied when computing the equivalent time positions of the sub-steps. By using the measure of the displacement obtained concurrently with the restoring force, the equivalent time position can be found. Based on the assumption that the fraction of the total displacement imposed corresponds to the fraction of time passed, the time positions of the sub-step force measurements, t_m, is defined as in equation (5.4), where d_m, d_n and d_{n+1} are the displacement in the current sub-step, the displacement at the start of the time step and the displacement at the end of the time step, respectively.

$$t_m = \Delta t \frac{(d_m - d_n)}{(d_{n+1} - d_n)} \tag{5.4}$$

The multipoint method of obtaining the time integral of the restoring force is under some circumstances particularly sensitive to experimental errors. While functioning well when the displacement step is clear and of a significant magnitude, it is more likely to display weaknesses when the actuator is kept stationary. This is especially the case during the start-up procedure of each test. Inaccurate force representations in the first step can adversely affect the entire response. An example of this can be seen in figure 5.7 below, where the same system has been subjected to an identical ground excitation a number of times.

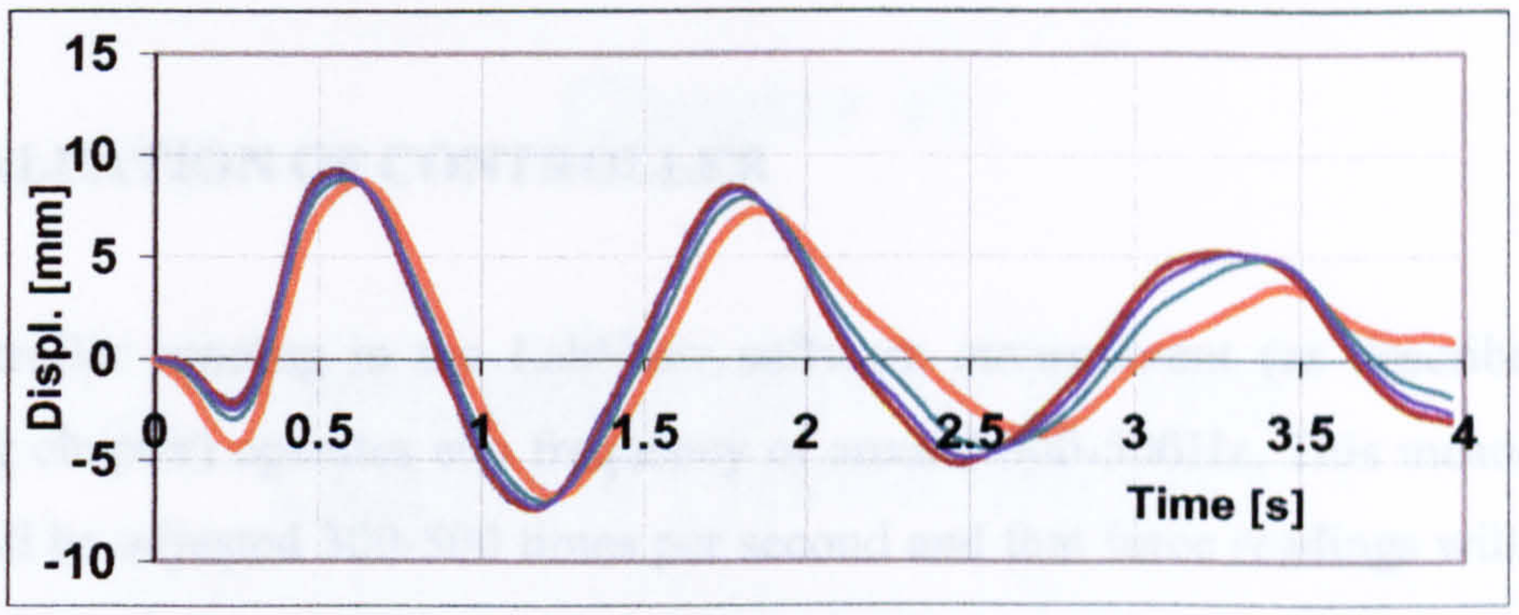


Fig. 5.7 Differential response under same excitation

As figure 5.7 shows, the repeatability of the tests is inadequate, due to variations in the computation of the integral of the restoring force in the first step. To mitigate this problem, the first time step could simply employ the two-point procedure instead. This would ignore any motion and electrical noise during the start-up and provide a more constant measure of the force integral. The improvement offered by using the two-point method in the first step can be seen in figure 5.8, showing 12 different responses to the same excitation. For information, both variables are logged through the entire tests and may be inspected during the tests or after completion.

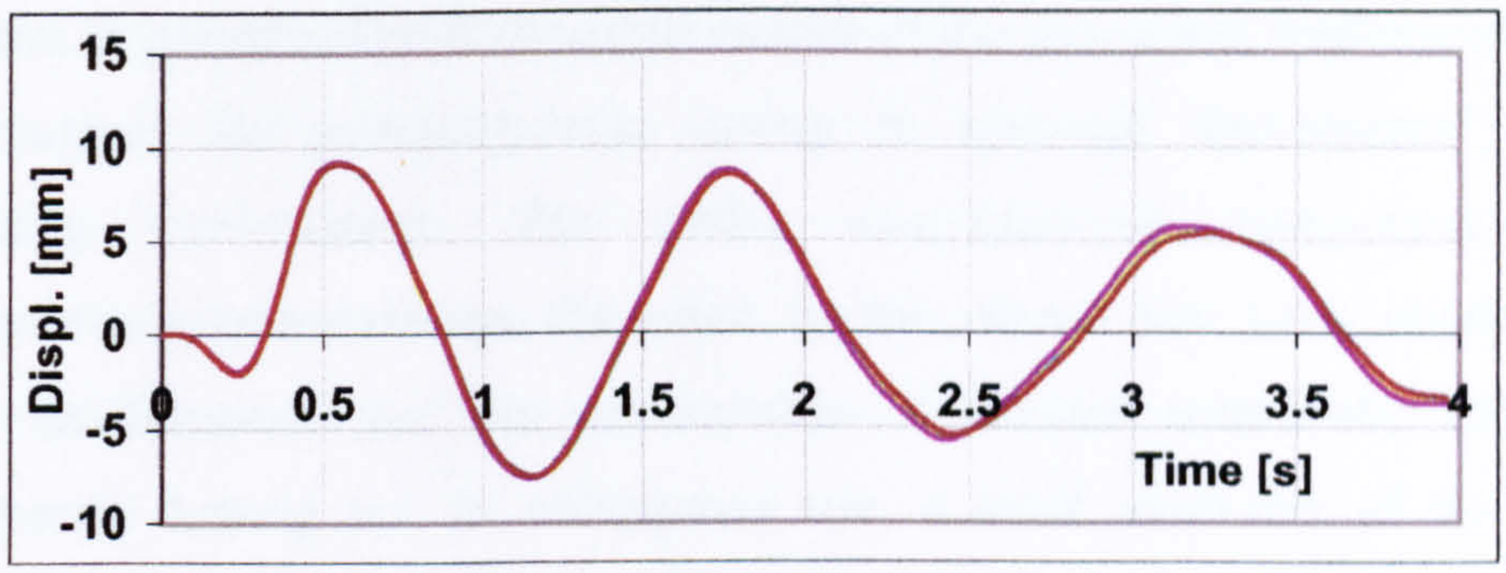


Fig. 5.8 Approximately equal responses to same excitation

5.5 EVALUATION OF CONTROLLER

The controller running in the LabView software environment (as described in the following chapter) operates at a frequency of around 300-500Hz. This means that the signal will be adjusted 300-500 times per second and that force readings will be taken with the same frequency. As explained later, implementation times may vary greatly, but modelling 4s of a real system response accurately using 0.04s time steps will require experimental implementation time of the order of 20s. This thus allows 0.2s per step, indicating that on average 60-100 iterations are provided on average. This should be more than sufficient to trace any non-linearities in the restoring force, and it represents a fairly modest, but sufficient number of control iterations.

The implementation time of 20s mentioned above can be considered a fairly good compromise between speed and accuracy. 20s represent 1/5 of real time strain rates, and with this speed a minimal accuracy of 200μm can be maintained. Reducing the speed to half of this will provide an accuracy that approaches that of the transducer, around 100μm. The maximum accuracy obtainable by the controller is of the order of 30μm, not taking transducer errors into account. The overall accuracy achieved depends not only on the implementation speed. The natural frequency of the tested structure, the time step size employed in the time stepping algorithm and to an extent the amplitude of the oscillations may all influence the accuracy. A full discussion on the performance of the implementation system on the whole will be provided later in Chapter VII.

Chapter VI:

IMPLEMENTATION IN THE LABVIEW ENVIRONMENT

This chapter is concerned with the development of the execution, implementation and control routines for pseudodynamic testing in National Instruments' LabView programming environment. The entire computational component of the pseudodynamic implementation discussed in this thesis has been created in the LabView environment. As this environment is neither commonly adopted for pseudodynamic testing nor in widespread use, a brief overview of the features, structures and logic behind this so-called *graphical programming* language will be provided. Following this, a fairly detailed account of the entire computational system will be presented. This includes the coding of the time stepping schemes and the control algorithms, as well as the execution system, which ties all the components together.

6.1 THE LABVIEW ENVIRONMENT

National Instruments' LabView 5 environment was chosen for the implementation of the computational components of the pseudodynamic tests for a number of reasons. First of all, the environment is specifically designed for hardware communication, which means that setting up the communication channels is a significantly less complex matter than it would have been with a low-level language. Secondly, being intended for use in laboratories, the environment offers excellent opportunity for creating visual representations and interactive screens. Lastly, although possibly not appreciated by the traditional programmer, the graphical form of the programming offers a great advantage by enabling the entire code and logic path to be visualised at the same time.

6.1.1 The graphical programming environment

In the LabView environment, programs, or *virtual instruments* as they are referred to, are built up using the so-called graphical code. This, to an extent, implies that the code is *drawn*, rather than *written* as in conventional languages. Examples of such graphical code are provided throughout this chapter.

In LabView, all the standard features of programming languages are available. These include structures like *for* and *while* loops or *case* and *sequence* structures, as elaborated on in section 6.1.3, conditional and logic statements as well as local and global variables and array, cluster and string operations to mention a few. Additionally, a vast range of communication, data acquisition and display functions exist.

6.1.1.1 Wired connections

Variables are in principle transmitted from one place in the program to another through *wires*. Once a variable is defined, a wire must be connected to it and "stretched" to the point(s) where it is used. The system resembles to a degree an electronic circuit, except the wires can carry any form of variable (numbers, arrays, strings, booleans).

Once defined, variables carried by wires can be split to enable their use on several locations; however they cannot be combined unless a numerical (or other applicable) operator is used. Depending on what format the variable is in, the colour and pattern of the wires changes for visualisation purposes. For example, a simple numerical variable will take the shape of a thin orange line, except if it is an integer, in which case a blue line is used. A 1D array of numbers will create a thick orange line and a 2D array of numbers a double orange line. Boolean variables are carried by green lines, while for example strings form pink lines. This system makes the code somewhat easier to follow, as one is immediately aware of what form of variable is carried by each wire.

The example shown in figure 6.1 is among the easiest imaginable executable code. Here, a variable is defined in the box denoted "input" and passed to the box denoted "output". If the program is run continuously, the output window will at any given time display the number defined in the input box. However, both the definition and display of variables here take place in the interactive *front panel*, as will be discussed in section 6.1.2. The figure on the left shows the colour scheme employed with normal double precision variables, while the figure on the right shows the blue colour adopted for integer variables.

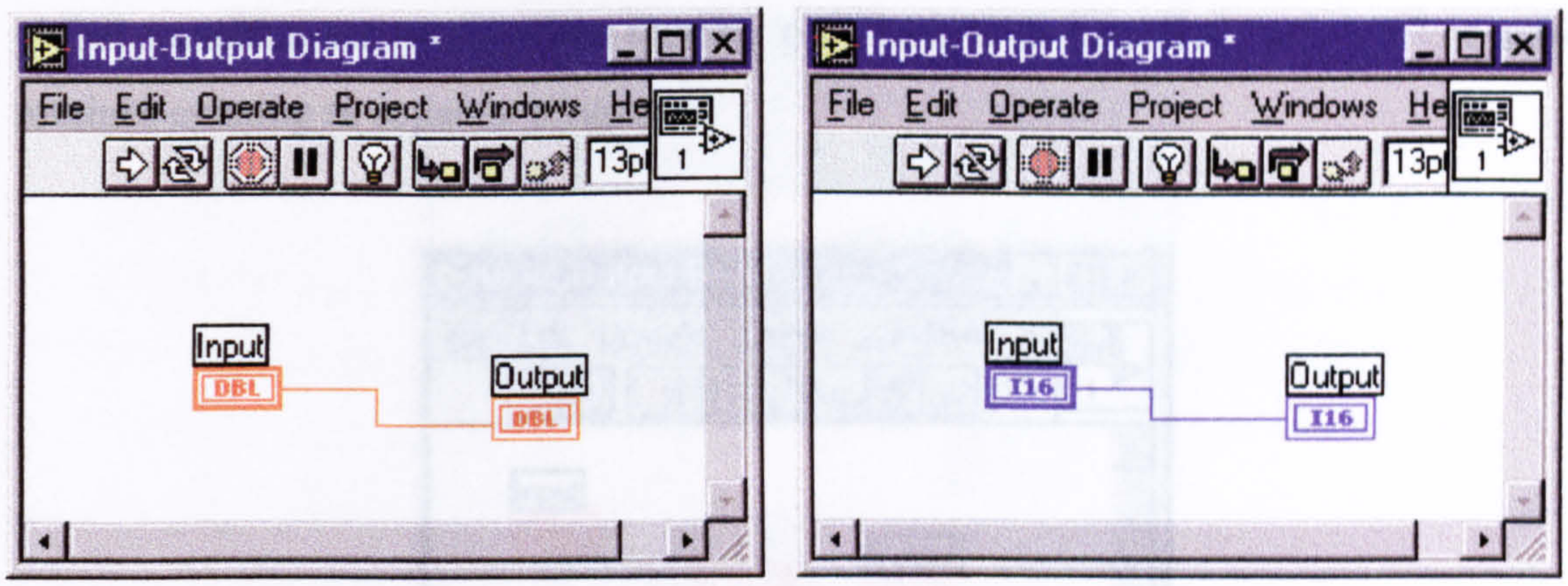


Fig. 6.1 Wired connection between "input" and "output" boxes

6.1.1.2 Wireless connections

In some instances, with more complex programs, wired connections may not suffice in transferring the variable to the right place at the right time. A typical example of this is when loops are employed. In this case, all wired variables have to be transmitted to the loop before it commences, and no wired output takes place before the loop has ceased to be operational. If it is desired to transfer information in or out of a loop while looping, so-called *local variables* may be used. In the context of graphical programming, these are essentially "wireless" connections.

When using local variables, the variable may be read from or written to at any point in time and from any place within the program. Whenever the variable is read, it will always read the most up to date definition, no matter where this is created. This method of transmitting variables clearly displays some advantages over the wired connections, but it does not ensure a correct data flow and effectively complicates the program considerably. A basic example of how the local variables are used can be seen in figure 6.2 below. This set-up will perform exactly the same task as in the example above, but using a local variable of the input.

In addition to the local variables, wireless communication may be realised through *global variables*. These are essentially the same as the local variables, except they may be written to and read from several different programs at the same time. The

global variables are useful when running programs with so-called "sub-VIs" or sub-routines existing as separate programs.

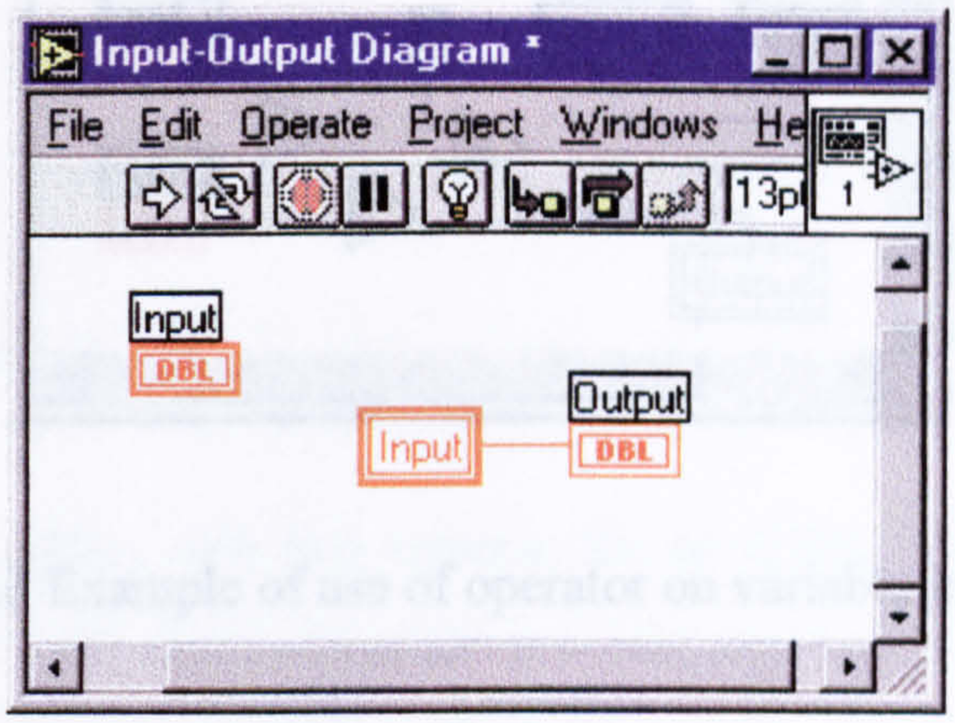


Fig. 6.2 Use of wireless local variables

6.1.1.3 Boolean, numerical and comparison operators

A vast range of operations can be carried out on the wires directly. These include all imaginable boolean statements, conditional and comparison operators as well as the full array of numerical operators. If wireless variables are used, a "receiver" must be created and connected by a wire to the operator.

The above operators are applied by simply connecting wires to their input and output. Depending on the operator concerned, the wire may contain variables in any applicable format. Most numerical operators will work with numbers, arrays and clusters, while the boolean operators naturally work on boolean variables and integers, but also on arrays and clusters of booleans and integers.

A simple example of the use of the operators described above can be seen below in figure 6.3. In this example, two numbers are entered as *Input 1* and *Input 2*. These are then multiplied together and the square root taken of the result. This is then compared to the integer *2* to see if *2* is larger or equal to the result or not. The result from this is a boolean *True/False* constant. This is then converted to a *zero/one* integer, which is subsequently transmitted as a local variable called output. Finally, this variable is received and connected to an output display.

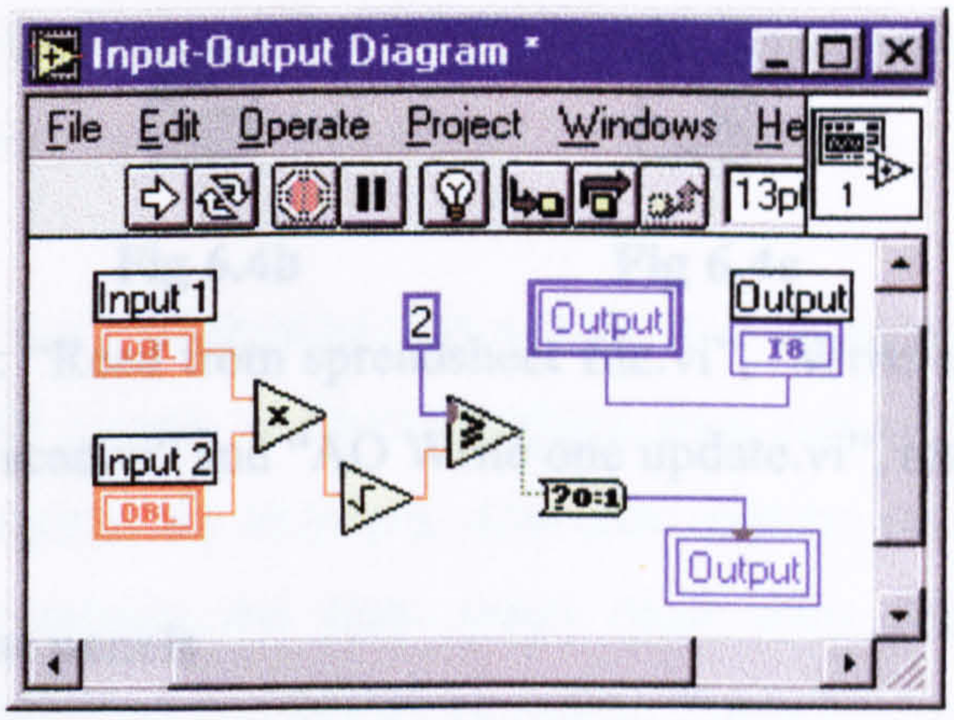


Fig. 6.3 Example of use of operator on variable in wires

6.1.1.4 Other functions

LabView offers a range of functions in addition to the standard operators mentioned above. Some of these are concerned with data acquisition, communication, timing and file I/O while others form a range of analysis tools. Although the majority of these functions are highly specialised and rarely utilised even by advanced users, some are vital for fairly basic programming and require explanation.

Among the functions widely used in the programming of the pseudodynamic implementation and control system are ones related to file input and output and to the analogue communication with the instrumentation. For file input and output, inherent functions called "Read from spreadsheet file.vi" and "Write to spreadsheet file.vi" are used, although the read and created files are ASCI files and not spreadsheet files as such, as discussed in sections 6.1.4.1 and 6.2.2.4.5. In the graphical environment, the functions are denoted as in figures 6.4a and 6.4b, respectively. For communication with the instrumentation and the servo valve, functions called "AI Read one scan.vi" and "AO Write one update.vi" are used, where AI and AO refer to *Analogue Input* and *Analogue Output*, respectively. The representation of these functions can be seen in figures 6.4c and 6.4d, respectively. A detailed explanation of the required input and output formats will be provided when and where the use of the functions are documented.

Fig 6.4a

Fig 6.4b

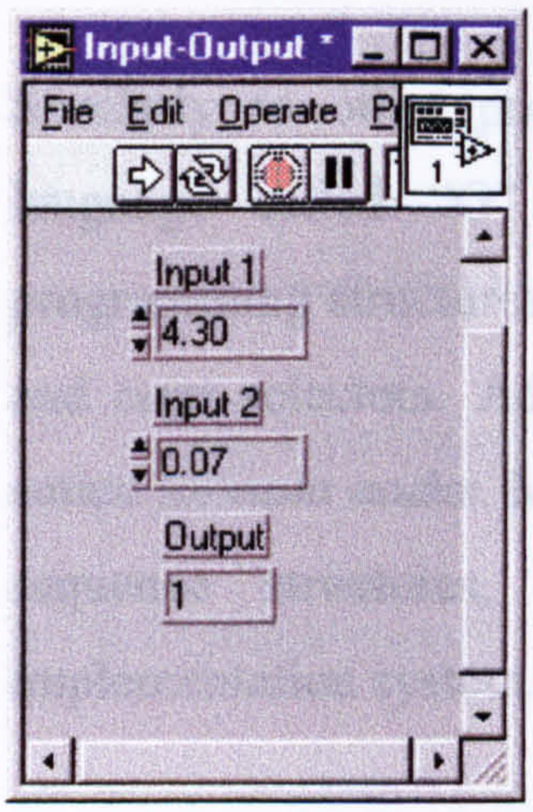
Fig 6.4c

Fig 6.4d

The functions: "Read from spreadsheet file.vi", "Write to spreadsheet file.vi", "AI Read one scan.vi" and "AO Write one update.vi", respectively

6.1.2 Interactive front panels

LabView operates in effect with two screens: the so-called "Block Diagram" which contains the graphical code discussed in the previous sections and the so-called "Front Panel" which will be discussed in this section. Essentially, the front panel is where the interactive input and output takes place.



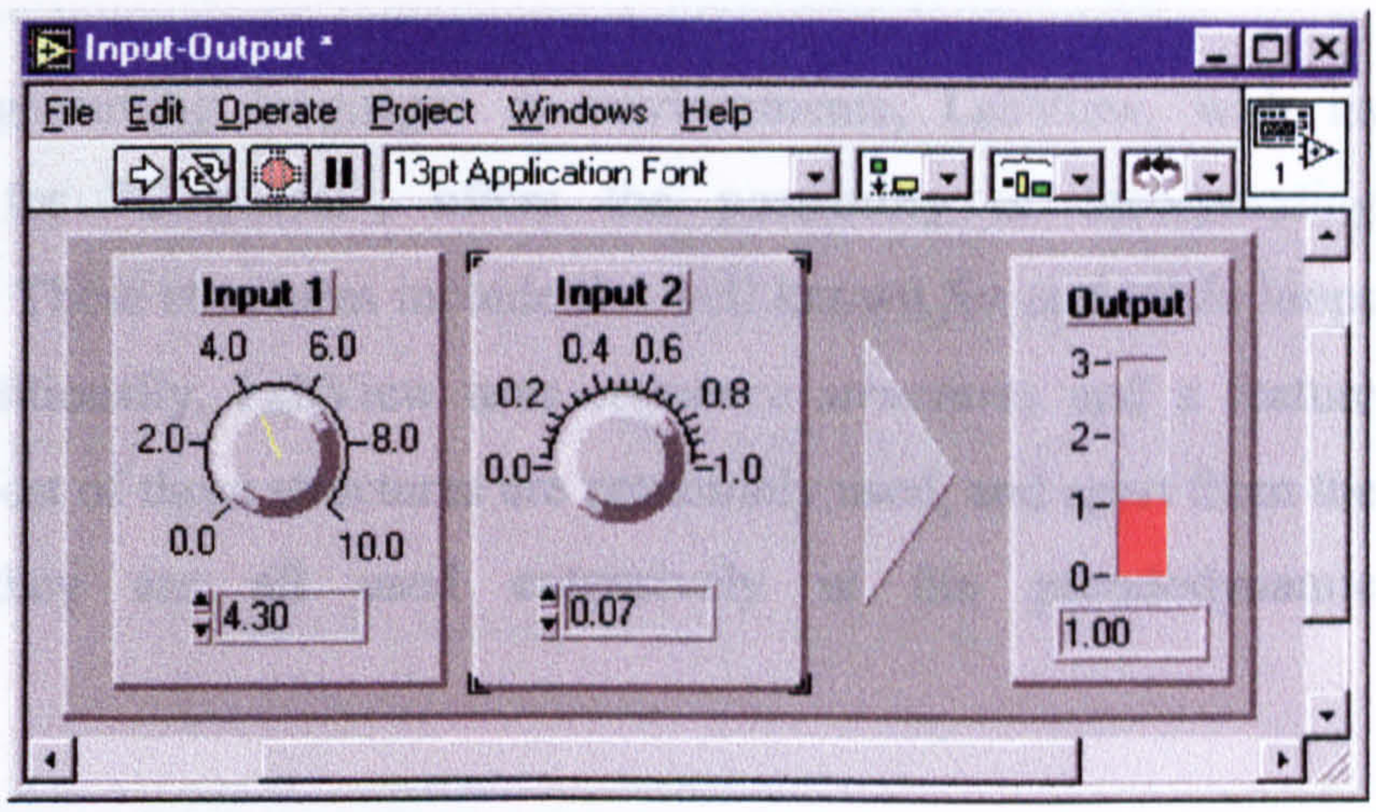


Fig. 6.5a & Fig.6.5b Examples of front panels with two inputs and one output in simple and visually elaborate forms, respectively.

In the examples given in sections 6.1.1.1 to 6.1.1.3, the input and output points were simply displayed as boxes, referred to as *terminals* in the block diagram. In the LabView environment, each terminal has a corresponding *indicator* or *control* on the front panel, depending on whether it represents an output or input, respectively. The controls and indicators may take on a large variety of shapes. While the basic may be so-called digital controls and digital indicators, simply displaying the number, more visual formats like dials, gauges and meters may be used for the same purpose. An example of two front panels corresponding to the example in figure 6.3 may be seen

in figure 6.5, where 6.5a represents the most plain display screen and 6.5b a more visually elaborate alternative. The two perform identical tasks.

The front panel for a pseudodynamic execution and control system will be quite extensive. If a flexible system is desired, which does not require alterations to the main code when changing, for example, structural system properties, time stepping schemes and control details, the front panel must offer definition and selection possibilities for a substantial number of variables. Furthermore, as a test progresses, an array of graphical and numerical output will be produced, requiring additional display facilities. The front panels for the actual pseudodynamic implementation system created will be discussed in section 6.2.2.

6.1.3 LabView programming structures

Similarly to other programming languages or environments, LabView, with its language called "G" for "Graphical", offers the possibility of incorporating programming structures. These structures include the well known *for* and *while* loops and *case* selectors. Additionally, LabView uses *sequence structures* and a feature called *formula nodes*. Most of these structures are commonly used, and apart from the sequence structures, they are all used extensively in the pseudodynamic implementation system.

6.1.3.1 Loop structures

The *for* and *while* loops function in the conventional manner; *for* loops loop a given number of times while *while* loops loop until a condition is satisfied. However, as the loops are created graphically, some further explanation will be provided.

The loops are defined by creating a rectangular loop box in the block diagram. All operations that are to take place within the loop are placed inside, and connected up with wires. Variables required within the loop are connected by wires passing in through the loop boundaries, while information generated within the loop is passed out through the boundaries. It is important to note that the loop will not initiate before

all input has become available, and no output will be passed out before the loop has terminated.

Simple examples of the *for* and *while* loops can be found in figures 6.6a and 6.6b, respectively. In the *for* loop, the N determines the number of times the loop is executed, while in the *while* loop the sets the condition for continuation, so the input required for these are integers and boolean constants respectively. In both cases, the i provides the number of loop cycles carried out at any stage. Lastly, the and the are so-called *shift registers* which pass information from one cycle to the next. In this example, both loops are essentially executed 100 times, adding 1 to the variable entering on the left each time.

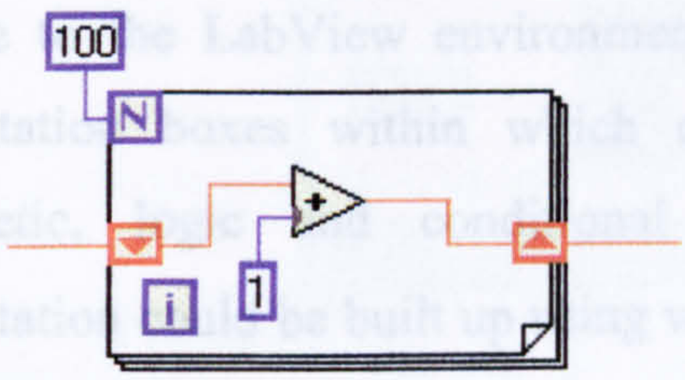


Fig. 6.6a Simple *for* loop

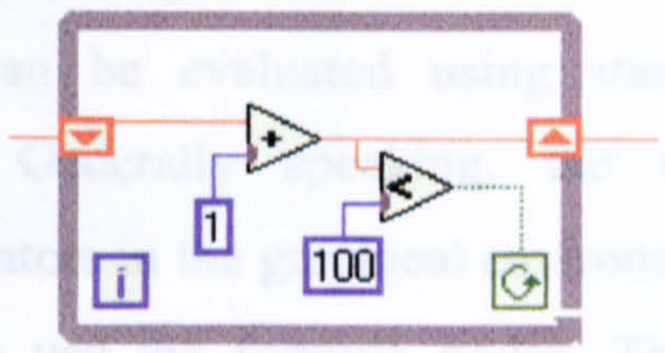


Fig. 6.6b Simple *while* loop

6.1.3.2 Case selectors

Case selectors are also important in LabView, while in compiler languages, the same functions may be constructed using conditional statements. Essentially, the case selectors enable the selection of a set of variables and operations according to a boolean or integer constant.

The case selector can be used to select either between two options or a number of options. If only two options are available, a boolean constant will be used. An example of this has been taken from the main implementation algorithm, where the option of saving an array to disk is offered. Figure 6.7 shows the *true* and *false* options, where if *true* is selected, the variable is saved, but if *false*, no action is taken. Frequently, it may be desirable to select from more than two options. There is no limit to the number of cases that may be included in a case structure. Rather than being

denoted as *true* or *false*, the cases will take on integer values from 0 to *n*, i.e. 0, 1, 2, 3, 4, n, where *n*+1 equals the number of cases.

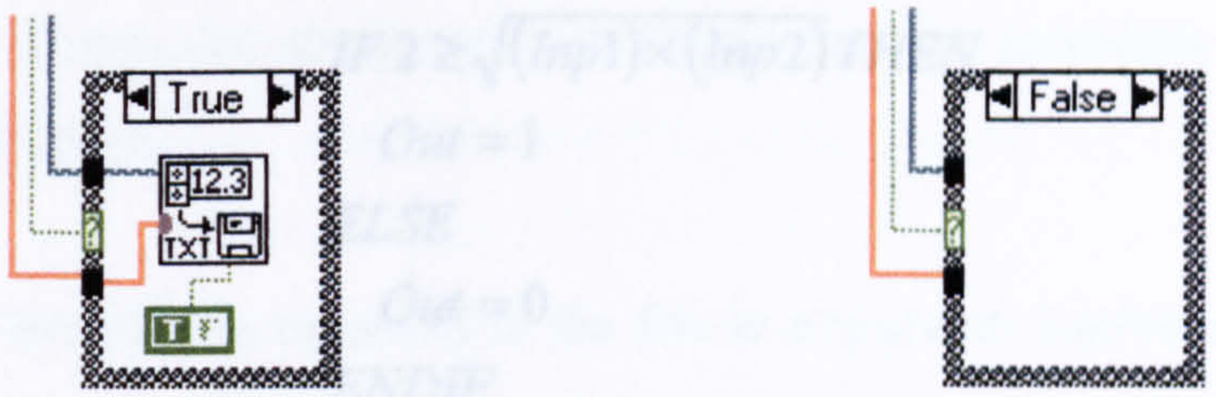


Fig. 6.7 Selector for saving array to disk

6.1.3.3 Formula nodes

Unique to the LabView environment, are the so-called *formula nodes*. These are computation boxes within which expressions can be evaluated using standard arithmetic, logic and conditional statements. Generally speaking, the same computation could be built up using wires and operators in the graphical environment, as in figure 6.3, but it may be advantageous to use the formula nodes. This is primarily because intricate computations easily become overly complicated to follow in the graphical environment. With formula nodes, long and complex computations, using a range of variables, can be concentrated and visualised a single box.

For all but the most basic LabView programming, advantage can be taken by using the formula nodes. During the development of the pseudodynamic implementation system, extensive use was made of the formula nodes. The formula nodes are defined by creating a rectangular formula box, then creating connectors for the variables around the perimeter. Each of the connectors need then be wired to the to appropriate variables. In all programs described here, the convention has been to position the input towards the left hand side of the box and the output on the right hand side.

As a simple example of the use of the formula nodes, consider the block diagram in figure 6.3. The calculation can be reconstructed using a formula node containing all the mathematical operations used. The two input variables are wired up as *Inp1* and *Inp2* and the output as *Out*. *Out* is then defined as the quasi code expression in

equation 6.1. The *Output* variable is evaluated using a formula node is figure 6.8, performing an identical task to that in figure 6.3.

$$\begin{aligned} &IF\ 2 \geq \sqrt{(Inp1)\times(Inp2)}\ THEN \\ &\quad Out = 1 \\ &ELSE \\ &\quad Out = 0 \\ &ENDIF \end{aligned} \tag{6.1}$$

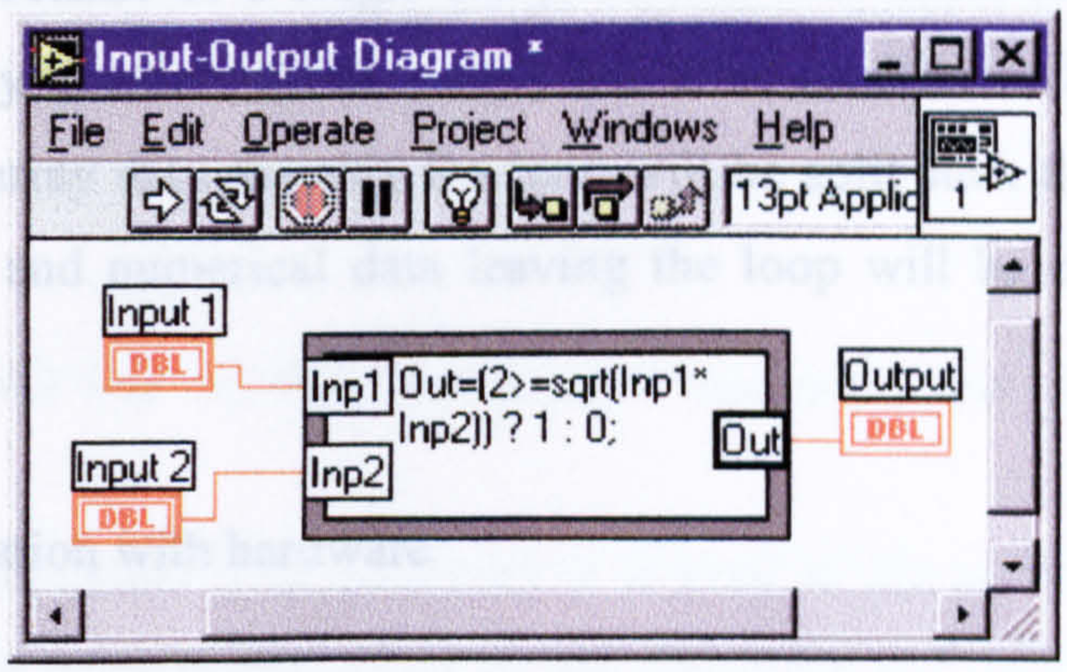


Fig. 6.8 Use of formula node for assessing equation (6.1)

6.1.4 Input and output formats

LabView greatly benefits from the fact that communication with both external files and hardware is arranged through inherent macros. The most commonly used ones have been introduced in section 6.1.1.4, but require some further explanation. The following sections will elaborate on the operation methods and required formats for the use of these macros.

6.1.4.1 Communication with external files

For the communication with external files, a range of the order of 50 macros are available for configuring, opening, closing, creating, altering, reading and writing files. For the purpose of loading up external accelerogram and displacement history files and carrying out data logging in the pseudodynamic implementation, two macros were found to function well. These were the "Write to spreadsheet file.vi" and the

"Read from spreadsheet file.vi", shown in figures 6.4a and 6.4b. Both offer the option of transposing the data and appending the data to an existing file. If the "append to file option" is chosen, the macro will open the file, add in the new data at the end and then close the file. If not, the macro will create a new file according to the path and filename wired into it.

The format of the data to be saved to the file is important. LabView will not accept single numerical data; the data has to be put into either a 1D or 2D array. However, the 1D array can consist of a single entry, so that individual numbers can still be saved. A built array function, , simply has to be used on such data. When using the *for* loops, 1D array data entering the loop will be split such that one entry enters during each cycle and numerical data leaving the loop will be compiled into a 1D array.

6.1.4.2 Communication with hardware

An impressive range of macros are available to the developer in the LabView environment for initiating and configuring communication channels and interpreting and conditioning instrumental data. This simplifies the setting up of the hardware and channels greatly, but the process is still fairly involved.

Following experimentation with a range options for analogue communication, the most suitable macros for communication with the instrumentation utilised were found to be the "AI Read one scan.vi" and the "AO Write one update.vi" macros, as shown in figures 6.4c and 6.4d, respectively.

The "AI Read one scan.vi" macro essentially carries out an instantaneous, untimed reading of the selected channels on a particular device. In the experimentation described in this thesis, only one device was used, the high speed communication card. As the card had around 20 input and output channels, this could easily handle all the communication. In general, the channels the software has to read are up to two displacement channels and up to two force channels. The "AI Read one scan.vi" macro enables reading of these channels at the same time. The selected channels must be input as an array of strings, as shown in figure 6.9 showing the macro with all the

connectors. The connectors annotated in grey are optional, so only the remaining four black connectors on the left hand side require input. These furthermore apply default values (shown in the parenthesis) if no variables are connected. The output from the macro, the so-called scaled data, is always in the 1D array format, even if only one channel is being read. To extract the data from each channel, an *index array* operator is used.

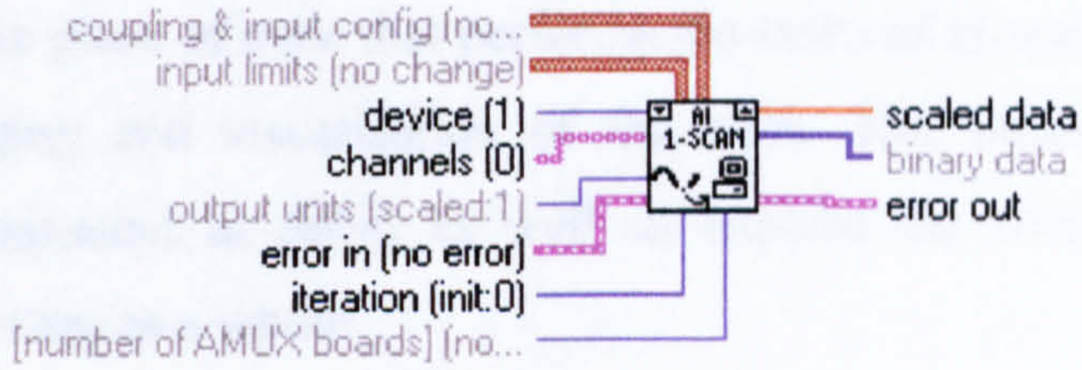


Fig. 6.9 The "AI Read one scan.vi" macro with connectors

The output communication with hardware is limited to the servo valve. This is the only variable that is actually controlled by the algorithm and ensures that the desired target displacements are reached.

When sending information to the communication card for the servo valve, the "AO Write one update.vi" macro is utilised. The effective function of this macro is to generate the signal defined in the controller, and to continue generating this until an update is received. Again, only some of the connectors need to be wired up, and others allow default values to be used.

The 1D array format is used here for the input data, so as only one channel is in use, the data simply has to be built into a 1D array with only one entry in it, using the *build array* operator. Figure 6.10 shows the macros with all the potential connectors.

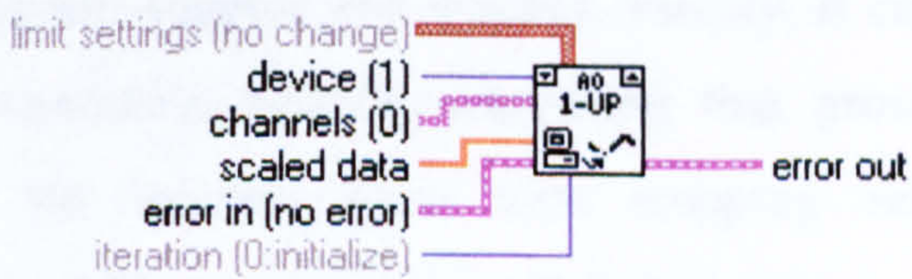


Fig. 6.10 The "AO Write one update.vi" macro with connectors

6.2 PSEUDODYNAMIC IMPLEMENTATION IN LABVIEW

An implementation system that executes and controls the pseudodynamic tests documented in this thesis has been created within National Instruments' LabView environment. The system is fully software based and operates without any hardware controller. Only a single high-speed communication card converts the instrumental signals to digital numbers used in the software and vice versa. The system has been designed as a single piece of code that performs the tasks of execution, time stepping, control, data logging and visualisation of the tests. The following sections will describe each component in detail as well as explain the overall data flow and operation of the system as a whole.

6.2.1 Software based implementation system

Experimental apparatus of the type employed for pseudodynamic modelling would normally be controlled using specialised hardware controllers, as explained in section 2.2. These may be either stand-alone devices or, more recently, programmable cards for PCs. In any case, the actual control and signal generation is carried out remotely from the computer executing the tests. The execution routine on the main computer will supply a target displacement, while the controller ensures the target is reached with a prescribed velocity profile.

The pseudodynamic implementation system described here does not utilise a hardware type controller. Instead, the control routine is programmed in the LabView environment as a part of the full implementation program. The controller thus runs on the CPU of the executing computer. The advantages of developing the controller as a piece of software are substantial. First of all, it eliminates the requirement for providing a costly piece of hardware. Secondly, programming and alteration of the device may be significantly simpler and quicker. Finally, it enables close integration with the rest of the execution system, something that proved advantageous with implementation with the integral form time stepping schemes. The potential disadvantage of this is that a much lower iteration rate can be achieved, but as described in section 5.5, the rate is sufficient to accurately control the SDOF system.

As described in detail in the next section, all the required processes have been coded into the same main program, without the use of sub programs other than those inherently used by the applied macros. This enables the whole block diagram to be displayed at the same time, with the exceptions of some options in the case selector structures. All the processes can further be controlled through a single front panel, which represents a medium for entering all test parameters as well as displaying a wide rage of intra-test results.

The utilisation of the fully software based implementation system for pseudodynamic testing is considered by the author as the most cost effective way of combining the experimental and computational components of such tests. It also enables fast and almost continuous execution of tests with highly limited resources through the close and elegant integration between the various processes in the implementation system.

6.2.2 Main execution program

The overall data flow and the detailed operation of the individual components in the implementation system will be explained in the following sections. The switches and displays in the front panel will be related to their terminals and functions in the block diagram, and the interaction between the two accounted.

The following sections will first describe the initial data input and selection procedures before the *main loop* is initiated. The main loop contains all the calculations that are carried out during the running of the tests, and include the time stepping scheme, the controller and the data logging. Thereafter, the controller itself, running as a sub loop, is documented in detail.

Shown in figure 6.11 and 6.12 are typical front panel and block diagrams of the implementation system. These give an indication as to what the overall system looks like, but are not exhaustive. Further front panels are used for the controller and the block diagrams only display one selected option at the time. Additionally, the scale does not allow smaller details to be read. Therefore, in the following sections, relevant portions of the figures have been included on a larger scale to improve readability.

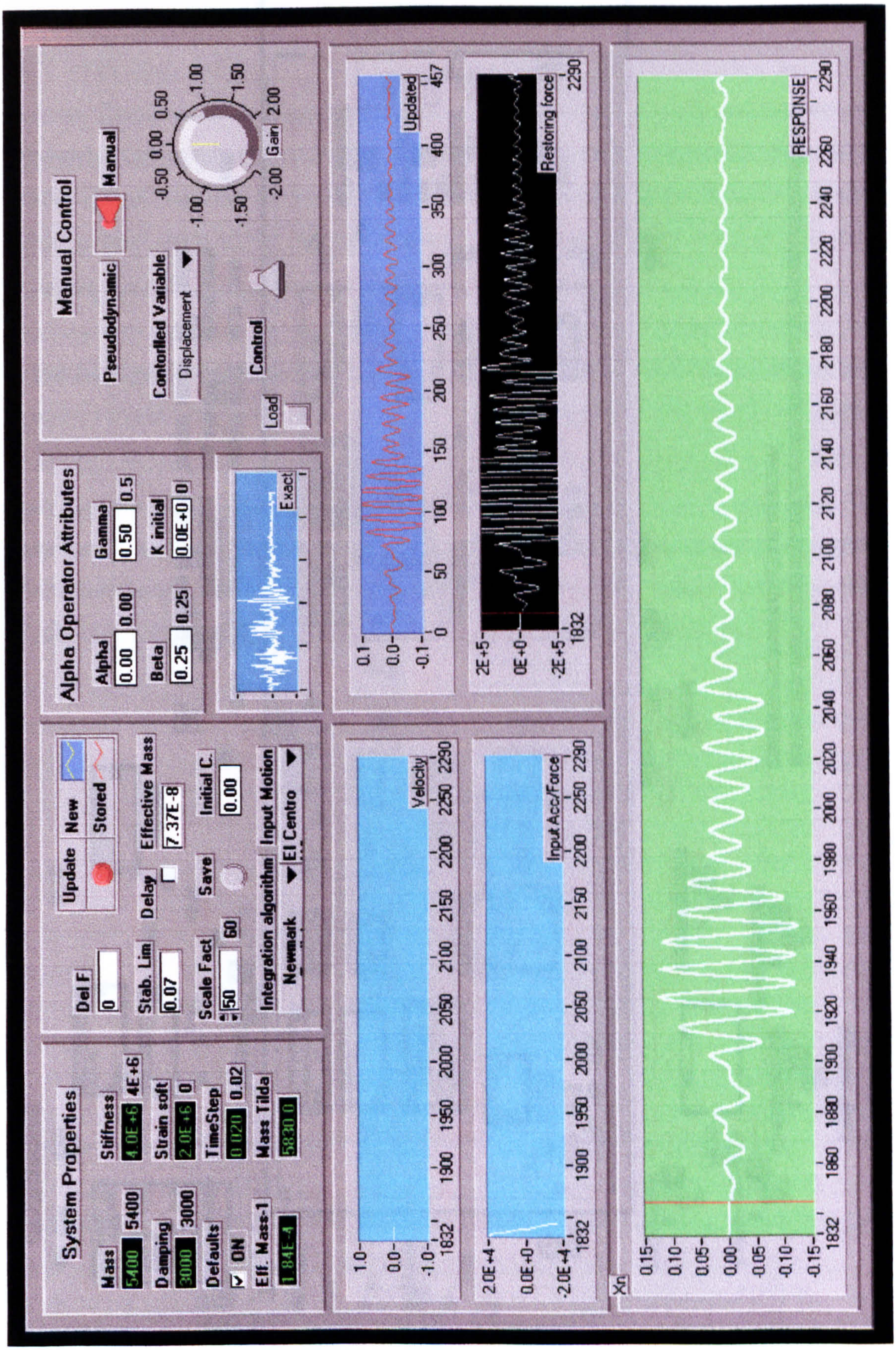


Fig. 6.11 The front panel of the LabView execution system for pseudodynamic testing

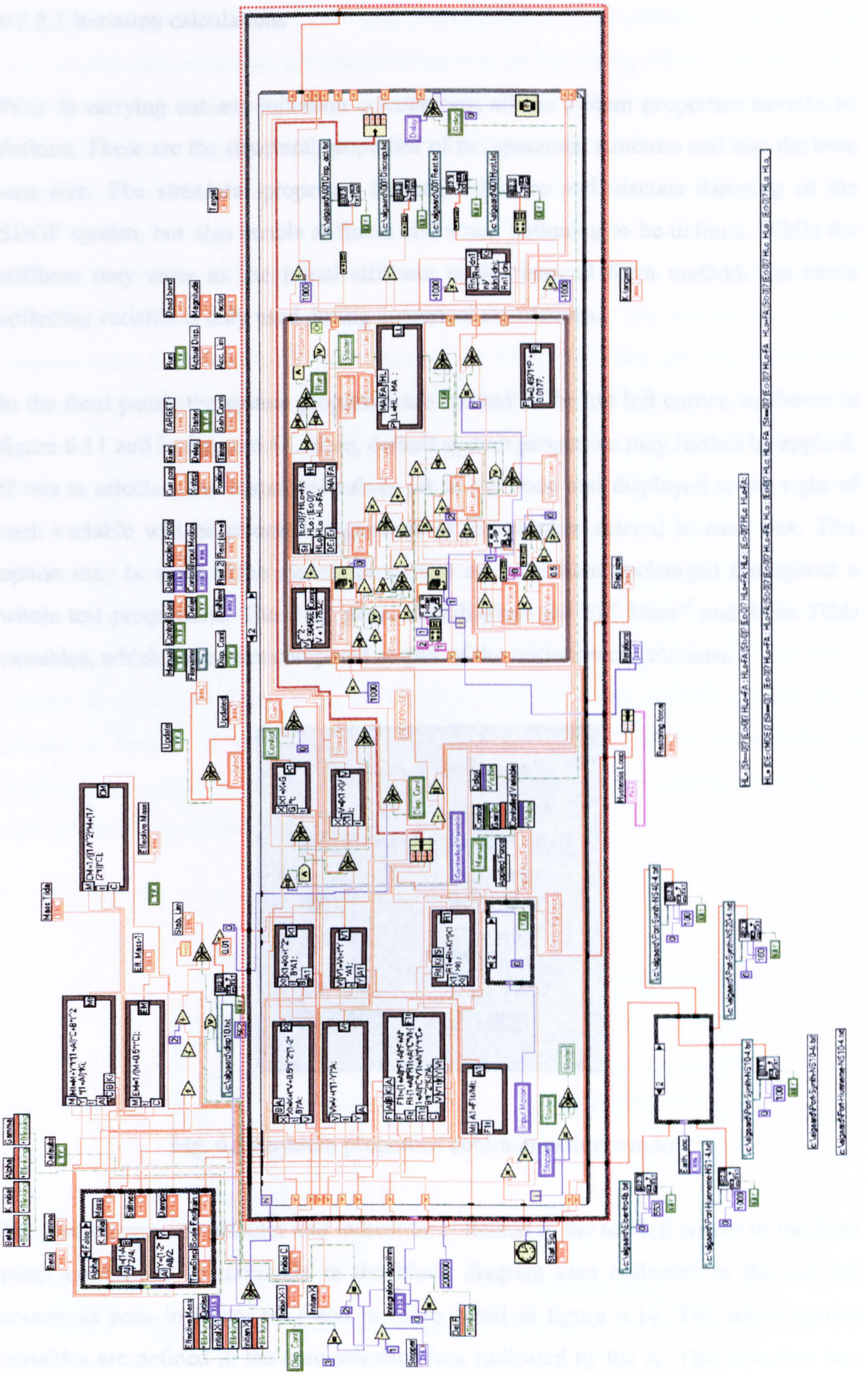


Fig. 6.12 The implementation system block diagram

6.2.2.1 Initiation calculations

Prior to carrying out any initiation calculations, all the system properties have to be defined. These are the structural properties of the specimen structure and also the time step size. The structural properties include the mass and viscous damping of the SDOF system, but also enable stiffness and strain softening to be defined. While the stiffness may enter as the initial stiffness in the integral form method, the strain softening variable is only used during numerical simulations.

In the front panel, the system properties are entered in the top left corner, as shown in figure 6.11 and in figure 6.13. Here, default system properties may further be applied. If this is selected, the numerical values set in the code and displayed to the right of each variable will be returned, regardless of the number entered in each box. This option may be used if the system properties are to remain unchanged throughout a whole test programme. The front panel also displays the *Eff. Mass*$^{-1}$ and *Mass Tilda* variables, which have been computed as part of the initiation calculations.

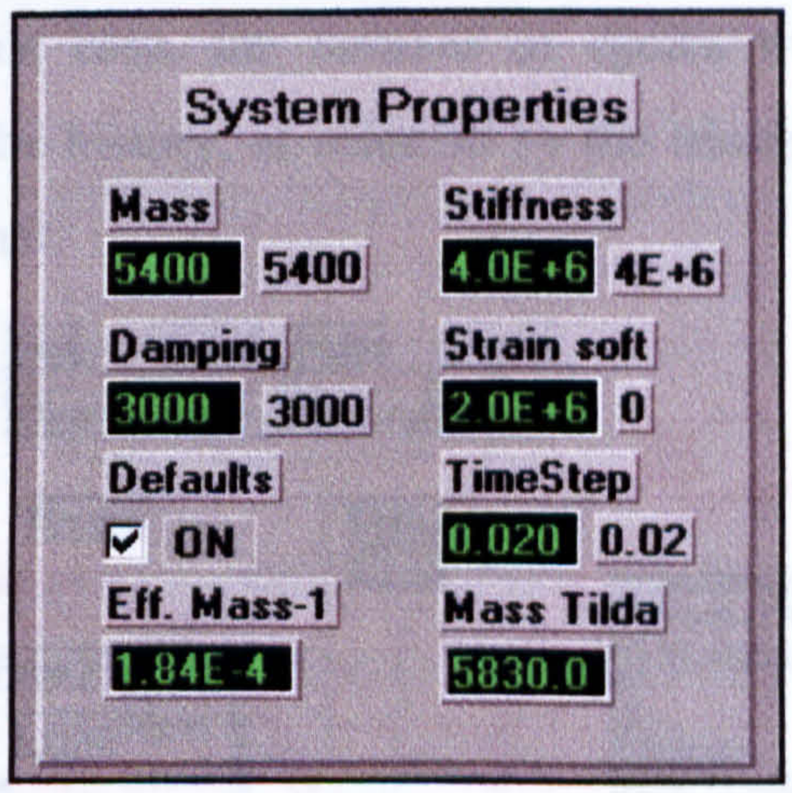


Fig. 6.13 System properties communication window

The system property selector and indicators collected in the top left corner of the front panel correspond to terminals in the block diagram also collected in the top left corner, as seen in figure 6.12 and in more detail in figure 6.14. The actual system variables are defined in the case selection box indicated by the **A**. This selection box switches the defaults on and off. If the default option is ticked, as in figure 6.13, an

alternative variable definition will take place under the *true* option in the selection box. In this case, constants are simply connected to the wires, rather than the *Mass, Stiffness* etc. terminals. The calculations that take place within the formula nodes are for β and γ when using the α-Operator Splitting method (Combescure & Pegon 1997), where the β and γ depends on the α quantity as discussed in section 2.5.2.4.

The computations of the *Eff. Mass*$^{-1}$ and *Mass Tilda* variables, two definitions of effective mass used by various time integration schemes, are carried out in the formula nodes indicated by the **B** in figure 6.14. These variables are transmitted back to the system property box and displayed for information. All the applicable variables created are wired into the main loop at the bottom of figure 6.14. Other functions that take place in this part of the code are calculation of the critical time step size (bottom right), some visual settings (extreme top and left) and the loading up of a displacement history file. The displacement file is used during cyclic testing and is active only if the *Disp. cont* option, seen in figure 6.15, is switched on. The blue *Cycles* terminal determines the number of cycles the main loop runs for and has a control icon shown in figure 6.15. This number is however not used if a displacement history is loaded. In this case, the number of cycles will equal the number of displacement values in the history, as selected by the triangular selection icon below right of the *Cycles* terminal.

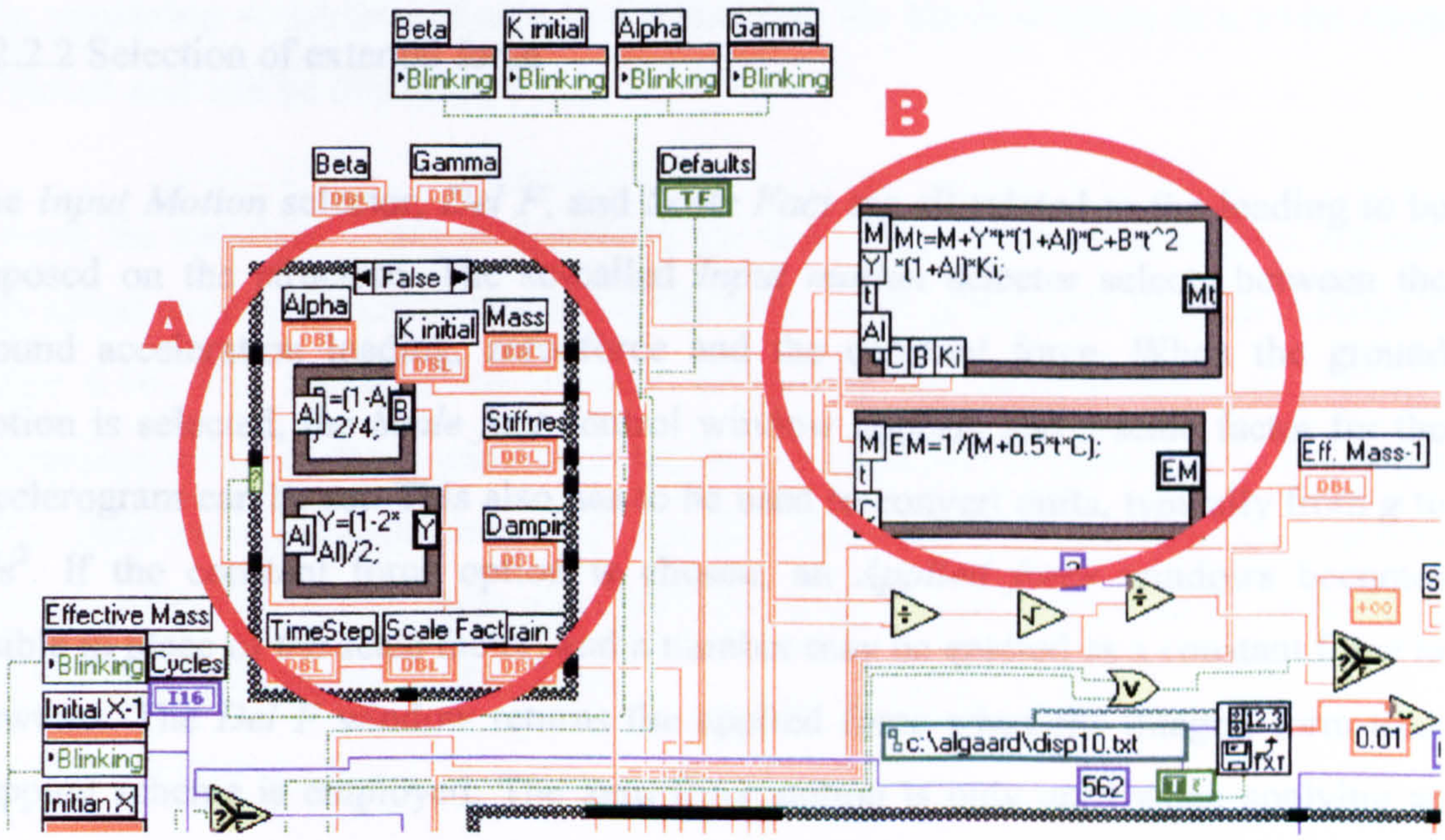


Fig. 6.14 Block diagram showing some of the initiation calculations

Following the definition of the system properties, the loading details, the time stepping scheme and some other variables are set. Considering figure 6.15 below, the *Stab. Lim* displays the critical time step size discussed above, while the large STOP button enables cessation of the process. The *Update* box with the *New* and *Stored* variables enables direct comparison in terms of displacement histories between the current and a stored graph. If the button is pressed, the current response will be saved as the *Stored* array. The two graphs are displayed under the *Updated* graph to the middle right in figure 6.11.

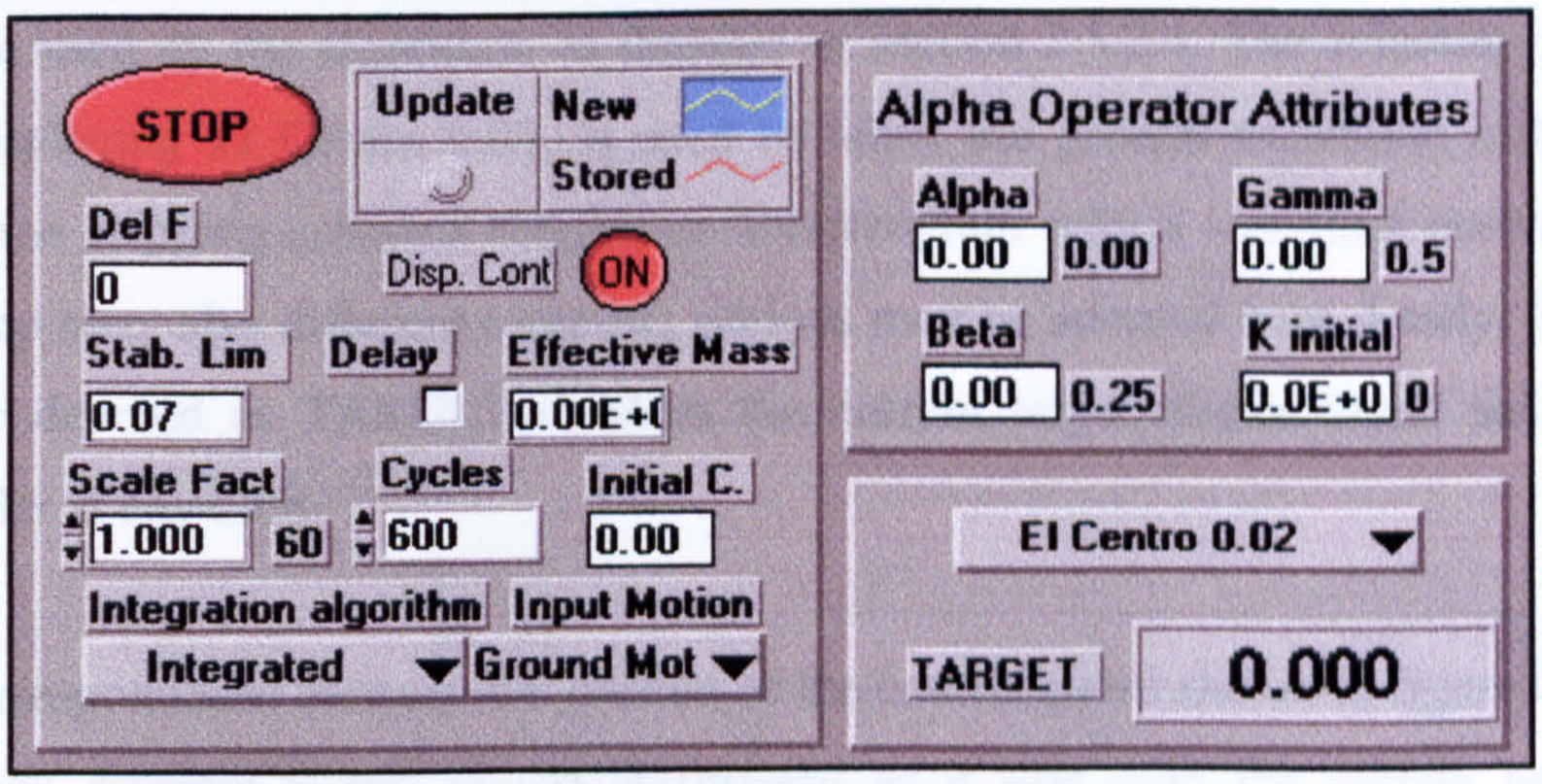


Fig. 6.15 Selector for loading, time stepping scheme and other functions.

6.2.2.2 Selection of external force

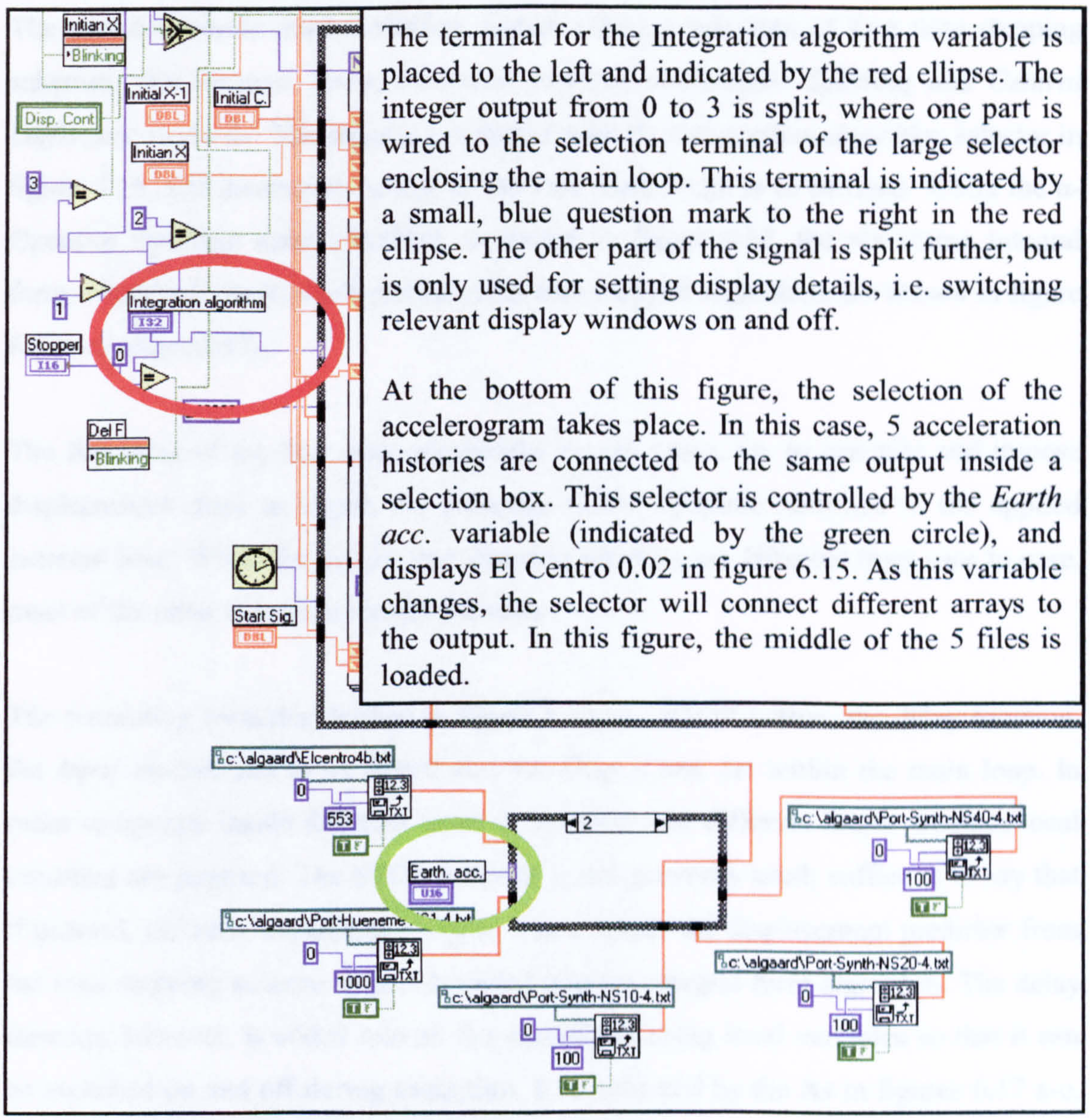
The *Input Motion* selector, *Del F*, and *Scale Fact* are all related to the loading to be imposed on the structure. The so-called *Input motion* selector selects between the ground acceleration loading, zero force and the constant force. When the ground motion is selected, the *Scale fact* control window appears and a scale factor for the accelerogram can be set. This also has to be used to convert units, typically from *g* to m/s^2. If the constant force option is chosen, an *Applied force* windows becomes visible in place of the scale factor, and a number may be entered as a constant force in Newtons. The *Del F* window returns the applied force when the integral form time stepping scheme is employed. The zero force option is only used when applying an initial displacement instead of a force. A delay function may further be employed. When ticked, a selection window appears within which a delay (in milliseconds) may

be added to each time step to slow tests down. Lastly, the *Initial C.* variable controls the initial values for displacement, velocity, acceleration and some other variables, and is generally set to zero.

The two windows to the right of the main selection window in figure 6.15 contain further settings for the α-Operator Splitting time stepping scheme and earthquake excitations. With the α-Operator Splitting scheme, the α parameter is defined (here, between 0 and +⅓) and the β and the γ are displayed, as computed in the formula nodes encircled by **A** in figure 6.14. The *K Initial* variable simply represents the initial stiffness used by the algorithm, as detailed in section 2.5.2.4. The selector, in figure 6.15 displaying El Centro 0.02, is used to select the ground excitation history. As most time stepping schemes require an accelerogram with a sampling period of the time step size, also different sampling periods may be selected here. Lastly, the large number denoted as TARGET displays the current target displacement sent to the controller at any time.

The correspondence between the portion of the control panel shown in figure 6.15 and the block diagram is not as straightforward as it was with the system properties portion. Some of the variables have already been mentioned, and will not be discussed further. These include the Stab. Lim, Cycles, Effective mass and the α-Op. variables. The remaining variables and selectors connect to the block diagram in a wider range of places and will be discussed here.

Among the variables acting externally to the main loop, is the accelerogram selector. The actual terminal of this is situated to the bottom left of the main loop, as detailed in figure 6.16. The accelerogram selector, employing the variable *Earth acc*, simply determines which of the 5 wires entering the selection box is connected to the output wire. Each case connects one wire, ignoring the rest. Also the *Updated* variable operates externally to the main algorithm, but only on completion of the main loop, and will not be discussed further here.



The terminal for the Integration algorithm variable is placed to the left and indicated by the red ellipse. The integer output from 0 to 3 is split, where one part is wired to the selection terminal of the large selector enclosing the main loop. This terminal is indicated by a small, blue question mark to the right in the red ellipse. The other part of the signal is split further, but is only used for setting display details, i.e. switching relevant display windows on and off.

At the bottom of this figure, the selection of the accelerogram takes place. In this case, 5 acceleration histories are connected to the same output inside a selection box. This selector is controlled by the *Earth acc.* variable (indicated by the green circle), and displays El Centro 0.02 in figure 6.15. As this variable changes, the selector will connect different arrays to the output. In this figure, the middle of the 5 files is loaded.

Fig. 6.16 The Integration algorithm and accelerogram selectors

6.2.2.3 Selection of the main algorithm

The setting of the Integration algorithm variable is fundamental. It actually selects the main algorithm, enclosed by the dotted red rectangle in figure 6.12, and encompasses the main implementation loop described later in this chapter. It not only selects which time stepping scheme to employed, but also the controller, as this is coded as part of the same algorithm.

The pseudodynamic implementation system offers a selection of four time stepping schemes: the *Integral Form*, *Newmark explicit*, *α-Operator Splitting* and *Central Difference* methods. The scheme is selected with the *Integration algorithm* selector in figure 6.15, and determines which of the four main loops is to be used. While the α-Operator Splitting main algorithm is shown in figure 6.12, the remaining integral form, Newmark explicit and central difference method algorithms are shown in figure 6.17a-c, respectively.

The functions of the four main algorithms are the same, i.e. to generate and impose displacement steps to obtain the response of the dynamic structure to the applied external load. While the actual time stepping schemes are different from case to case, most of the other functions remain the same.

The remaining variables defined in figure 6.15, the STOP button, the delay function, the *Input motion* and to an extent also the *Disp. Cont*, act within the main loop. In order to operate inside different cases of selectors and different loops, wireless local variables are required. The STOP function is not generally used; suffice is to say that if pressed, the zero displacement signal will override the displacement predictor from the time stepping scheme. It is only coded into the integral form algorithm. The delay function however, is coded into all the algorithms using local variables so that it can be switched on and off during execution. It is indicated by the **A**s in figures 6.17 a-c. The input motion is again part of all the four alternative main algorithms, and indicated by the **B**s in the figures. The Disp. Cont determines whether to carry out a cyclic or pseudodynamic test, and, as the STOP button, it overrides the displacement target set by the time stepping scheme. This is shown by the **C**s in the figures below. The same variable also determines the number of cycles to be carried out in the main algorithm during cyclic test, as discussed in section 6.2.2.1.

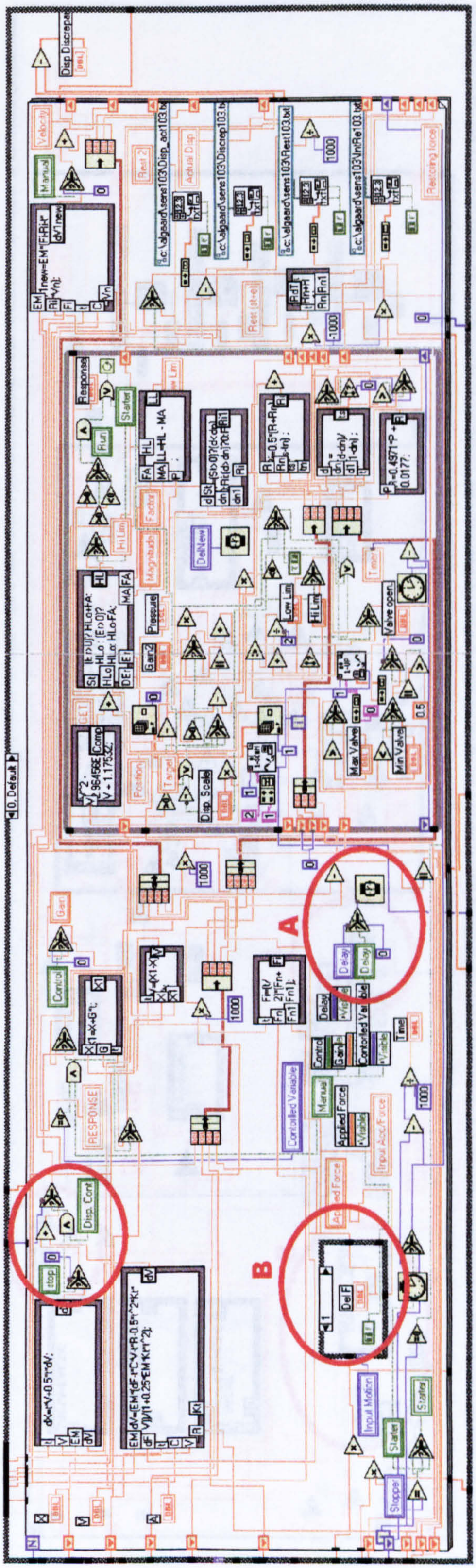


Fig. 6.17a The integral form main algorithm

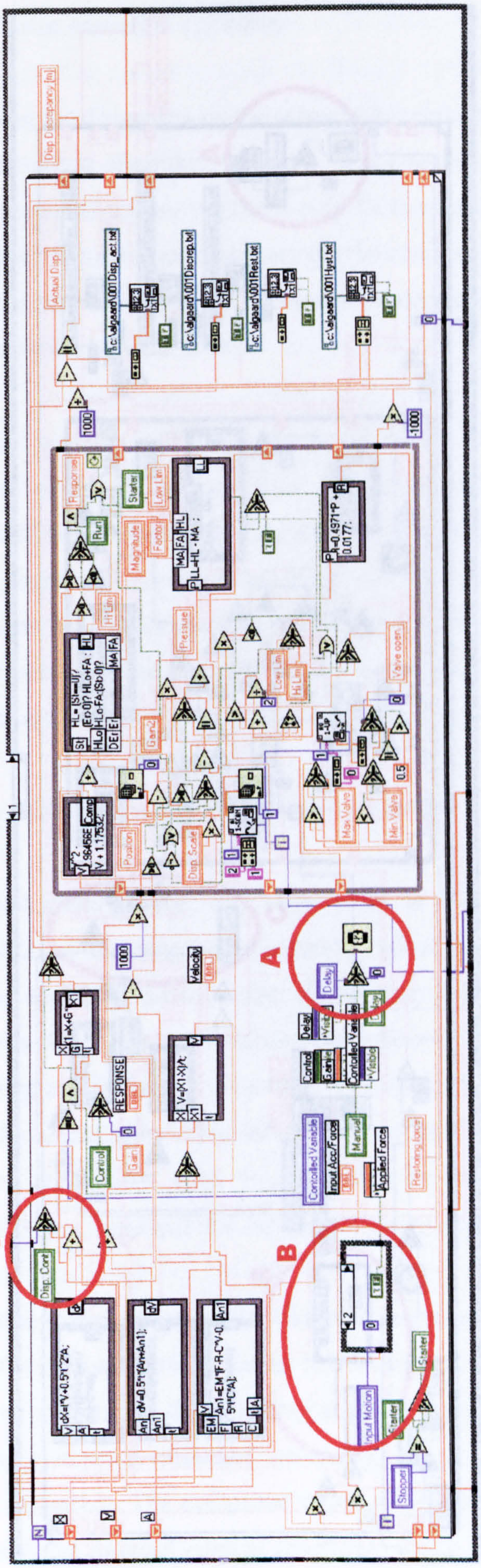


Fig. 6.17b The Newmark explicit main algorithm

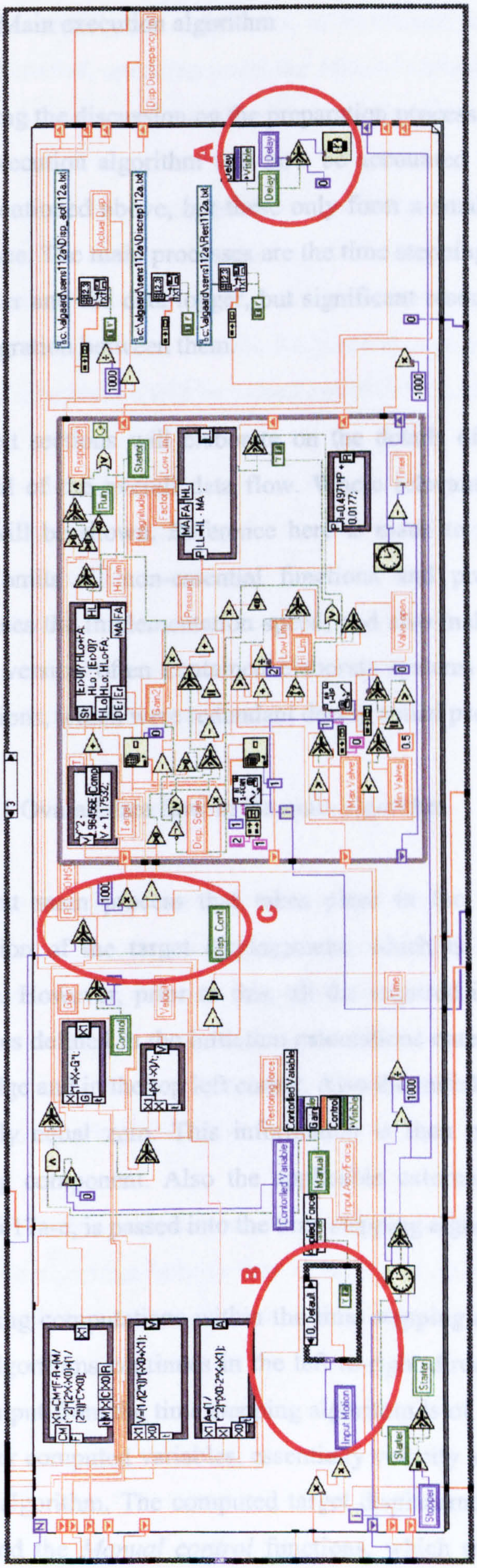


Fig. 6.17c The central difference main algorithm

6.2.2.4 Main execution algorithm

Following the discussion on the preparation process and the initiation calculations, the main execution algorithm will now be accounted for. Some functions have already been mentioned above, but these only form a small part of the many processes that take place. The main processes are the time stepping schemes, the manual control, the controller and the data logger, but significant resources are also required to organise the integration between them.

The next sections will elaborate on the details of the main processes, following a portrayal of the overall data flow. Where relevant, the correspondence to the front panel will be shown. Reference here is made to the final implementation version, which omits all non-essential functions and processes. This stripped algorithm maximises the implementation speed, and also makes the data flow easier to follow. Earlier version often contained elaborate systems used primarily during numerical simulations, which were redundant during actual pseudodynamic implementation.

6.2.2.4.1 Overall data flow in the main algorithm

The first main process that takes place in the main execution algorithm is the generation of the target displacement, which is carried out by the time stepping scheme. However, prior to this, all the required information needs to be available. Variables defined in the initiation calculations enter the main algorithm along the left hand edge and in the top left corner. Also the initial conditions enter here, all of which generally equal zero. This information is then passed to the right, into the time stepping component. Also the applicable external force component, denoted **B** in figure 6.17a-c, is passed into the time stepping algorithm.

Following computations within the time stepping algorithm, the data flow inside the main algorithms continues in the left-to-right direction as shown in figure 6.18. The vital output from the time stepping algorithm is of course the target displacement, but the other computed variables, essentially velocity and acceleration, are also passed on by the algorithm. The computed target displacement is then challenged by the *Disp. Cont* and the *Manual control* functions, which may override the signal. The *Disp.*

Cont is used only for cyclic tests, as mentioned in the previous sections, while the *Manual control*, operating under the *Manual* variable as described in section 6.2.2.4.3, switches between manual control and pseudodynamic or cyclic tests.

Once the final target displacement has been set, this signal enters the control loop. The controller additionally requires definition of a whole array of variables, which will be discussed in section 6.2.2.4.4. The loop will then iterate until the measured displacement equals or exceeds the prescribed target displacement. On completion, a range of information will be passed out of the controller, again on the right hand side.

Once the controller has terminated its action, further calculations must be carried out, particularly when employing the integral form method. At this stage, the data logging tasks may also be carried out. Their objective is principally to save the required information from that time step to an external file. Once this is done, all that remains is to pass any information used by subsequent time steps into the shift registers on the right hand edge of the main loop.

6.2.2.4.2 Time stepping schemes

The time stepping scheme generates the target displacement based on the current displacement, velocity, acceleration and forces acting. The detailed operation of the four time stepping schemes employed here is described in section 2.5 and chapter IV. In all cases they work by satisfying equilibrium in the equation of motion, and are integrated into the pseudodynamic testing framework by using an experimentally measured restoring force term.

The time stepping schemes are essentially coded into the left hand quarter of the main execution algorithm box, as they constitute the first main task to be carried out. The following four sections will detail the coding of each scheme in turn.

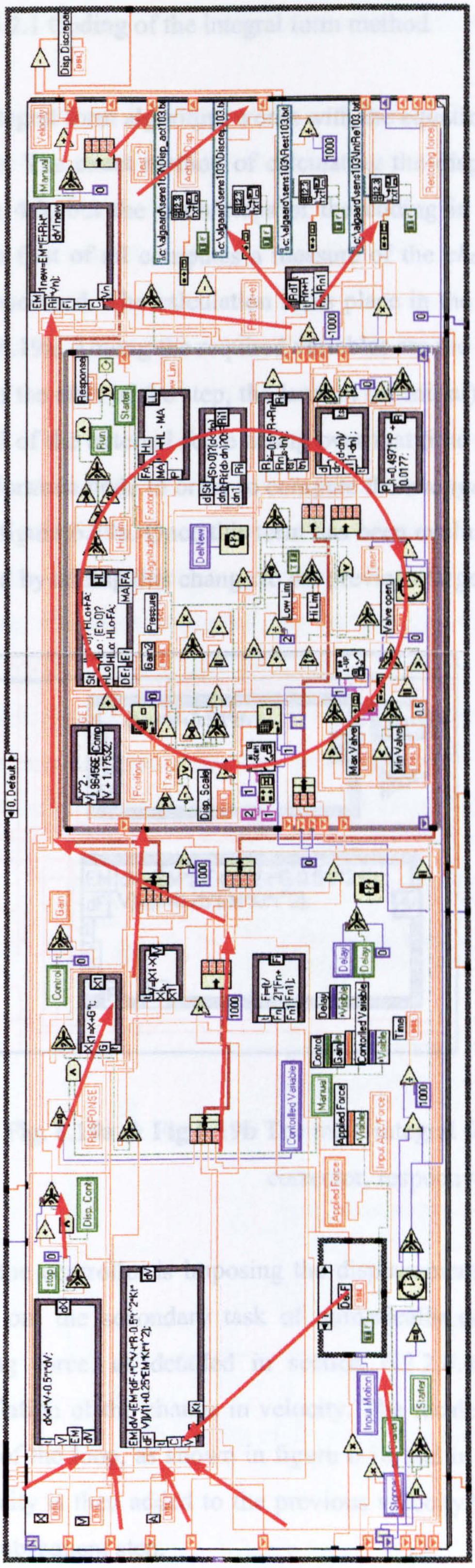


Fig. 6.18 Overall data flow in main algorithm

6.2.2.4.2.1 Coding of the integral form method

The integral form algorithm works with the equation of motion integrated with respect to time. The exact method of calculating the displacement predictor is described in section 4.3, but the main steps of the coding in LabView are presented here. The scheme first of all computes a measure of the *change in velocity*, dV, over the time step concerned. The calculation takes place in the lower of the two formula nodes in figure 6.19a. Among the required variables are the effective mass matrix, the restoring force at the start of the step, the tangent (or initial) stiffness, the velocity and the time integral of the external force acting over that time step. The dV term is then wired up to the formula node in order to compute the *change in displacement*, dX, shown at the top of figure 6.19a. Once this term has been evaluated, the actual target displacement is found by adding this change to the previous target.

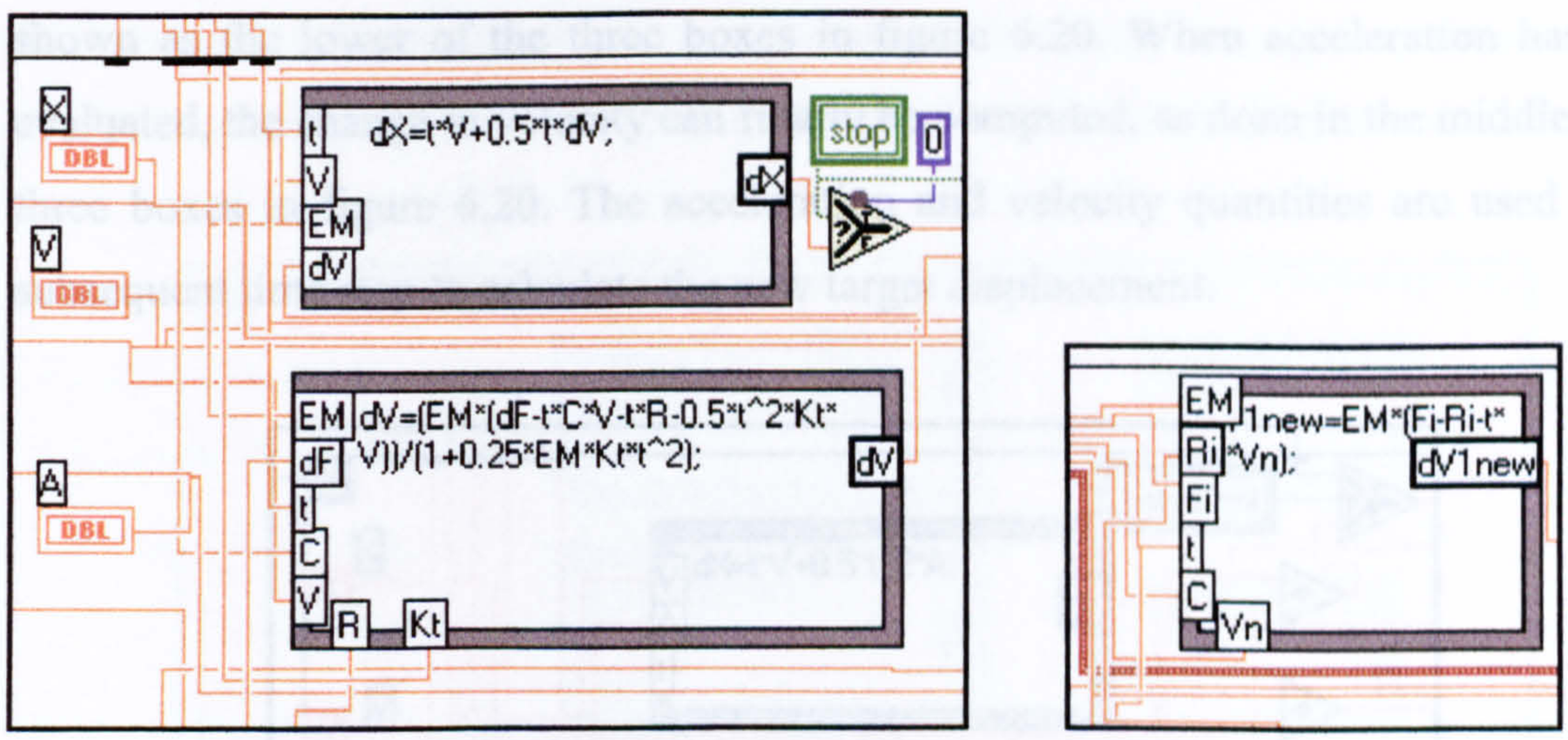


Fig. 6.19a & Fig 6.19b The main integral form calculations; predictor and corrector, respectively.

While the controller is imposing the displacement step on the specimen structure, it carries out the secondary task of numerically computing the time integral of the restoring force, as detailed in section 6.2.2.4.4.4. This term is required in the recalculation of the change in velocity. The recalculation takes place in the top right corner of the loop, as shown in figure 6.18 and in detail in figure 6.19b. This change in velocity is then added to the previous velocity to yield the corrected velocity used in the subsequent step.

6.2.2.4.2.2 Coding of the Newmark explicit method

Implementation of the Newmark explicit method is somewhat simpler than the integral form method. This is mainly due to the fact that the method is genuinely explicit and does not require any corrector step. The theory behind the method is described in more detail in section 2.5.1.2.

The Newmark explicit method proceeds by calculating a change in displacement based only on terms obtained in the previous time step, as seen in the top formula node in figure 6.20. The displacement target is then obtained by adding this change to the previous target, and the new target is then sent to the controller. Unlike with the integral form method, the Newmark Explicit, and all the other methods, require only the restoring force measured at the target displacement. Once this variable is available, the value is passed into the formula node for calculating the acceleration, shown as the lower of the three boxes in figure 6.20. When acceleration has been evaluated, the change in velocity can finally be computed, as done in the middle of the three boxes in figure 6.20. The acceleration and velocity quantities are used in the subsequent time step to calculate the new target displacement.

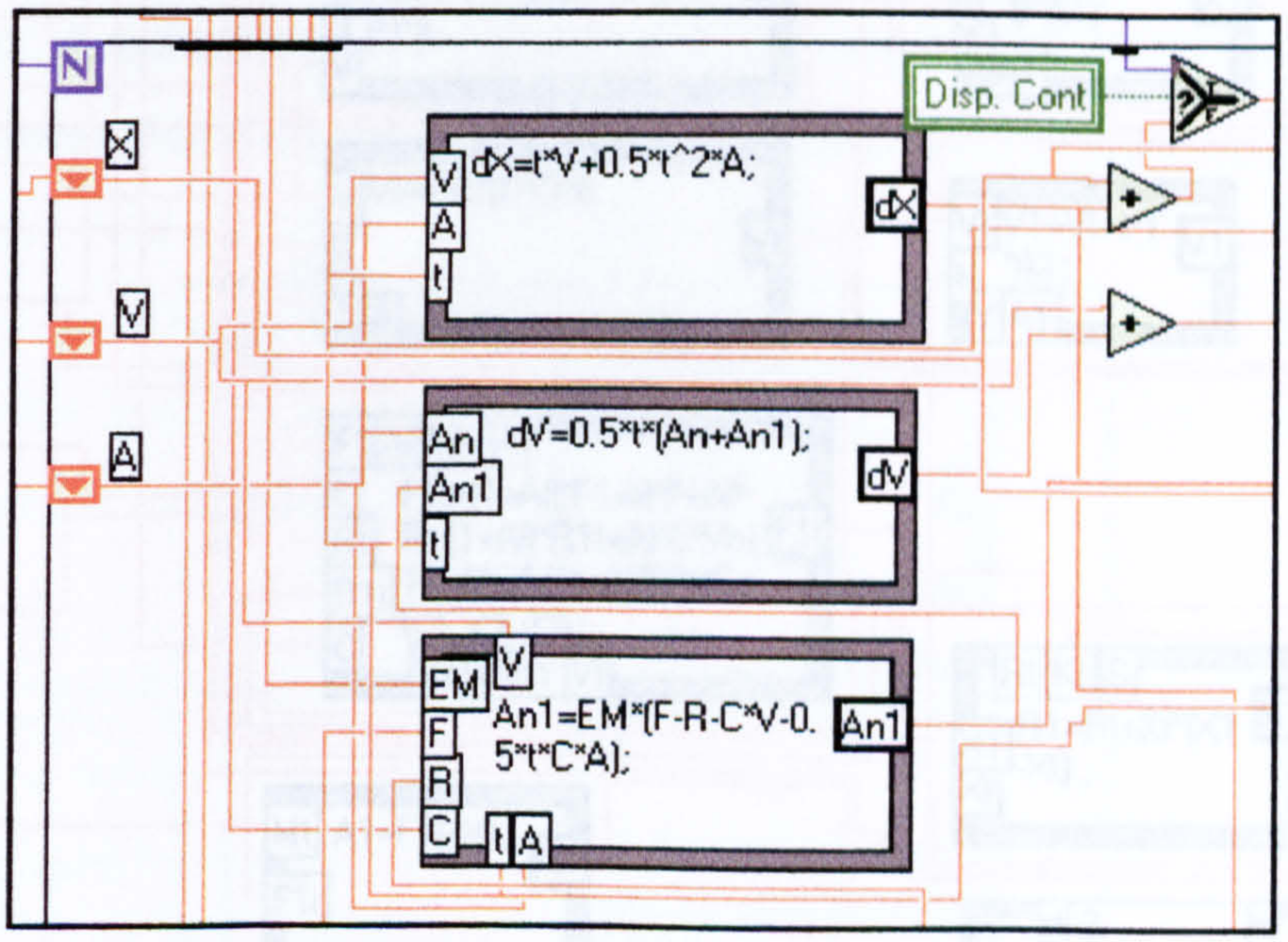


Fig. 6.20 Coding of the main Newmark explicit calculations

6.2.2.4.2.3 Coding of the α-Operator Splitting method

The α-Operator Splitting method is a predictor-corrector method (suggested by Combescure & Pegon 1997) and detailed in section 2.5.2.4. Explicit expressions for the displacement and velocity, Xt and Vt, respectively, are computed in the two top left formula nodes in figure 6.21. The displacement expression forms the target displacement sent to the controller. Once imposed, the resulting restoring force is passed back to the time stepping scheme. This term then enters into the complex expression for the so-called pseudo-force vector, F1t (Combescure & Pegon 1997), computed in the second from bottom left formula node in figure 6.21. The implicit acceleration, A1, is then solved for using this pseudo-force expression and the effective mass matrix, in the bottom left formula node. When the implicit measure of the acceleration is finally found, the implicit expressions for displacement and velocity, X1 and V1, respectively, can also be evaluated, as shown in the two top right formula nodes. Additionally, it is required to calculate an approximation of the implicit restoring force, R1, and this is done in the bottom right formula node.

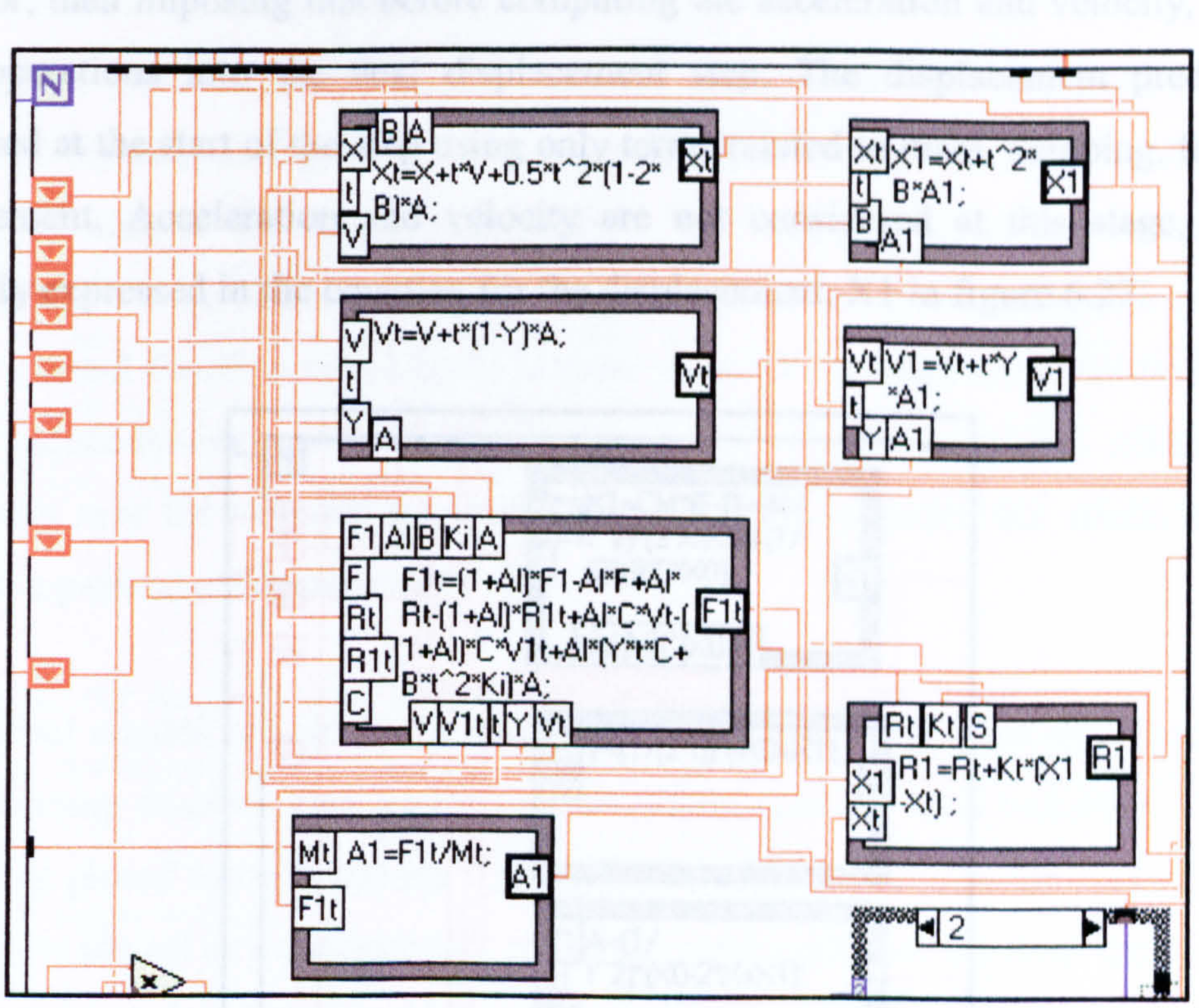


Fig. 6.21 Coding of the main α-Op. Split. method calculations

The α-Operator Splitting method is highly unusual in the sense that only part of the final displacement step is imposed. The implicitly calculated corrector part is used only in the subsequent time step where it forms the base for the next displacement step. Likewise, the restoring force corresponding to the corrected displacement is never obtained experimentally; an extrapolated approximation is used.

6.2.2.4.2.4 Coding of the central difference method

The central difference method is mathematically very similar to the Newmark explicit method (Shing & Mahin 1986). It is explicit, so the displacement predictor can be computed with information available at the start of the time step. However, as detailed in Shing & Mahin (1986) and section 2.5.1.1, the method requires data from the two previous time steps and thus needs a start up procedure.

The method is implemented rather differently from the Newmark explicit method although the end result is equivalent. Instead of calculating the displacement predictor, then imposing this before computing the acceleration and velocity, it shifts some operations into the next displacement step. The displacement predictor is calculated at the start of the step using only terms related to mass, damping, force and displacement. Acceleration and velocity are not considered at this stage, but are indirectly expressed in the equation for the displacement, X1 in figure 6.22.

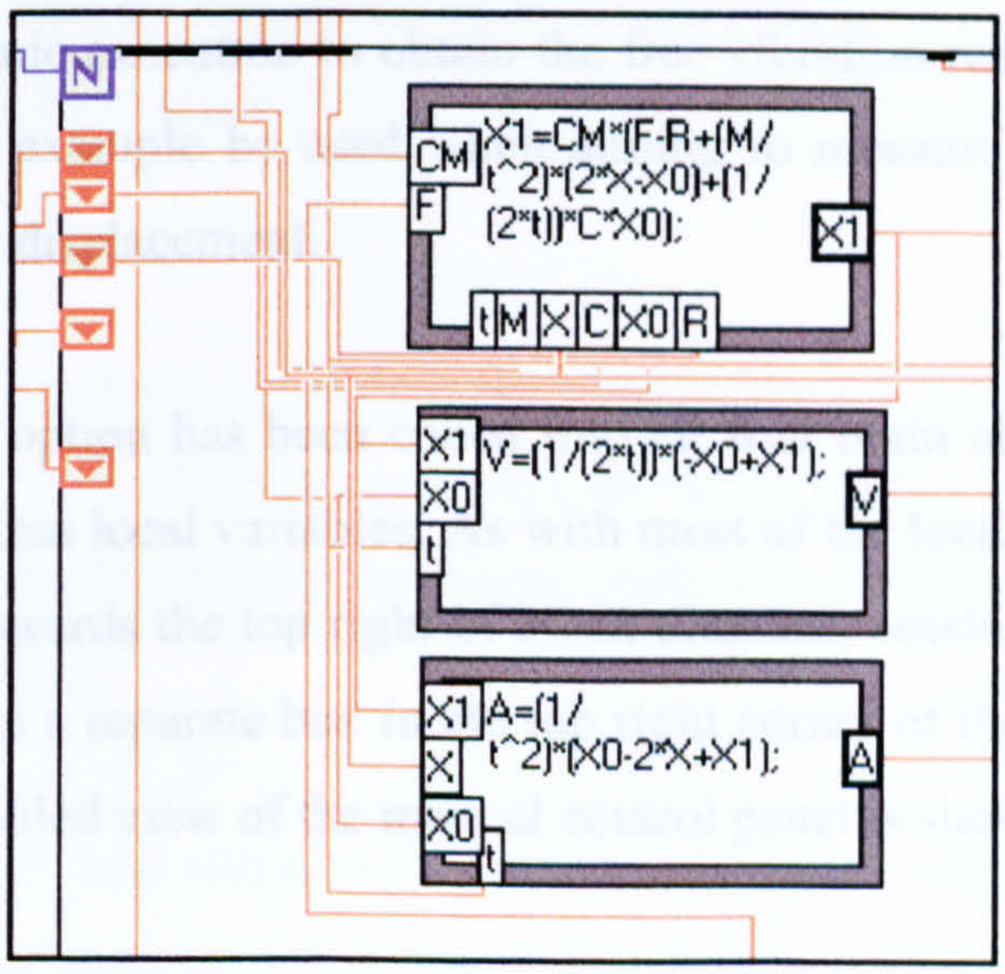


Fig. 6.22 Coding of the central difference method

Following the calculation of the predictor displacement, the expressions for velocity and acceleration, V and A, respectively, can be evaluated, before imposing the step. Once imposed, the measured restoring force is passed into the next time step, where it is used in the displacement calculation. Interestingly, the velocity and the acceleration terms are not required for the computation of subsequent displacement steps. In fact, they are not required for any calculations and are used only for information purposes.

As the central difference method uses displacement data from the two previous time steps, particular use has been made of the shift register, the device that carries data from one cycle into the next. By having a double output terminal, seen as the  icon in figure 6.22, input entered in the two previous iterations becomes available. No particular start up procedure has been employed. All the variables have been given the same initial condition of zero. This has not been found to affect the overall response significantly.

6.2.2.4.3 Manual control

The flexible pseudodynamic implementation system has been designed to enable quasistatic (cyclic) testing as well as pseudodynamic testing. Additionally, the option of manual control has been incorporated. The set-up allows for instantaneous switching between manual control and pseudodynamic execution. Typical use of the manual control function could be to impose a certain displacement and then switch over to pseudodynamic execution to obtain the free vibration response. Alternatively, the system may for example be used when aiming to measure the elastic restoring force for a particular displacement.

The manual control option has been coded into all four main algorithms and is thus based on using wireless local variables. As with most of the local variables, the actual terminal is placed towards the top right of block diagram, outside the main loop. The controls are placed in a separate box in the top right corner of the front panel, as seen in figure 6.11. A detailed view of the manual control panel is shown in figure 6.23.

Switching between pseudodynamic and manual control is done with the red switch at the top of figure 6.23, in this case indicating manual. This switch selects between the pseudodynamically and manually created displacement targets. Furthermore, the manual control panel gives the option of selecting the controlled variable, where displacement is the default but velocity and acceleration are also available. However, only the displacement option has been activated in the code, so the remaining two options are not applicable.

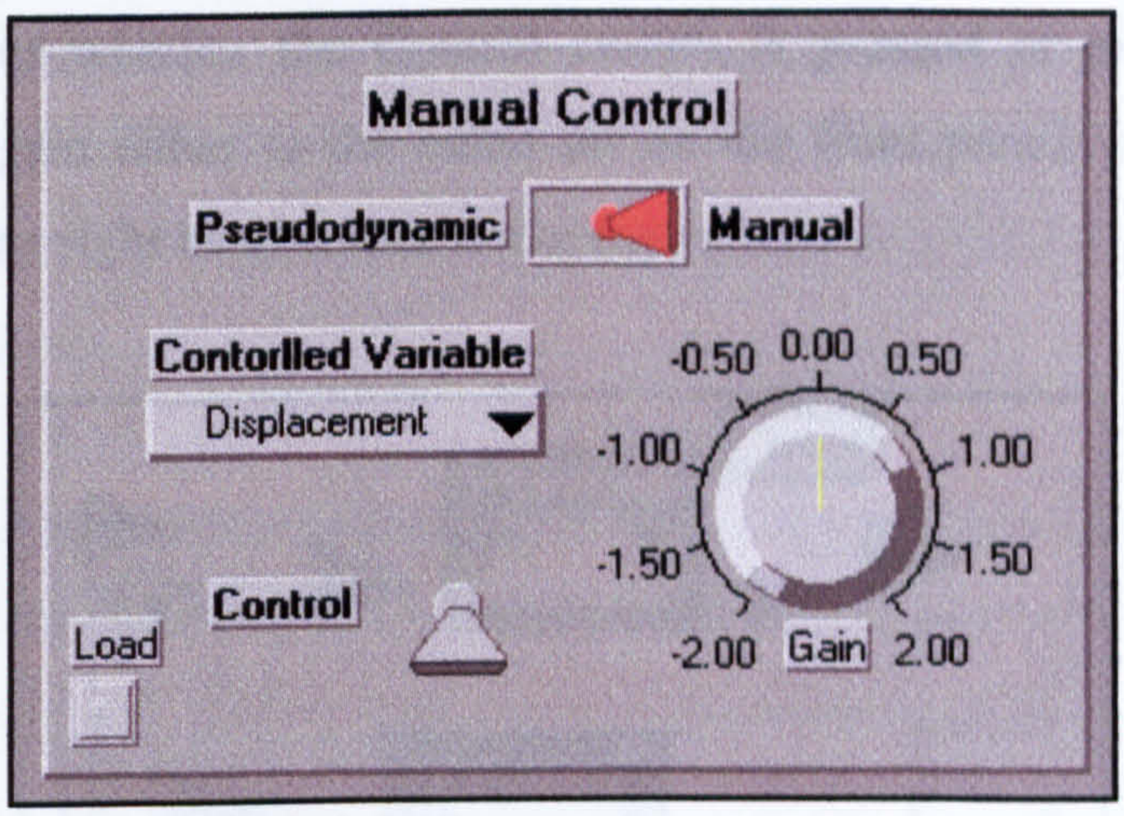


Fig. 6.23 Manual control panel

Assuming the displacement is selected as the controlled variable, the *gain* has to be set. The gain essentially controls the rate with which the controlled variable is changed, and in this case represents the millimetre change per time step. The gain can be set on the dial, or by using a digital control option if an exact numerical figure is required.

Changes to the displacement are only carried out when the switch denoted *Control*, here grey, is pressed using the mouse pointer on the PC. It has been set such that the switch must be continually pressed to induce motion to reduce the risk of unwanted damage to the specimen.

When considering the interaction with the block diagram, only the code from one main loop will be displayed, as the set up is very similar in all four cases. The variable that chooses between the pseudodynamic and manual input is called *Manual*, and is

located towards the bottom left of figure 6.24. This boolean variable is connected to a selector function through a logic *AND* switch along with a boolean signal related to whether the controlled variable is set to displacement, all shown in the top left corner of the figure. If both variables are true, the pseudodynamic signal is overridden.

An alternative signal to the pseudodynamic one has to be generated, as this is done in the formula node at the top of the figure. It defines the new displacement target as the previous target plus the gain multiplied by the time step size. This calculation takes place regardless of whether the Control switch is pressed or not. However, this variable sets the gain either to the value set on the front panel, or to zero. This is illustrated in the top right hand corner in the figure below.

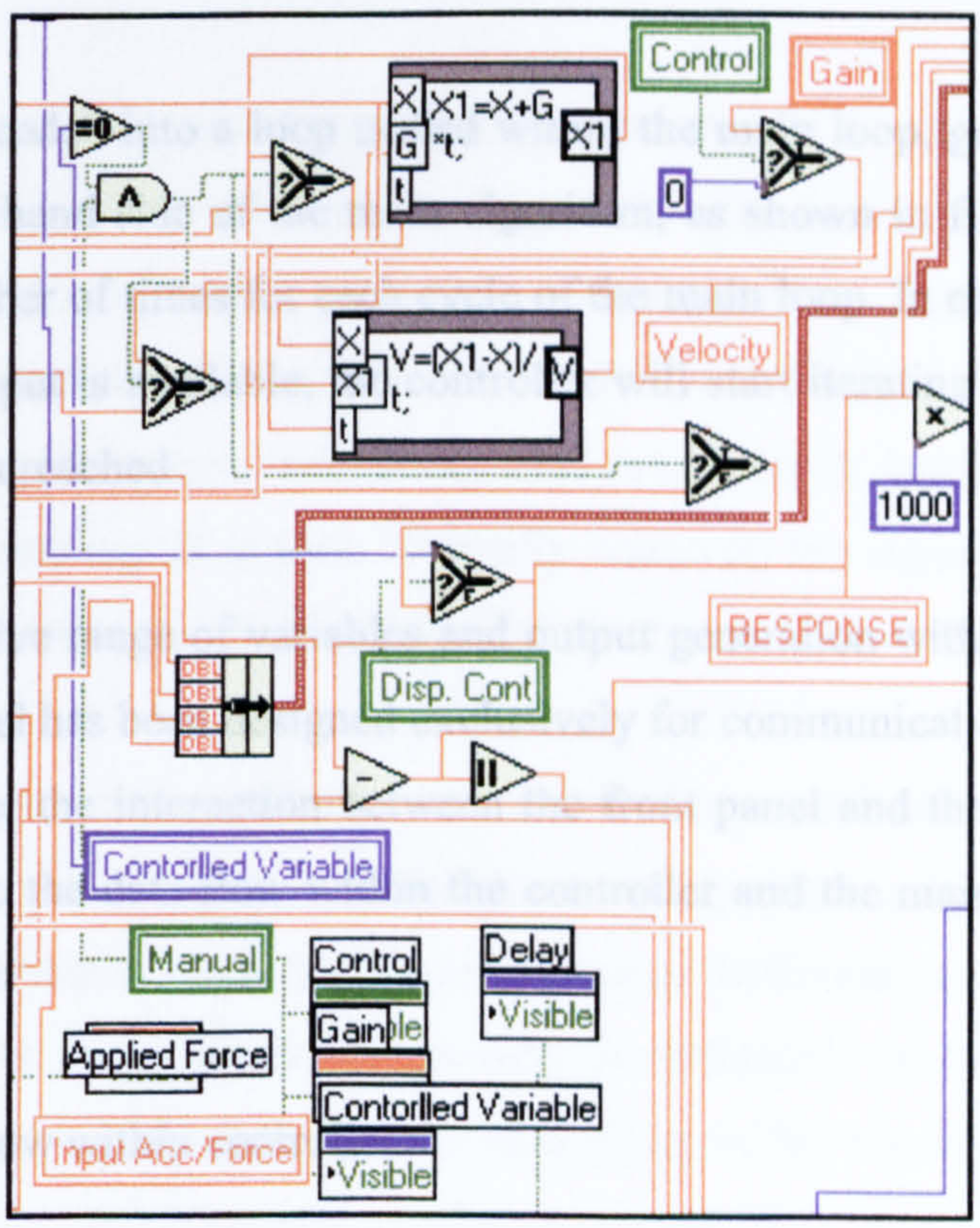


Fig. 6.24 Coding of the manual control option

A formula node also exists for calculating the effective velocity during the time step. This is simply done by dividing the displacement change by the duration of the time step, and is used both for information and when switching over to pseudodynamic execution.

A system also exists for setting a range of visual parameters on the front panel. For example, all the options for manual control remain hidden until the execution is switched from pseudodynamic to manual. This is done using so-called *attribute nodes*, which are collected in the centre bottom part of the figure.

6.2.2.4.4 Controller

The controller represents a piece of code that communicates with the hardware and ensures that the target displacement generated by the time stepping scheme is actually imposed by the experimental apparatus. The way it operates is described in detail in chapter V; the following sections will discuss how the theory has been coded into the LabView environment.

The controller is coded into a loop nested within the main loop, generally positioned towards the right hand side of the main algorithm, as shown in figure 6.12. Here, it will iterate a number of times for each cycle of the main loop. In each time step, once all the required input is available, the controller will start iterating and continue until the target has been reached.

Due to the extensive range of variables and output generation within the controller, a separate front panel has been designed exclusively for communication with this. In the following sections, the interaction between the front panel and the control code will be discussed. Also the data flow within the controller and the main functions will be explained in detail.

6.2.2.4.4.1 Data flow within controller

The data flow within the controller is not as organised as the data flow within the main algorithm. The flow is not only from left to right or top to bottom, however the input to the loop is still on the left edge and the output on the right hand side.

The target displacement passed into the loop at the start forms the basis for a lot of the operations within the controller, and is indicated by the arrow entering the loop in the top left corner in figure 6.25. The next main task to be carried out is to read the

displacement transducer and convert this to a displacement value. The reading is carried out in the box encircled and indicated by **A**, while the conversion and the software correction of the displacement signal is carried out in and around the formula node at the top left corner of the figure.

The finished measured displacement value is then compared with the target displacement to yield a measure of the displacement error. This error is an important variable in the computation of the *dead zone* limits. The dead zone is the voltage range within which a change in the signal does not result in a change in the output from the valve, as discussed in detail in section 5.3.3. The dead zone calculations are subsequently carried out within the two large formula nodes towards the top right corner, as shown in section 6.2.2.4.4.3.

The dead zones form the basis of the signal generation, and the exact basis line for the signal is determined in the centre of the loop. When the appropriate signal has been created, this is sent to the bottom left corner of the loop. Here, final checks are carried out to ensure that the signal is suitable for transmission to the valve. If the signal is outwith the allowable range, something that would only happen under unusual circumstances, a zero signal is sent. Normally however, the signal will pass through the checks and be sent to the hardware through the macro indicated by **B**.

In parallel with the signal generation and the dead zone calculations, the controller will investigate the relationship between the current position, the step direction and the error to determine when to cease iterating. Different criteria for defining achievement of the target displacement were investigated, as discussed in section 6.2.2.4.4.5. In any case, the calculations take place in the top right hand corner of figure 6.25.

In addition to the processes mentioned above, possibly the most important task of the controller is to measure the restoring force offered by the specimen structure. This is done during every iteration at the same time as reading the displacement position, indicated by the **A** in the figure above. The signal from the load cell or pressure transducer is converted to a force signal through a simple linear relationship in the formula node in the bottom right hand corner.

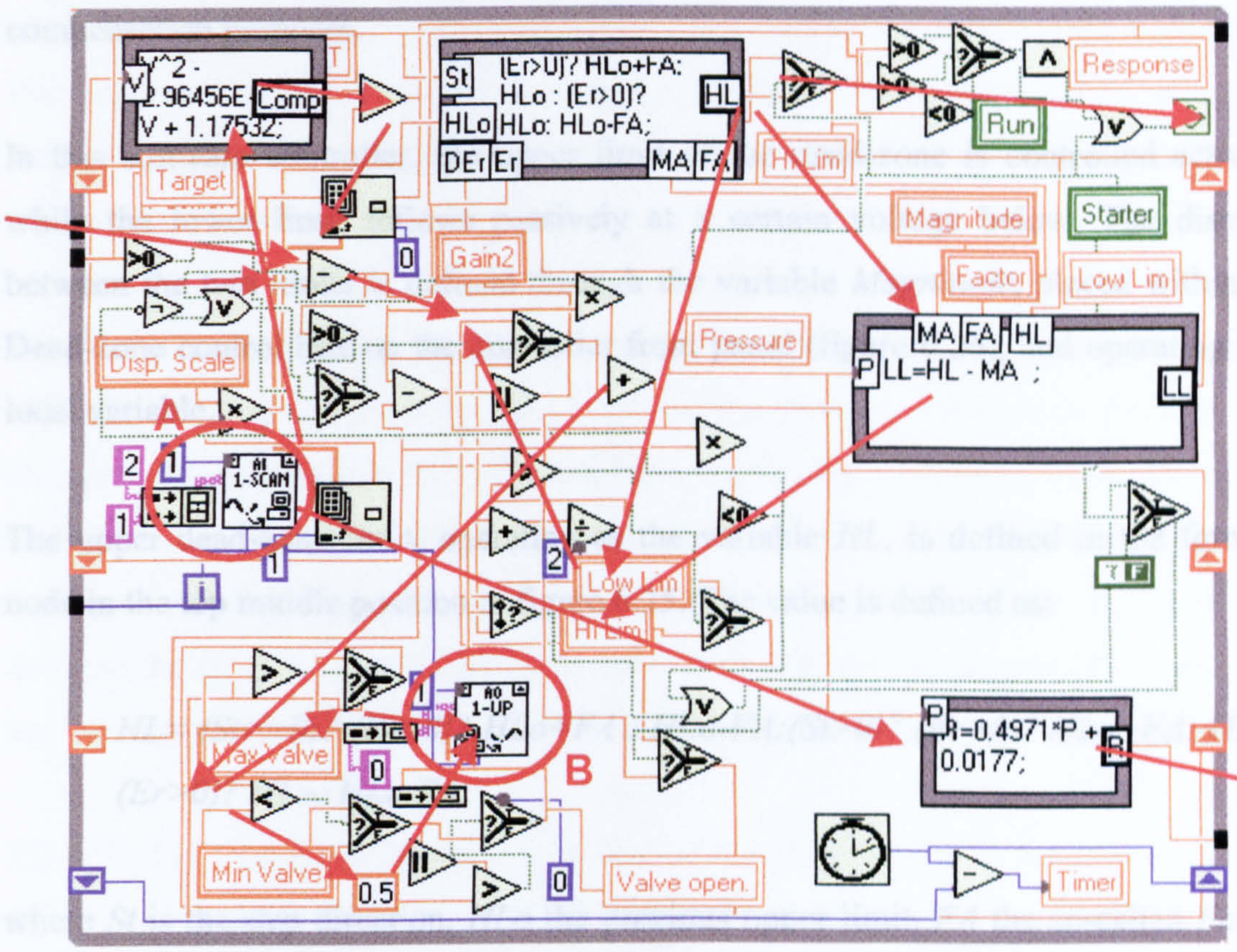


Fig. 6.25 Data flow within the controller

6.2.2.4.4.2 Dead-zone calculations

This section describes the implementation of the required dead-zone operations as detailed in section 5.3.3. Briefly, the existence of the dead-zone is a property of the valve and hydraulic system, and implies that a change in the electrical signal to the valve may not result in a change in the pressure, or at least not in the displacement. As the dead-zone position is not constant, but pressure dependent, any compensatory system will need to take both the position and magnitude into account. Successful compensation of the dead-zone was enabled through combination with the integral control mode. The integral control mode, described in detailed in section 5.3.4, essentially creates a signal change proportional to the integral of the error in the system. Basically, the integral control mode was incorporated into the system by changing the limits of the dead-zone. This ensured both that the final target was

always met and that the dead-zone was effectively shifted to the correct position for compensation purposes.

In this software controller, the upper limit of the dead-zone is controlled actively, while the lower limit follows passively at a certain voltage below. The distance between the two limits is defined through the variable *Magnitude*, placed within the Dead-zone control box on the controller front panel (figure 6.26), and operating as a local variable.

The upper dead-zone limit, operating as the variable *HL*, is defined in the formula node in the top middle position in figure 6.25. The value is defined as:

HL= (St==0)? (Er>0)? HLo+FA : HLo-FA:(St>0)? (Er>0)? HLo+FA: HLo : (Er>0)? HLo: HLo-FA;

where *St* is the step direction, *HLo* the previous upper limit, *FA* the so-called *Factor* and *Er* the displacement error present in that instant. Written in the quasi code, the expression becomes:

```
IF St = 0 THEN
    IF Er > 0 THEN
        HL = HLo+FA
    ELSE
        HL = HLo-FA
ELSE
    IF St > 0 THEN
        IF Er > 0 THEN
            HL = HLo+FA
        ELSE
            HL = HLo
        ENDIF
    ELSE
        IF Er > 0 THEN
            HL = HLo
        ELSE
            HL = HLo-FA
        ENDIF
    ENDIF
ENDIF
```

The so-called *Factor* variable determines the rate of change of the position of the dead-zone. As seen in the quasi code above, the position of the dead zone is shifted by the magnitude of the *FA* variable in each step. The variable is manually set on the front panel in the *Dead-zone* control box and takes the unit of Volts, which is also the unit the dead-zone and the signal generation system operate with.

The first condition in the expression for the dead-zone limit, $St = 0$, is only really applicable during the start up procedure. During tests, a certain step distance will always exist, and the direction of this determines the next selection. Then, the error is considered. Assuming the step is in the positive direction, if the error is still positive, the signal will be increased by the magnitude of *FA*. However, if the target has been met and the error is negative, the dead-zone limit will not be changed. This condition has been set to avoid the possibility of unwanted intra-step load reversal taking place.

The effect of this definition of the position of the dead-zone is to linearly change the signal as long as an error is present. This not only overcomes the problem of the dead-zone, but also enables a form of the integral control mode. With this system, it is always ensured that the target position is met. The effect of changing the *Factor* is to set the speed with which this happens. It should be noted that the signal change from this origin is not determined by the magnitude of error, and will thus not produce a desirable velocity profile. If only this mode is enabled, the velocity of the actuator will be near constant.

The lower limit of the dead-zone follows as mentioned passively behind the upper limit. The lower limit uses the variable *LL*, which is defined in the large formula node to the right in figure 6.25. The signal generator switches between the two limits according to the desired direction of the actuator. Both the upper and lower limits are displayed at any time on the front panel in the Valve settings box under the variable names of *Hi Lim* and *Low Lim*, respectively. The initial limits may also be set here at the start of the test, but will be controlled by the program during execution.

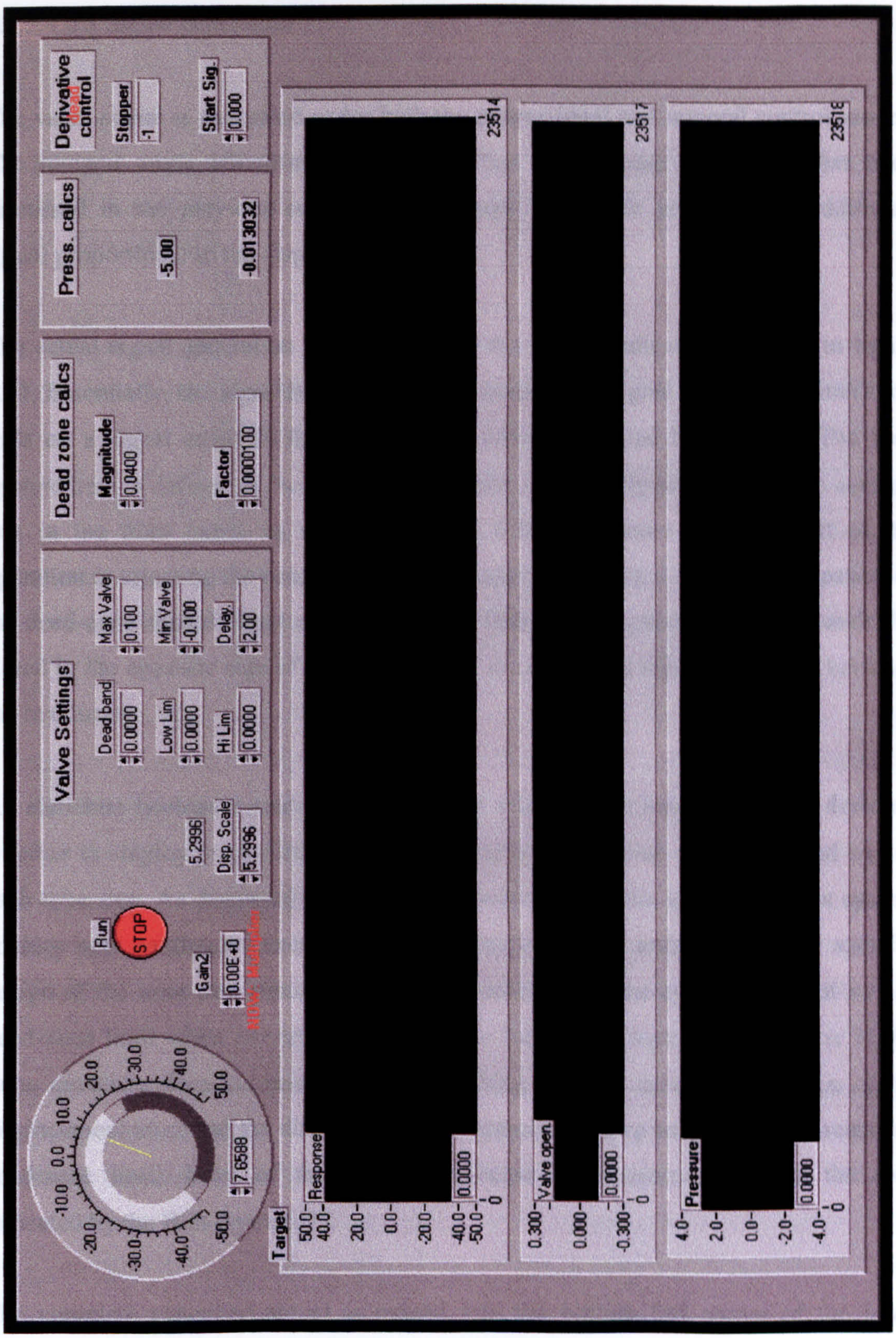


Fig. 6.26 Controller front panel

6.2.2.4.4.3 Signal generation

The valve signal is generated under both the proportional and integral control modes. The integral mode effectively sets the position of the base signal, and has been discussed in the previous section. The proportional mode generates an additional signal proportional to the displacement error.

The actual signal generation is carried out in the region indicated by the **A** in figure 6.27. Essentially, the algorithm picks up the correct base signal (dead-zone limit) and adds on a signal equal to the displacement error multiplied by the gain. The gain applied here is defined as the variable *Gain2* and is manually set in the Valve settings box in the front panel, as shown in figure 6.26. The more complex part of the operation is selecting the base signal. As discussed in section 5.3.3, any occupation of the dead-zone should effectively be avoided. Therefore a system exists that sends the signal to the opposite side of the dead-zone if the computed signal should lie between the two limits.

An elaborate boolean system is also used for selecting the base signal and deciding whether to employ proportional gain during the initial start-up at the start and end of each time step. As discussed in section 5.3.5 and 6.2.3.1, this implementation system enables near continuous execution of the pseudodynamic tests through the specific design of the inter step signal generator. In broad terms, the system does not switch dead-zone limit when overshoot occurs if the following displacement step is in the same direction. This has been done by switching between using the direction of the displacement error and the direction of the displacement step as criterion for selecting dead-zone limit. Most of the selection process is concentrated within the area indicated by the **B** in figure 6.27.

The complete generated signal is passed into the bottom left corner of the loop, indicated by the **C**, where it goes through the last safety check before being sent to the valve. The safety check investigates whether the signal lies within the prescribed limits set on the front panel, denoted *Max Valve* and *Min Valve*. These limits could be set to limit the force exerted by the actuator or to ensure that the signal sent to the valve is not so high that it could damage it. A final check has also been included to

ensure that the signal generated does not exceed 0.5V regardless of what is entered into the front panel.

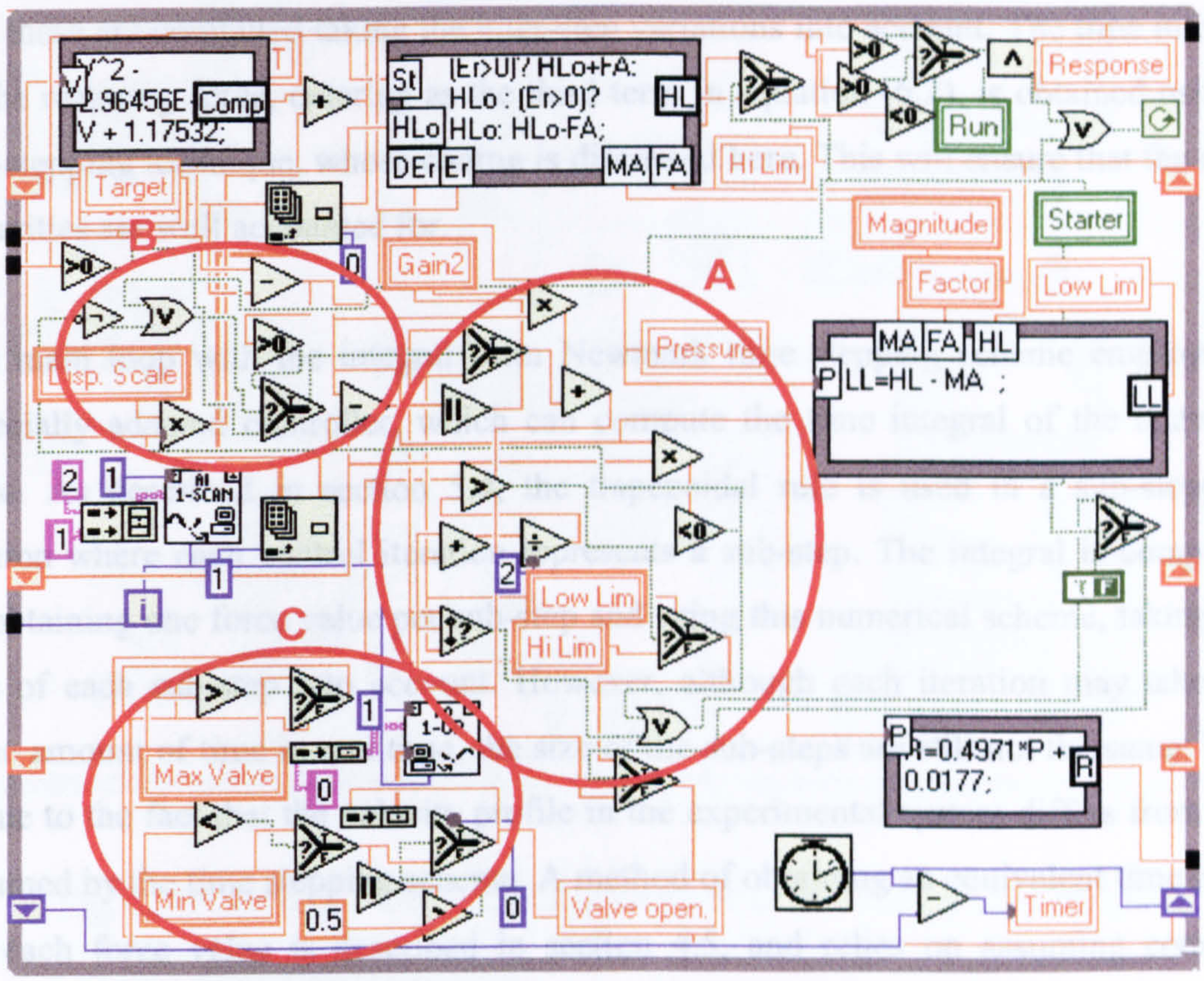


Fig. 6.27 Signal generation in the control loop

6.2.2.4.4.4 Computation of the integral of the restoring force

The controller has a secondary objective when implemented with the integral form time stepping scheme. In addition to imposing the required displacement step and obtaining the restoring force, the controller has to numerically integrate the restoring force over each time step. The time integral of the restoring force is a required term for the integral form algorithms, discussed in detail in chapter IV. The integral form algorithms operate with the equation of motion integrated with respect to time, as shown in equation (6.2).

$$M\Delta\frac{dx}{dt}+C\Delta x+\Delta\int r(x)dt=\Delta\int f dt \tag{6.2}$$

Initially suggested by Chang *et al.* (1998), it is argued that this form displays improved abilities in dealing with rapidly varying loads and stiffness properties. This can be reasoned through the fact that the time integrals of the forces are employed and that these are computed taking the inter-step variations into account. The time integral of the restoring force, entering as the third term in equation (6.2), is obtained using a sub-stepping technique, whose coding is discussed here. This will ensure that the non-linearities are well accounted for.

The main loop with the integral form Newmark time stepping scheme employs an especially adapted controller, which can compute the time integral of the restoring force. As described in section 5.4, the trapezoidal rule is used in a sub-stepping fashion where each control iteration represents a sub-step. The integral is computed by obtaining one force value per sub-step and using this numerical scheme, taking the size of each sub-step into account. However, although each iteration may take the same amount of time in real time, the size of the sub-steps are still not the same. This is due to the fact that the velocity profile in the experimental system differs from that assumed by the time stepping scheme. A method of obtaining an equivalent time point for each force value is described in section 4.5, and relies on assuming constant velocity during each step. The time point of the sub-step is defined such that the fraction of completed time equals the fraction of the completed displacement step. For example, when ½ of the displacement step has been imposed, it is said that ½ of the time has passed as well even though maybe only ¼ of the actual time has passed as the actuator moves faster at the start.

The numerical integrator is coded into the lower right hand side of the controller, as shown in figure 6.28. It utilises some additional formula nodes for computing the equivalent time positions and the time-force area for each iteration. Some problems were present due to noise in the displacement channel giving rise to overly large, small or even negative equivalent sub-step durations. However, the introduction of some additional criteria in the computation of the time points mitigated these problems. The criteria introduced included the requirement that displacements measuring short of the starting position of the time step resulted in an equivalent time of zero, while any displacements measured beyond the target resulted in an equivalent time point equalling the time step size.

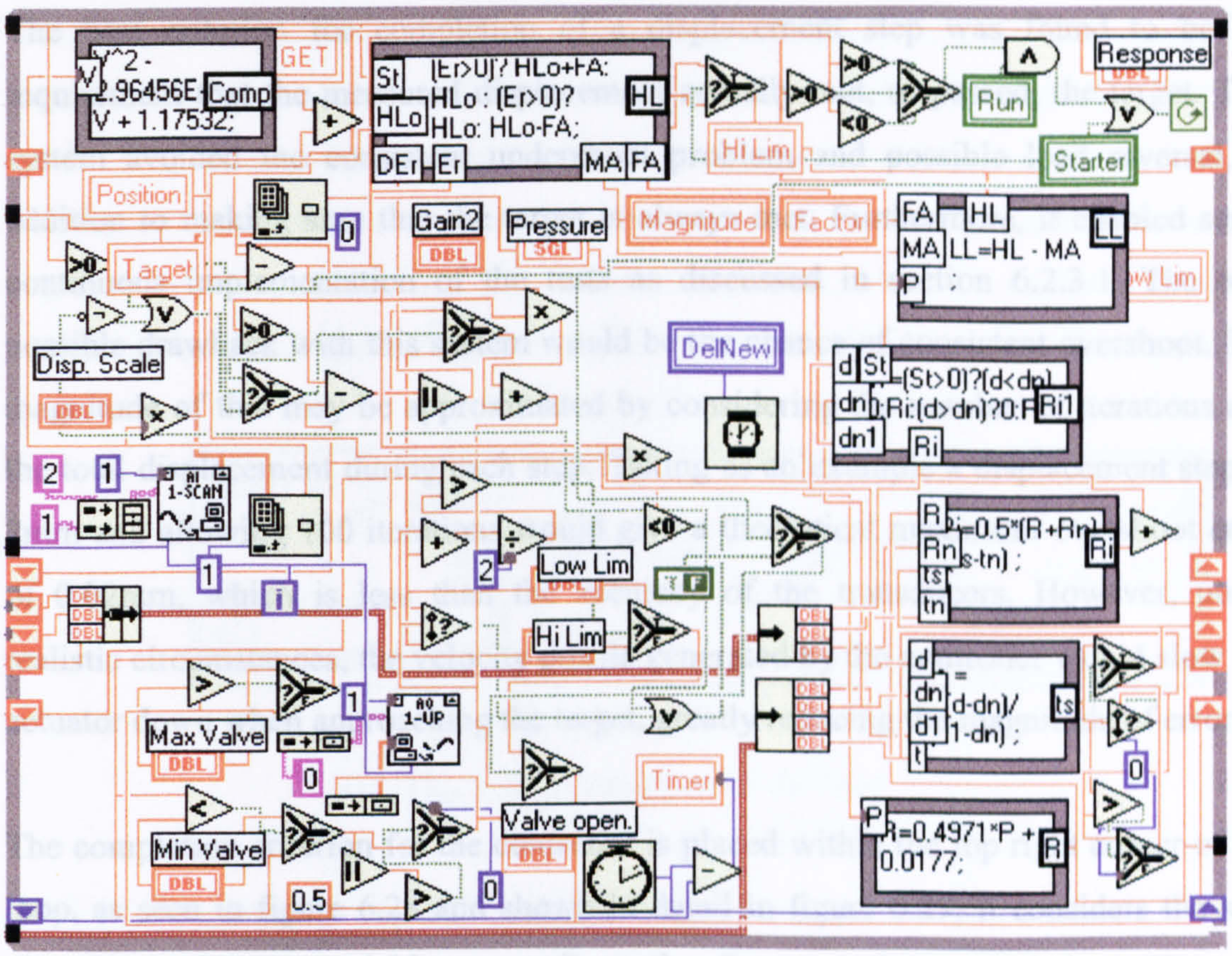


Fig. 6.28 Controller with additional function of calculating time integral of the restoring force

6.2.2.4.4.5 Coding for completion of the loop

A range of different completion criteria were considered during the development of the controller. These included allowing a certain number of iterations per step, reaching the target displacement within a given tolerance or simply reaching or passing the target displacement. The common system of setting a tolerance was not found to be the best solution. This method resulted either in consistent undershoot, as the controller would cease as soon as it reached the target less the tolerance, or load reversal, as the controller would attempt to correct overshot displacements. Allowing a certain number of iterations per step did in principle work. However, a generous number of iterations had to be allowed to ensure completion of the displacement step under all conditions, resulting in discontinuous, classic implementation, as discussed in section 5.3.5.

The best criterion for completion of a displacement step was found to be the requirement that the measured displacement actually met, or passed, the target. This system avoided the consistent undershoot problem and possible load reversal, in addition to making sure that the target is always met. Furthermore, it enabled semi-continuous implementation of the tests as discussed in section 6.2.3.1. The only possible drawback with this system would be the chance of consistent overshoot. The magnitude of this may be approximated by considering the number of iterations and the total displacement during each step. Taking as an example a displacement step of 2mm and allowing 100 iterations would give a theoretical maximum overshoot error of 0.02mm, which is less than the accuracy of the transducers. However, under realistic circumstances, the velocity profile generated by the controller would slow the actuator down when approaching the target, greatly reducing the magnitude of error.

The completion criterion for the controller is placed within the top right corner of the loop, as seen in figure 6.28 and shown in detail in figure 6.29. It considers the step direction and error variables as well as the *Run* and *Starter* switches. The step direction variable is first queried to determine which direction the step is in. If >0, the result is a boolean *true*, if not, it is *false*. The *true/false* signal is then sent right, into a selector box. At the same time, the error signal is considered. First of all it is split, then the two parts are evaluated to check the direction of the error, in one case whether it is positive and in the other whether it is negative. One of the two signals is then selected, depending on the step direction, to yield a final *true/false* signal. As an example, consider a positive step direction and a negative error. The step direction will result in a *true* signal, which will select the upper of the two error checks, i.e. whether it is >0. As it is negative, the final signal will be *false*. If on the other hand the step direction is negative, the *false* statement from this will select the lower of the two checks on the error, i.e. whether it is negative. As it is, the final signal will be *true*.

In any case, once the final signal from the step and error calculations has been computed, this goes through some operations with the Run and Starter variables. The Run variables should in principle always be *true*. The function is only for emergency use, but when *false* will not allow the loop to iterate (unless the starter function is *true*). So when the signal from the step and error are combined with the Run signal in

the *OR* operation, the resulting signal should always equal the step/error signal. This signal is finally combined with the Starter signal in an *AND* operation. The starter is used during the start-up procedure to let the controller run and hold a position without moving on to new time steps. So when this is defined as *true*, the final signal will in any case be *true* and the loop will continue. However, when the starter is *false*, the step/error signal will govern and the loop will iterate until the target has been passed.

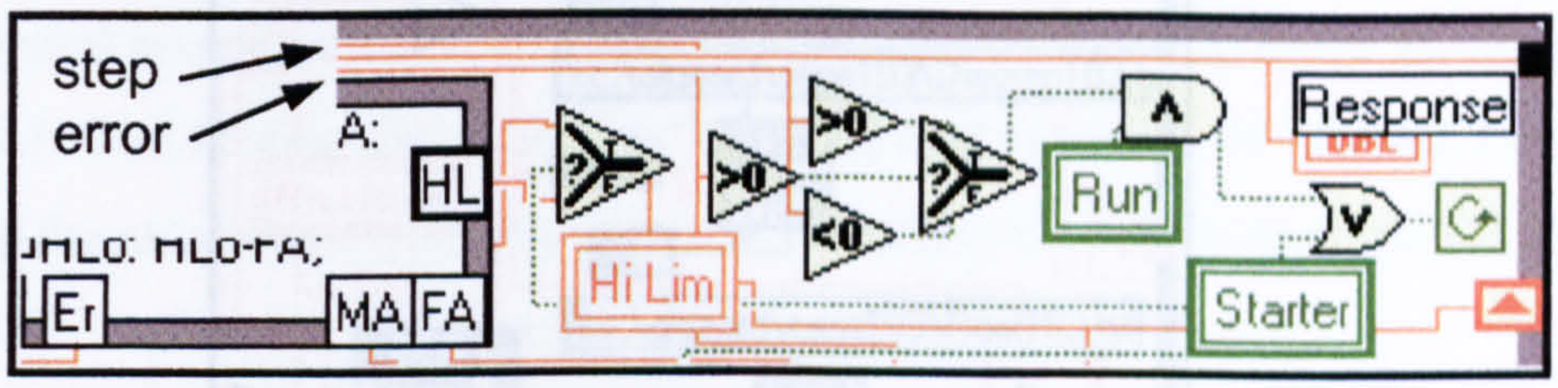


Fig. 6.29 Completion of the loop

6.2.2.4.5 Data logging, concluding calculations and completion of the loop

Data logging is generally carried out at the end of each time step. This ensures that the relevant information from each step is saved as soon as it is obtained and cannot be lost. Although the process is somewhat computationally expensive, the alternative of carrying out the data logging at the completion of the main loop has not been elected as data would then be lost if, for some reason, the test should not be completed.

As the general data flow within the main loop is from left to right, the data logging process is located on the far right hand side within all the algorithms. The process can be seen with the file communication macros (shown in figures 6.4a-d) to the far right in figure 6.18 and in detail in figure 6.30 below.

In most of the main algorithms, four of the five variables are logged: the actual displacement reached at cessation of the control loop, the discrepancy between the reached and targeted displacements, the restoring force at cessation, the time integral of the restoring force over time step and the force-displacement values for that time point. Of course, the latter of these can be built up *a posteriori*, but is still practical to create during the test.

In most cases, the desired variable can be wired directly into the file communication macro through a required build array icon as discussed in section 6.1.4.1. In other cases, simple numerical operations are carried out to create the correct signal.

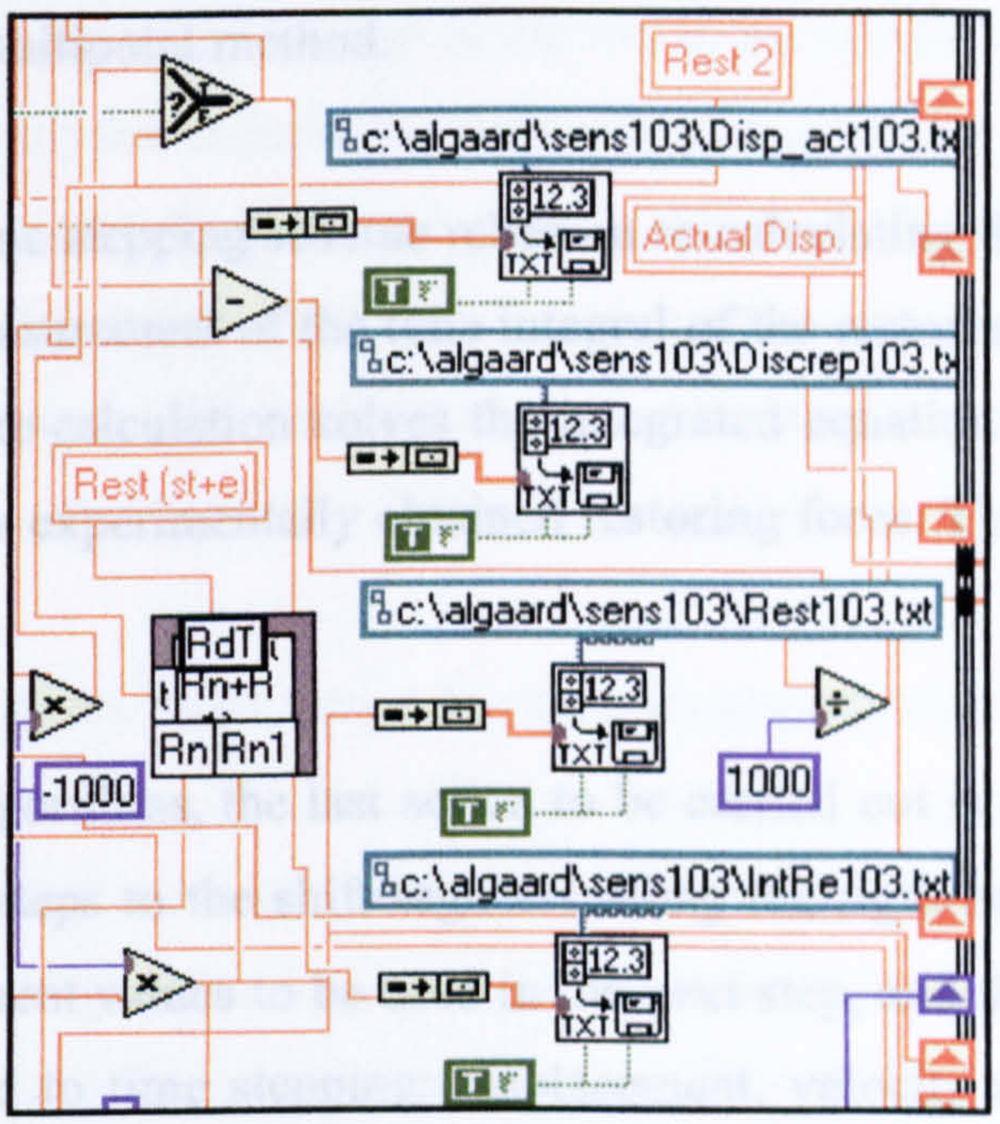


Fig. 6.30 Data logging macros in main loop

A certain times during the development of the implementation system, it was desirable to trace the changes in the variables during the time steps. This had to be done by coding the macros into the controller itself. Although very computationally expensive, this provided a unique opportunity to evaluate the various processes. This was particularly valuable during the development of the computation of the time integral of the restoring force. However, for normal use, it is not required to trace the intra-step changes, and the function is not included.

When employing the integral form algorithm, certain tasks related to the time integration are carried out after the controller has imposed the target displacement. These tasks are referred to as concluding calculations, and include a computation related to the time integral of the restoring force and the velocity component in the time stepping scheme.

As indicated, the time integral of the restoring force is in principle computed within the controller. However, during the start-up, this computation is not completely reliable, so the alternative 2-point method is used, as described in section 5.4. The 2-point variable is also computed throughout the test to provide comparison and verification for the multipoint method.

The integral form time stepping scheme relies on re-calculating the velocity following the experimental measurement of the time integral of the restoring force, as described in section 4.2. The re-calculation solves the integrated equation of motion, equation (4.4a), using the now experimentally obtained restoring force. The operation is shown in figure 6.19b.

With all the main algorithms, the last action to be carried out is to write all variables used in subsequent steps to the shift registers along the right hand side of the loop. This enables the current values to be used in the next step, and the variables typically include those related to time stepping: displacement, velocity and acceleration and some related to the controller: valve signal, limit of dead-zone and the restoring force. On completion of this task, the main algorithm will proceed to the next time step and continue doing this until the test has completed. Only then will any information be passed out of the main loop.

6.2.2.5 Final data logging and completion of the main loop

Once the final iteration of the main loop has been carried out, the few tasks that rely on information from within the main algorithm can be carried out. These largely consist of data logging and display functions.

When using wired connections for variables passed out of loops, arrays will build up during each iteration and the information will only be available on completion. If it is desirable to carry out operations on whole arrays, these can naturally done be on completion of the loop. In the pseudodynamic implementation system described here, the set-up will combine the force and displacement histories to create hysteresis loops. This can only be plotted as a graph on the front panel at the end of a test as it requires a cluster of arrays as input. The coding of the combination and plotting of these

variables is shown in the bottom middle of figure 6.12. The full arrays may at this stage also be saved to disk in a single operation to essentially provide an additional copy of the data saved at the end of each time step.

Another graph that is created at the end of the test is the comparison between a stored and a new graph, using the variable Updated. This is generated in the middle above the main loop in figure 6.12 and displayed on the main front panel in the screen denoted Updated. As the response from each test is saved individually, the comparison can of course also be done *a posteriori* using, for example, spreadsheet programs.

6.2.3 Operation of the software based implementation system

This section evaluates the execution of the software based implementation system, giving an indication to the overall control of the experimental component. It will, in particular, discuss the semi-continuous running through the novel step continuity system discussed in sections 5.3.5. The so-called step-continuity system aims to avoid a hold period and/or unloading at the completion of each time step. Continuous or semi-continuous running implies that actuator motion is essentially continuous from the end of one step to the start of the next. This can be contrasted to "classic" pseudodynamic testing, e.g. (Negro 1996), where a hold period is explicitly included for taking force measurements and carrying out data logging and computation of the next displacement target. This is enabled by not necessarily switching dead-zones at completion of the step, as described in section 6.2.2.4.4.3 and by defining immediate completion of the control loop upon reaching the target, as described in section 6.2.2.4.4.5. The section further contains a short discussion on the time requirements for each process within the implementation system.

6.2.3.1 Semi-continuous running

Pseudodynamic tests were until recently generally executed using the so-called classic method, where a hold period was prescribed for each displacement step, during which the actuator remained stationary to allow force measurements and computation of the next displacement steps to take place. However, it was realised that by omitting the

hold period, higher strain rates and more realistic deformation of the specimen structure could be enabled. During the second half of the 1990s, continuous implementation of pseudodynamic tests became more common, e.g. (Magonette *et al.* 1998), (Williams *et al.* 1999), (Magonette *et al.* 2000), (Thewalt & Mahin 1994).

The software based implementation system documented here is referred to as a semi-continuous system. It is continuous in the sense that no hold periods are prescribed, but not fully continuous, as the code has to switch between control and time integration. Essentially, the system relies on the execution of the time integration component being so quick that control is not lost for an appreciable amount of time.

Earlier in this chapter, in section 6.2.2.4.1, the data flow within the main loop was discussed, and effectively showed how first the time stepping computation was carried out, then the control before finally display functions and data logging. It is at this stage important to appreciate that although these processes utilise comparable amounts of code, the controller is by far the most computationally expensive as it sub-iterates between 50-1000 times during each time step. Additionally, the controller is dependent on the hydraulic system keeping pace. In practice therefore, the control process may take up over 99% of the CPU time.

The exact coding of the hardware communication is such that the finished signal for the servo valve is written as an update. The macro will thus continue transmitting the same signal also when the controller does not run. However, the finished signal will be correct for the conditions present at the time of cessation of the control loop, and may not be adjusted until the loop re-starts iterating.

As described in section 6.2.2.4.4.3, the implementation system is coded such that when the controller registers that the target is met, no action is taken in that time step to stop the actuator. This is the core of what makes the implementation continuous. Although the signal generated in the last control iteration typically only gives rise to a low actuator velocity, the signal will, if left unchanged, result in overshoot. Only at the start of the next time step, if the next target displacement is in the opposite direction, will the actuator be stopped and reversed. So the inter step time for correction of the signal will be the time of completion of the present main loop and

carrying out one time integration step and one control iteration in the new loop. As discussed in the next section, the relative time requirements for each process are such that this becomes a viable system.

In practise, the system of switching between the processes functions robustly. If a significant proportional gain is employed, the actuator velocity will be low when approaching the target and overshoot will be negligible. In any case, as the force measurement is taken before the next time integration is carried out, the effect of potential overshoot would be limited. The controller front panel enables real-time display of the measured displacements with the corresponding targets superimposed. Although fairly computationally expensive, creating this display provides an excellent possibility for evaluating the quality and continuity of the displacement control. An example of such curves can be seen in figure 6.31

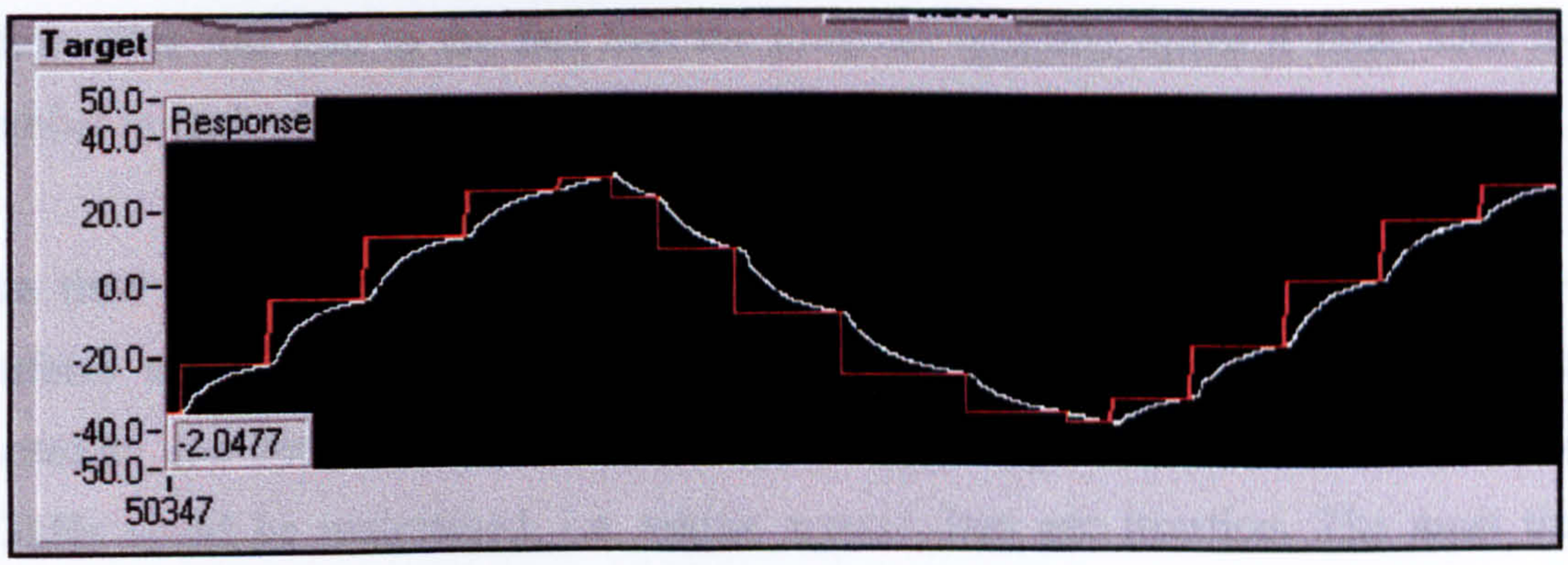


Fig. 6.31 Real time measured displacements with applicable targets

6.2.3.2 Time requirements

A small study into the time requirements of the various processes within the implementation system has been carried out. This has partly been done to justify the theory behind the semi-continuous implementation system and partly to optimise the execution speed.

During the development of the system, a timer was coded into the controller and main loop to monitor the various time requirements. It was quickly found that the apparent calculation of the displacement step took a variable, and sometimes substantial,

amount of time. The processes within the main loop were cut down to the essentials to attempt to isolate the cause, but the duration of each loop could still range from around 30ms to over 1s. Naturally, loss of active control for 1s would result in considerable overshoot in the semi-continuous system and was not a satisfactory option, when on the other hand it was known that the computation could be carried out in less than 30ms.

Further tests revealed that the cause of the occasionally very slow generation of displacement steps was the creation of temporary files on the hard drive related to the graphical representation of the progress of the test. It was found that by minimising the number of displayed graphs to a single one showing the target displacement of each time step, a consistent time requirement for generation of a new displacement step of 30-40ms could be maintained. In the main loop, the data logging process was found to be somewhat resource consuming and required a significant portion of he 30-40ms. This was due to the fact that the program typically opens 4 files, adds in a number in each of them and then closes the files.

The time requirements of the controller were also investigated to see if this was uniform and optimised. Again, all redundant functions were removed from the controller and iteration rates monitored. It was generally found that a rate of around 1/3kHz could be maintained, i.e. taking around 3ms per iteration. The most time consuming process was here believed to be the hardware communication macros, which, of course, are indispensable.

The effect on the time requirement when using more elaborate time stepping schemes and controllers was also considered to see if this could affect the accuracy and implementation speed. It was generally found that the level of complexity did influence the iteration speed somewhat, but not significantly enough to select specific algorithms on this basis. The iteration time for the main loop ranged from around 30ms with the simplest central difference method while the integral form method required around 40ms. The sensitivity of the system to the implementation speed and choice of algorithm etc. is discussed further in the next chapter and in Algaard *et al.* (2001a) and Algaard *et al.* (2001c).

Chapter VII:

VERIFICATION AND SENSITIVITY STUDY

This chapter aims to evaluate the performance of the complete implementation system by assessing its reliability and accuracy. The chapter contains two main sections: an account of the sensitivity study carried out and a verification analysis. The sensitivity study addresses the sensitivity of the system with respect to accuracy and response as affected by the implementation speed, choice of time stepping scheme, time step size etc. It involves experimental testing of two specimen types, a reinforced concrete column and a steel column. The verification test aims to verify the implementation system as a whole, and involves a real-time dynamic snap-back test and pseudodynamic tests on the same steel column.

7.1 SENSITIVITY STUDY

Considerable work has been carried out to study the error propagation effects in pseudodynamic tests, e.g. (Combescure & Pegon 1997), (Shing & Vannan 1990), but, to the best of the author's knowledge, little or no research has been performed to investigate implementation sensitivities in general. Pseudodynamic implementation sensitivities may range from those concerned with the time integration to those concerned with the experimental implementation. In terms of time integration, the choice of scheme, the time step size and other details will affect the response while on the experimental side, the rate and method of loading, controller capabilities, accuracy obtained and measurements details may equally influence the displacement history.

In this section, the relationships between: *time step size*, *speed of implementation*, *discrete step accuracy* and *final system response* are considered. Two different time stepping algorithms are employed to evaluate how the above relationships may be affected by algorithmic differences in both time integration and control.

7.1.1 Implementation variables

As introduced above, implementation variables include those from both algorithmic and control related origins. The variables will be discussed in the following sections.

7.1.1.1 Algorithmic variables

Broadly speaking, the algorithmic implementation variables are those related to the time stepping scheme. All time stepping algorithms display some form of approximation, resulting in response inaccuracies. The inherent approximations in the linear case when obeying stability limits may extend only to periodicity errors, while under non-linear conditions they can additionally appear in terms of amplitude errors.

Regardless of which time stepping scheme is employed, the algorithmic errors will increase with increasing time step sizes. This is due to the fact that increasing time step sizes increase the duration over which the time integration variables are linearised. All the time stepping schemes described in this paper are second order accurate and assume constant velocity for the duration of the time step. In the linear case, the approximation will be due to the assumption of constant acceleration acting over the entire step, while in the non-linear case, the intra-step stiffness variation will introduce additional approximations.

The time step size naturally becomes the primary variable, as most algorithms can be made accurate if small enough time steps are employed. However, more complex algorithms employ a range of other variables that may influence the response. With for example the integral form algorithms, a replacement approximation for the tangent stiffness is used. Discrepancy between the replacement and the actual tangent stiffness may introduce period elongation, as discussed in section 4.6.2.2. On the other hand, the α-Operator Splitting algorithm, as described by Combescure & Pegon (1997), introduces beneficial numerical damping through the inclusion of the α variable. Also this algorithm employs a replacement approximation for the tangent stiffness.

The algorithms themselves display inherent properties affecting the generation of the response. With increasing time step sizes, algorithms generally create increasing period error, but the level and direction of the error will vary between the algorithms. This is also the case with the amplitude errors that are sometimes present. Detailed evaluation and comparisons of traditional algorithms can be found in Wilson & Bathe (1976) and Géradin & Rixen (1994).

7.1.1.2 Control variables

The control variables extend to the variables that are related to the implementation of the experimental component of the test. This physical experimentation introduces a whole range of variables and scope for error generation. In the controller, the desired proportional and integral gains are set, and these determine the speed and velocity profile of the actuator. It is envisaged that the gain settings may potentially affect the response directly through rate dependency in the specimen material properties or indirectly through the introduction of control inaccuracies.

Experimental errors in pseudodynamic tests can generally be categorised as measurement or control errors. As the measurement system is essentially a part of the apparatus and remains the same throughout the testing, this does not introduce additional variables. The control errors on the other hand describe the inaccuracy with which the prescribed displacement steps are imposed. This normally implies the discrete error introduced in each displacement step, but also the potential cumulative effect of these.

While the primary variable related to control is probably the experimental implementation speed, the fact that for example a different controller is employed with the integral form algorithm should be considered. This is primarily because the additional calculations may reduce the iteration speed of the controller, as well as displaying different time requirements for the time integration. Additionally, the control settings will affect the computation of the time integral of the restoring force, creating a combined control/algorithmic effect. With all the other time stepping schemes, only the final measure of the restoring force is used, making it more exclusively subject to the control variables.

7.1.2 Evaluation programme

The evaluation programme was attempted to obtain relationships and trends between the following four properties: time step size, speed of implementation, accuracy and response. These have been categorised into 6 relationships, investigating the effect of

one variable on another for given conditions. The relationships are summarised in table 7.1 below.

1	**Time step size**	vs.	**Speed**	for given	**Accuracy**
2	**Time step size**	vs.	**Accuracy**	for given	**Speed**
3	**Speed**	vs.	**Accuracy**	for given	**Time step size**
4	**Response**	vs.	**Time step size**	for given	**Accuracy**
5	**Response**	vs.	**Speed**	for given	**Time step size**
6	**Response**	vs.	**Accuracy**	for given	**Time step size**

Table 7.1 Relationships between properties.

Some of the relationships in table 7.1 are of course interdependent, with one relationships being the inverse of the other. In addition to these 6 relationships, the effect the time integration algorithm may have on the response, accuracy and speed has also been investigated.

In order to evaluate the relationships discussed above, repeated pseudodynamic tests have been carried out on two different dynamic systems. These comprise the reinforced concrete stub column and the slender steel column, described in sections 3.3.3.1 and 3.3.3.2, respectively. Both structures were assigned lumped virtual masses on top, yielding inverted pendulum systems. The masses were tuned to create structures with fundamental periods of around 1.2 seconds, with the concrete structure being given a mass of 46800kg and the steel structure 1714kg. The structures were exposed to a scaled 1957 NS Port Hueneme accelerogram, and the first 4 seconds of the response were modelled using the central difference and the Newmark Implicit – Integral Form methods. Zero viscous damping was applied throughout.

In order to vary the implementation speed, the proportional and integral gain settings were adjusted. This was done in a manner which intended to optimise the speed/error relationship. Most scope for velocity increase was found through increases in the proportional gain. If the integral gain was set too high, this resulted in considerable overshoot.

7.1.2.1 Tests on the reinforced concrete specimen

The reinforced concrete column was designed as short, stubby and non-flexible and the intention was for it to display a significant non-linear behaviour. Furthermore, the specimen was slightly damaged prior to testing by imposing displacements exceeding those expected during pseudodynamic sensitivity tests. This was done to ensure repeatability of tests by avoiding further damage taking place and to ensure non-linear, dissipative behaviour. Typical force-displacement curves for the specimen can be seen in figure 7.1 below. Maximum displacements of ±10mm were aimed for during these tests. The detailed design of the specimen is accounted for in section 3.3.3.1.

Time step sizes of 0.004, 0.04, 0.08 and 0.16 seconds were employed, requiring 1000, 100, 50 or 25 steps, respectively, to model the first 4 seconds of the response. Total testing time varied from 8 seconds to 2 minutes and 18 seconds.

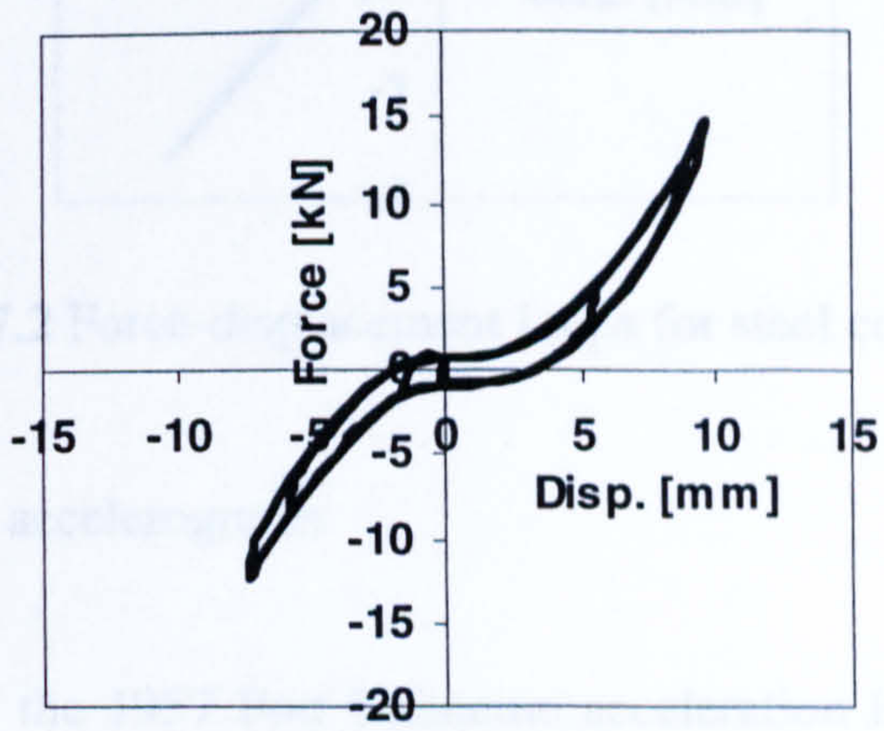


Fig. 7.1 Force-displacement loops for concrete column

7.1.2.2 Tests on the steel specimen

A slender steel column was designed for two reasons. Firstly because it would display a near perfectly linear force-displacement relationship providing a contrast to the non-linear reinforced concrete specimen. Secondly, due to its flexibility, the mass required to provide the desired frequency is small. This enables the true dynamic system to be created, opening for the possibility of a real reference solution, a so-called *snap-back*

response, to be obtained and used in the verification test. Some of the tests on the steel specimen were also reproduced numerically. This was done by coding the time stepping schemes for a SDOF linear system in Microsoft Excel 2000.

The column comprised a 1600mm long rectangular hollow steel section providing a second moment of area in the weak direction of 46.7cm^4, as detailed in section 3.3.3.2. The section was expected to remain elastic over the full stroke of the actuator (±50mm). Typical force-displacement loops can be seen in figure 7.2. In this case, time step sizes of 0.04, 0.08 and 0.16 seconds were selected, requiring 100, 50 or 25 steps. Total testing time here ranged from 9 to 50 seconds.

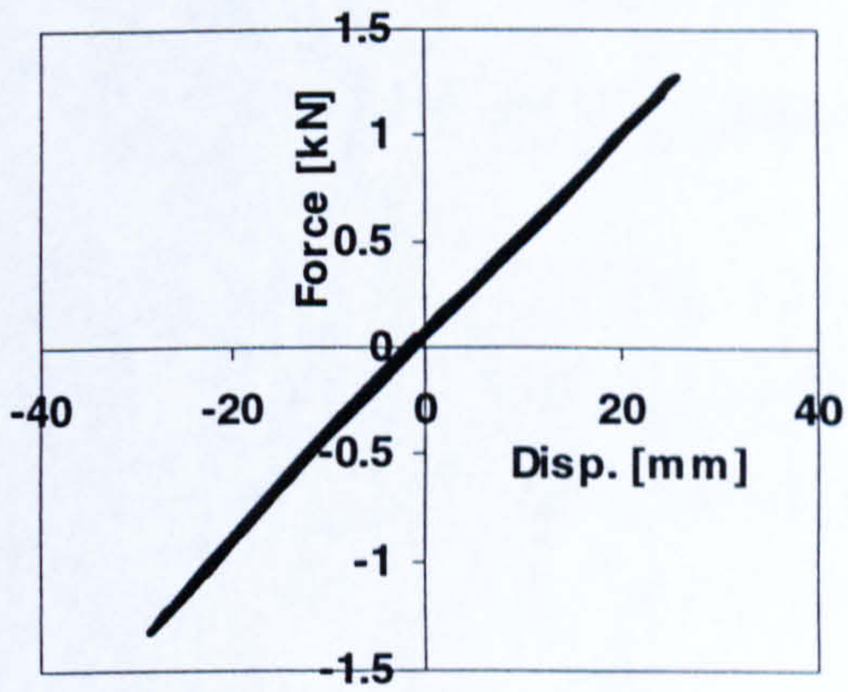


Fig. 7.2 Force-displacement loops for steel column

7.1.2.3 Ground motion accelerograms

The NS component of the 1957 Port Hueneme acceleration history was selected for these tests. As this accelerogram consists mainly of a single acceleration impulse, as seen in figure 7.3, it was believed that this might reveal differences within the implementation better than a more varying history like for example the 1940 El Centro acceleration.

In order to investigate the effect on the response resulting from differences in the time step size, it is imperative that the structure is always subjected to identical loading for all time step sizes. This will not normally be the case if a single acceleration value is selected for each step, as is typically the case with non-integral form algorithms. The

sampling period of the applied accelerogram is 0.004s and step sizes vary from 0.004s to 0.16s. Truncation techniques are therefore required for all step sizes different from that of the sampling period.

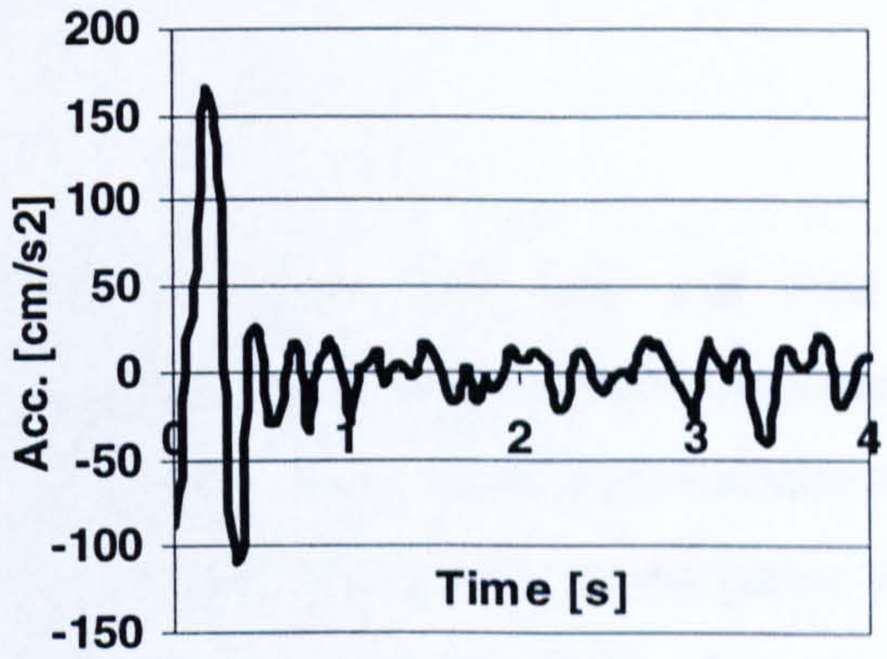


Fig. 7.3 Port Hueneme NS accelerogram.

In the case of the 0.16s time step size, only every 40th available acceleration value would be used if no averaging technique were employed. It can easily be seen how the peak acceleration could be missed in this case. With the integral form algorithms, the time integral of the acceleration for each time step will be calculated *a priori* with all peaks being taken into account.

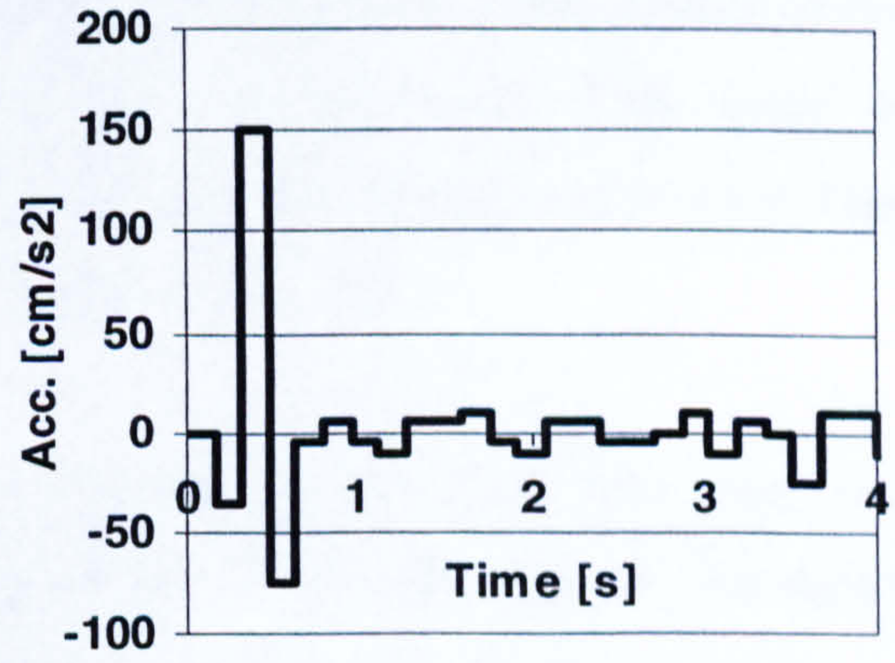


Fig. 7.4 Artificial accelerogram, sampling period 0.16s

To ensure that the structures were subjected to the same base excitation, regardless of time step size and time stepping scheme, an artificial accelerogram was generated. This aimed to resemble the Port Hueneme excitation, whilst keeping the sampling rate such that all time stepping schemes would interpret it the same way. This basically meant that the sampling period of the artificially generated accelerogram had to be the

same as the largest time step size, or in other words 0.16s. When employing smaller time step sizes, the same acceleration value would simply be repeated 2, 4 or 40 times for the 0.08, 0.04 and 0.004s time step sizes respectively. The synthetic accelerogram can be seen in figure 7.4.

7.1.3 Results

For the given conditions, i.e. time step size and time integration algorithm, proportional and integral gains were varied in order to investigate the effect on the duration, accuracy and response. This should theoretically indicate results to all 6 parts of the sensitivity test, while also giving an indication to the effect of the time integration algorithm. The initial results from the investigations into the 6 relationships are all based on test carried out on the reinforced concrete column. Some further results are based on tests carried out on the steel column.

7.1.3.1 *Time step size* vs. *speed* for given *accuracy* (Relationship 1)

For this test, data obtained with the Newmark Implicit – Integral Form algorithm using the numerically integrated measure of restoring force was used. The time step size vs. speed relationship was considered independent of the time stepping scheme and a characteristic of the controller only. This could be argued as the time requirement for the time integration and controller were found to be similar for all schemes, as explained in section 6.2.3.2.

Around 10-20 tests were carried out for each time step size but with varying gain settings to obtain a range of implementation speeds. The selected results below should be considered a typical representation of the range of results.

By setting first the required accuracy for all errors to be less than 0.20mm, the results for the various time step sizes can be seen below in table 7.2. By setting the maximum error to be somewhat larger, 0.5mm, the results are presented in table 7.3.

Time step size [s]	Duration [m:s]	Error [mm]
0.004	2:18	0.20 or more
0.04	0:15	0.15 – 0.20
0.08	0:15–0:27	0.15 – 0.20
0.16	0:08–0:11	< 0.20

Table 7.2 Durations required for keeping an error threshold of 0.20mm

Time step size [s]	Duration [m:s]	Error [mm]
0.004	1:30	0.30
0.04	0:11	0.40 – 0.60
0.08	0:09	0.30 – 0.40
0.16	0:08 – 0:11	< 0.20

Table 7.3 Durations required for keeping an error threshold of 0.50mm

7.1.3.2 *Time step size* vs. *accuracy* for given *speed* (Relationship 2)

The results obtained here are based on the same experimental data as that used in the previous section. For this relationship, time step size vs. accuracy for given speed, the results should essentially be the inverse of the above. Allowing for example 12s for the duration of the tests, the accuracies obtained with the various time step sizes are as summarised in table 7.4.

Time step size [s]	Duration [m:s]	Error [mm]
0.004	N/A	N/A
0.04	< 0:12	0.40 – 0.60
0.08	< 0:12	0.30 – 0.40
0.16	< 0:12	< 0.15

Table 7.4 Accuracy obtained for constant 12s test durations

When using the 0.004s time step size, no tests could be completed within the allowed 12s, so no experimental results exist for this time step size in the table above. This was also the case if the test duration was set to 20s. The results for the other time step sizes for the constant 20s duration tests can be seen in table 7.5.

Time step size [s]	Duration [m:s]	Error [mm]
0.004	N/A	N/A
0.04	< 0:20	0.20
0.08	< 0:20	0.20
0.16	< 0:20	< 0.10

Table 7.5 Accuracy obtained for constant 20s test durations

Again, when allowing 35s for the test duration, tests with the 0.004s time step size could not be completed. The results from the other time step sizes are given in table 7.6.

Time step size [s]	Duration [m:s]	Error [mm]
0.004	N/A	N/A
0.04	< 0:35	0.05 – 0.15
0.08	< 0:35	0.05 – 0.15
0.16	< 0:35	0.05 – 0.07

Table 7.6 Accuracy obtained for constant 35s test durations

7.1.3.3 *Speed* vs. *accuracy* for given *time step size* (Relationship 3)

This section considers a particular time step size at the time to obtain the relationship between speed and accuracy. As in the two previous sections, the data from the tests obtained using the integral form algorithm has been used.

Four sample points have been provided for each time step size to give an indication to the effect on the error as the test durations vary. The results are presented in tables 7.7 to 7.10 for the time step sizes of 0.004, 0.04, 0.08 and 0.16s, respectively.

Time step size [s]	Duration [m:s]	Error [mm]
0.004	2:18	0.20
0.004	2:02	0.30
0.004	1:30	0.30 – 0.40
0.004	1:10	0.70

Table 7.7 Accuracy and speed for 0.004s step size

Time step size [s]	Duration [m:s]	Error [mm]
0.04	1:00	0.05
0.04	0:36	0.10
0.04	0:15	0.20
0.04	0:11	0.40

Table 7.8 Accuracy and speed for 0.04s step size

Time step size [s]	Duration [m:s]	Error [mm]
0.08	0:37	0.05 – 0.10
0.08	0:28	0.10 – 0.15
0.08	0:16	~ 0.20
0.08	0:09	~ 0.40

Table 7.9 Accuracy and speed for 0.08s step size

Time step size [s]	Duration [m:s]	Error [mm]
0.16	0:11	0.03 – 0.06
0.16	0:09	0.07
0.16	0:08	0.06
0.16	0:08	0.05

Table 7.10 Accuracy and speed for 0.16s step size

7.1.3.4 Analysis of speed – accuracy – time step size relationship

Based on the numerical data in tables 7.2 to 7.3 and 7.7 to 7.13, the graphical representation in figures 7.5 and 7.6 was created. Consider first figure 7.5, which shows the relationship between the test duration and the maximum step error, i.e. the largest discrepancy between the targeted and achieved positions, for each time step size. Here it is clear that the increased test durations, or in other words slower tests, maintain better accuracy. Furthermore, it shows that increasing time step sizes also enable superior accuracy. It is also noted that the largest time step size is very accurate even for the shortest test durations, and that with the smallest time step this level of accuracy cannot be achieved even with considerable test durations.

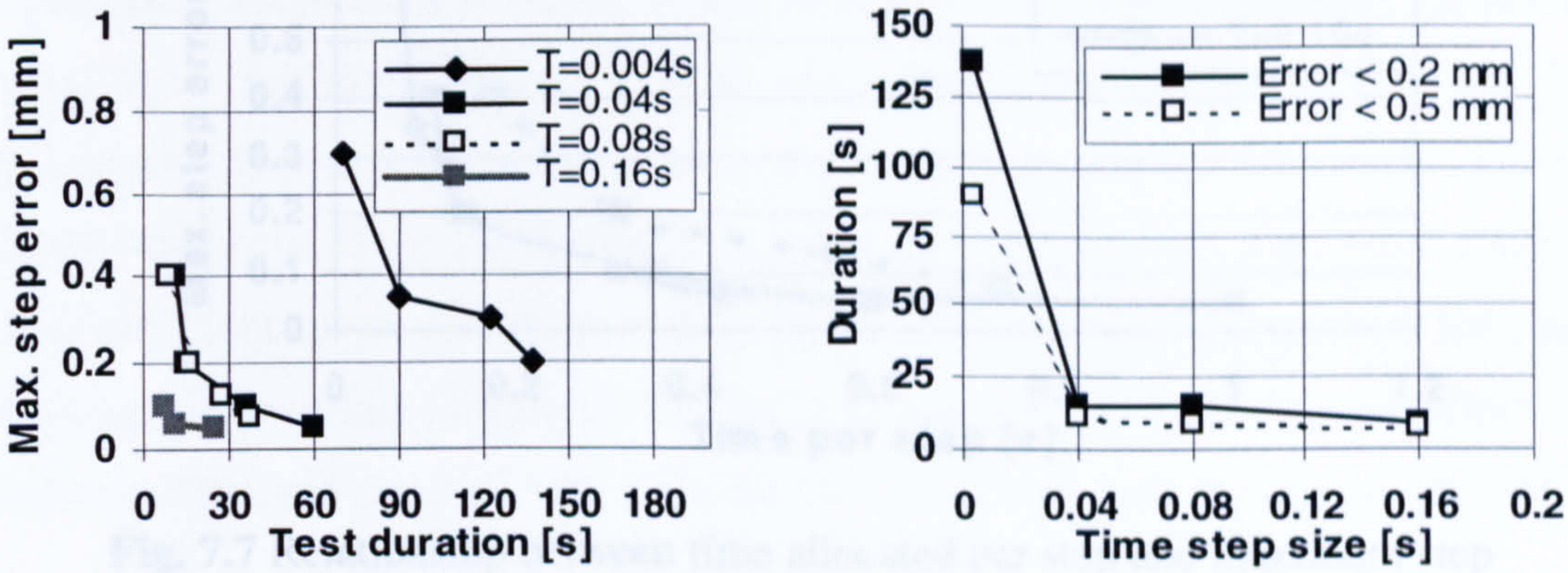


Fig. 7.5 and **Fig. 7.6** Test duration vs. maximum step error and time step size vs. test duration, respectively.

Figure 7.6 gives an indication of the testing time required to maintain a minimum accuracy of 0.2mm and 0.5mm for the range of time steps. Again, it shows that the smallest time step, 0.004s, requires considerable more time than the larger ones. The differences between the 0.04, 0.08 and 0.16s steps are not considerable, but still display the general trend that the larger time steps require less overall testing time to maintain a specified accuracy.

The trend displayed in figure 7.5 suggests that increasing the test duration and the time step size improve accuracy. As longer test durations provide more time per time step, more time is available for control. This allows for more control iterations and lower actuator speeds, so it is evident that improved accuracy can be obtained.

Similarly, increased time step sizes imply more time available for each step, yielding the same effect.

As an example, implementation using the 0.08s step size can maintain a minimum accuracy of 0.2mm when running the test in 15s while the 0.004s step size requires 135s for similar accuracy. However, 15s allows 0.3s per step using 0.08s steps, while 135s allows only 0.135s per step using 0.004s steps. Similarly, the very fast and accurate test using 0.16s steps, 0.07mm maximum error for a 9s duration, does in fact allow 0.36s per step, while comparable accuracy using the 0.04s steps is obtained by providing 36s test duration, or, again, 0.36s per step.

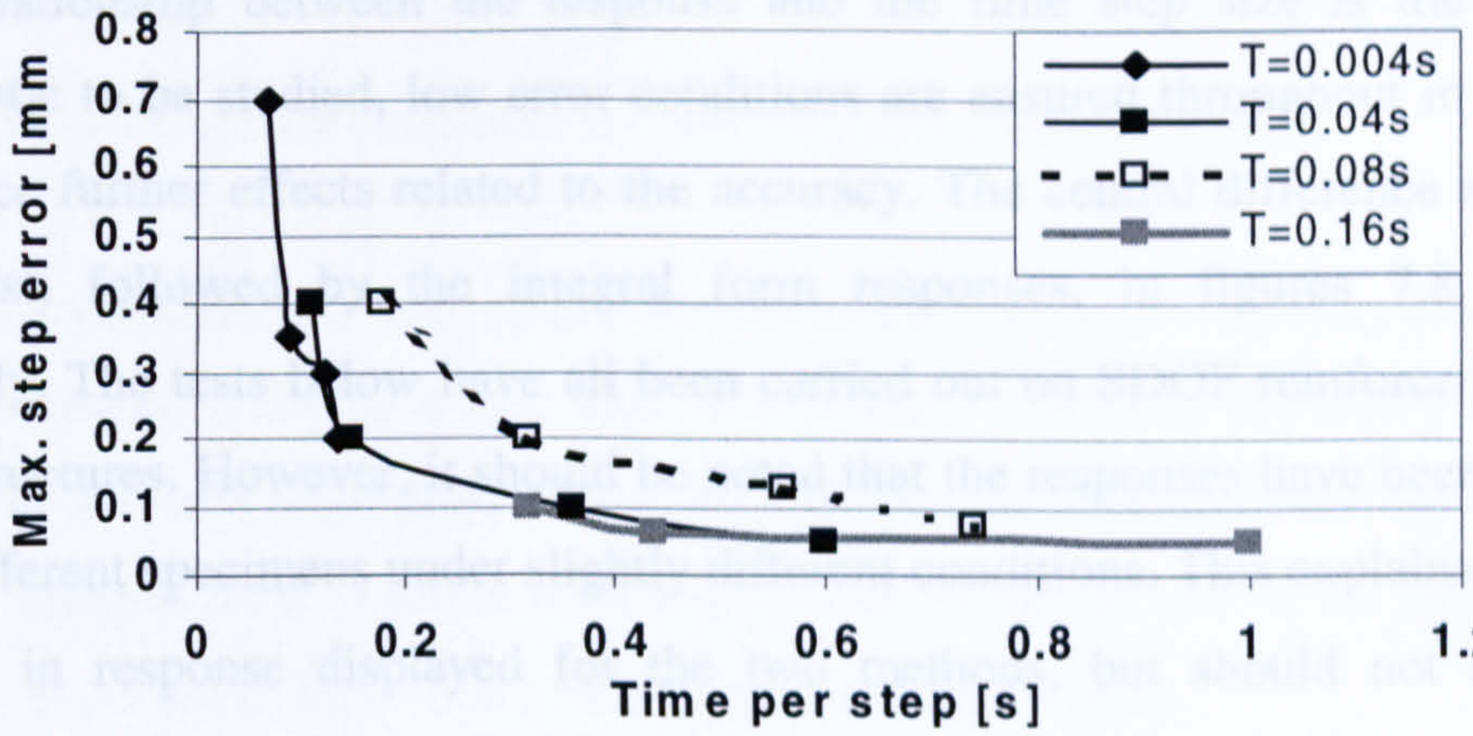


Fig. 7.7 Relationship between time allocated per step and maximum step error for various time step sizes.

It appears therefore that the discrete step accuracy is mainly a function of the implementation time allowed for each step. Figure 7.7 shows that accuracy appears to be a function of time per step, tending asymptotically towards a minimum implementation time of 0.05s per step and minimum displacement error of 0.05mm. The fact that for the larger time step sizes the actuator has to travel a considerably longer distance per step appears to be irrelevant.

7.1.3.5 *Response* vs. *time step size* for given *accuracy* (Relationship 4)

While the three preceding sections were concerned with the maintained accuracy, the following three are concerned with the generated *response*. The response provides an indication to the overall effect the implementation characteristics have on the test,

while the accuracy measures the maximum error occurring, while imposing each individual displacement step. One or even repeated large errors may not affect the overall response very much. However, existence of cumulative experimental errors and algorithmic effects are likely to influence the response, while not affecting the accuracy significantly. As algorithmic effects are likely to affect the response considerably, both the Newmark Implicit - Integral Form and the central difference methods have been considered.

This section considers the response to an artificial accelerogram, figure 7.4, obtained using the two time stepping schemes employed with the three largest time step sizes. As the relationship between the response and the time step size is the principal characteristic to be studied, low error conditions are ensured throughout in order not to introduce further effects related to the accuracy. The central difference results are shown first, followed by the integral form responses, in figures 7.8 and 7.9, respectively. The tests below have all been carried out on SDOF reinforced concrete column structures. However, it should be noted that the responses have been obtained on two different specimens under slightly different conditions. This explains the small difference in response displayed for the two methods, but should not affect the response difference obtained with different time steps with each integration algorithm.

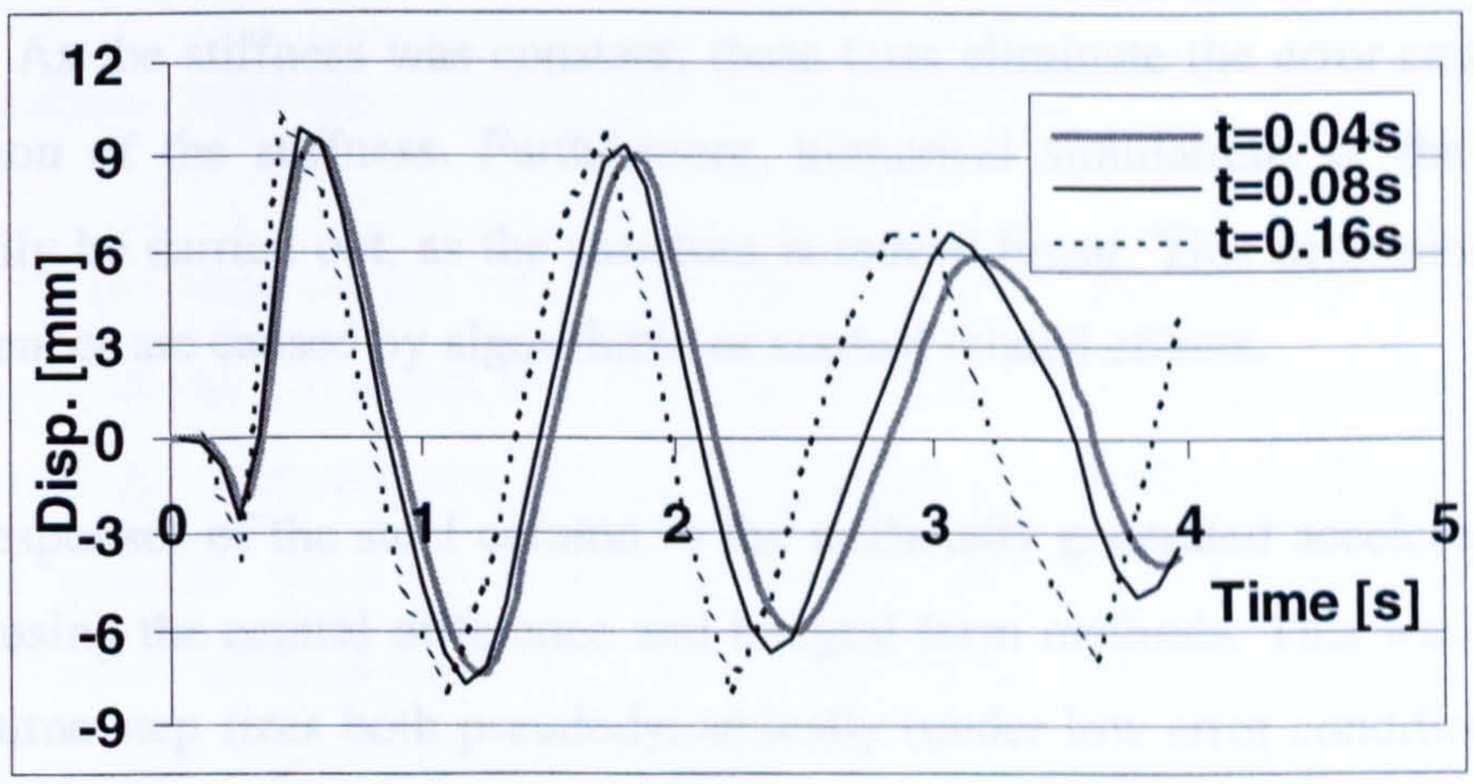


Fig. 7.8 Response obtained with the central difference method and various time step sizes

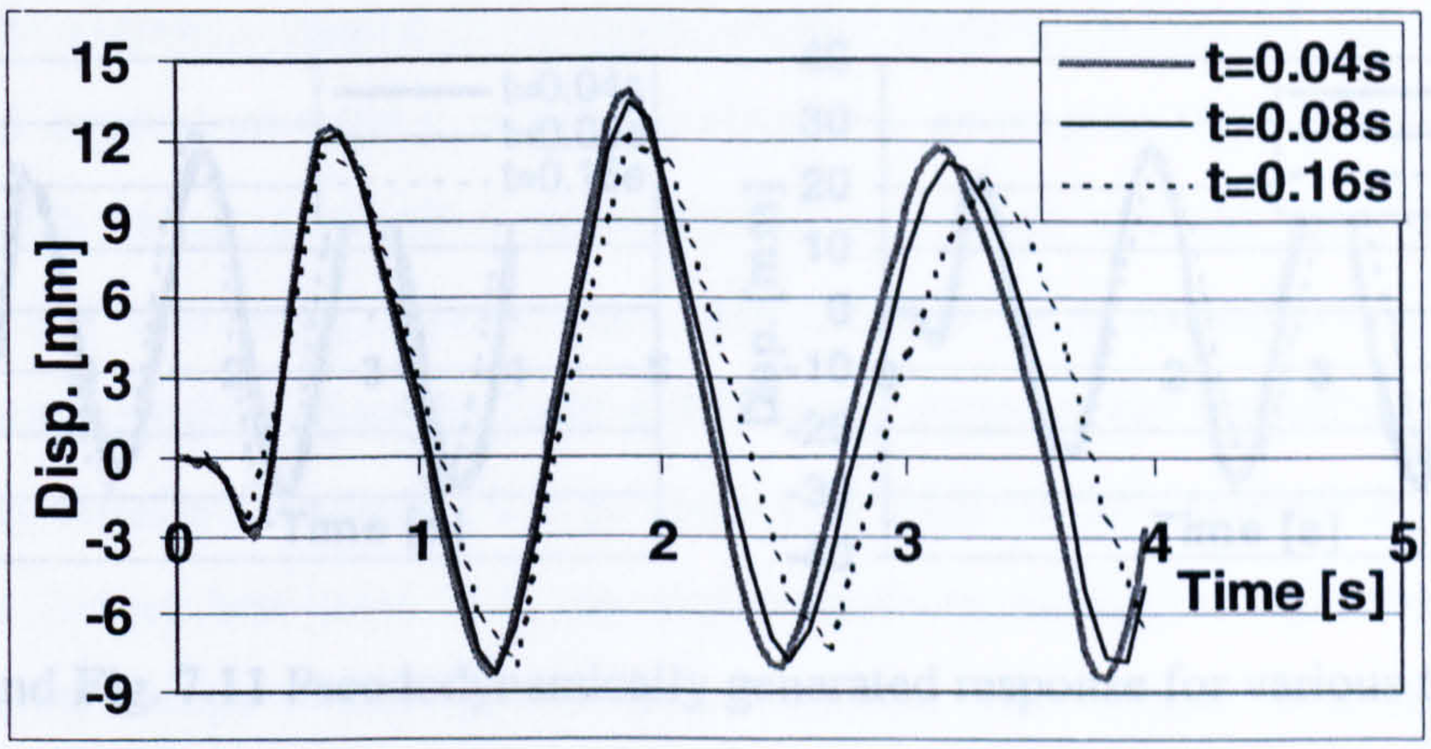


Fig. 7.9 Response obtained with integral form algorithm and various time step sizes

7.1.3.6 Further results for *Response* vs. *time step size* (Relationship 4)

In the previous section it was found that large differences in the response resulted from the changes in the time step size. It was also found that the effect was vastly different for the two algorithms concerned, the central difference and the Newmark Implicit – Integral form. Further tests were therefore carried out on the steel column, which was expected to behave in a near perfectly linear fashion.

These tests go further in revealing the origin of the observed differences in the response. As the stiffness was constant, these tests eliminate the error caused by the linearisation of the stiffness. Furthermore, numerical simulations of the excitation could easily be carried out, as the structure is indeed linear. This determines whether the differences are caused by algorithmic or control related effects.

System responses of the steel column to the artificially generated accelerogram were obtained using the central difference and integral form methods. This was done for a range of time step sizes both pseudodynamically (under low error conditions) as well as numerically. The results from the pseudodynamic tests can be found in figures 7.10 and 7.11 for the central difference and integral from methods, respectively, while the numerical equivalents can be found in figure 7.12 and 7.13.

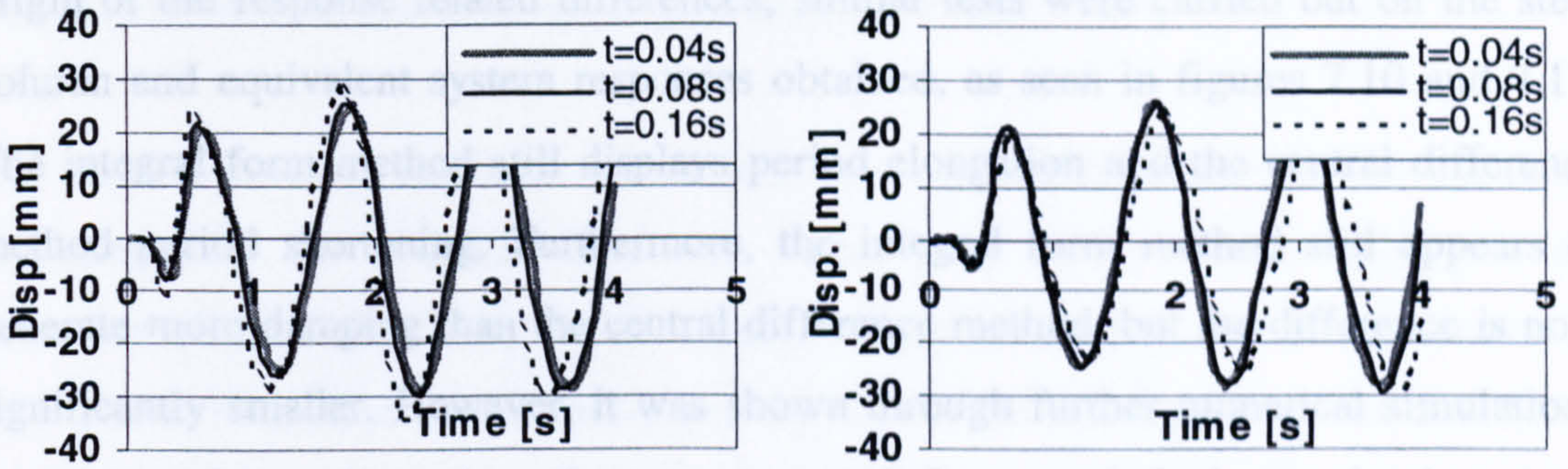


Fig. 7.10 and **Fig. 7.11** Pseudodynamically generated response for various time step sizes obtained with central difference and integral form methods, respectively.

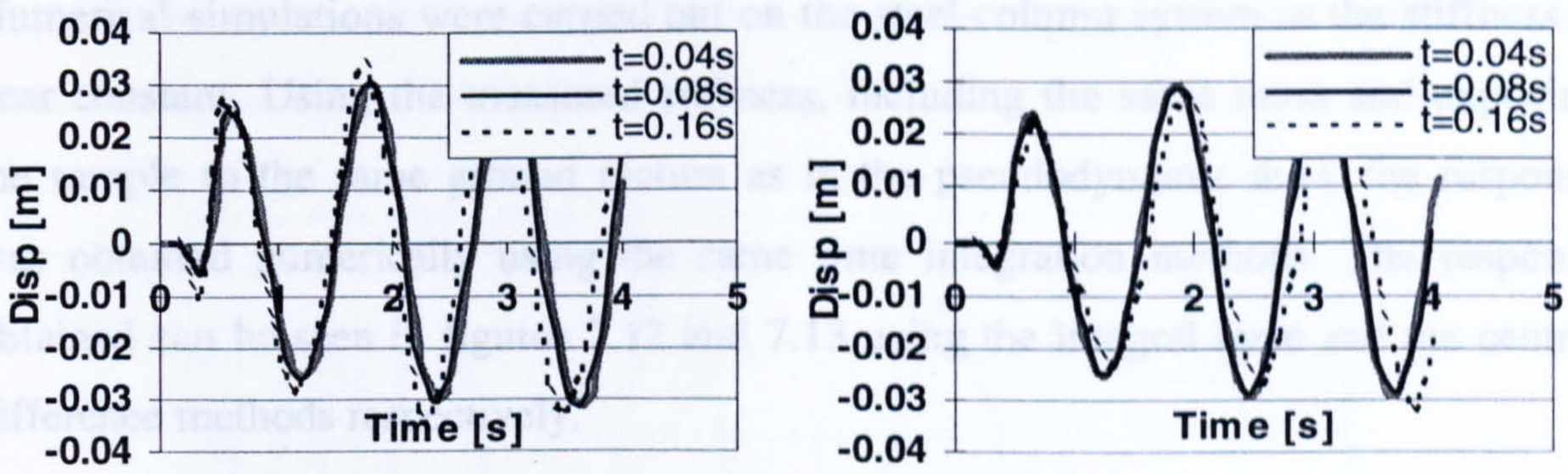


Fig. 7.12 and **Fig. 7.13** Numerically generated response for various time step sizes obtained with central difference and integral form methods, respectively.

7.1.3.7 Analysis: response - time integration relationship

Considering the tests on the reinforced concrete specimen, figures 7.8 and 7.9, it immediately becomes apparent that large differences in the response result both from the choice of the time step size and the integration algorithm. Assuming the shortest time steps generate the most accurate response, increasing the step size with the central difference method results in a period shortening, while for the integral form it results in period elongation. Additionally, the central difference method displays evidence of amplitude amplification, while the integral form method displays some evidence of amplitude decay.

While some of the effects mentioned above could have been anticipated as typical algorithmic effects, others are more unexpected. In order to attempt to isolate the

origin of the response related differences, similar tests were carried out on the steel column and equivalent system responses obtained, as seen in figures 7.10 and 7.11. The integral form method still displays period elongation and the central difference method period shortening. Furthermore, the integral form method still appears to generate more damping than the central difference method, but the difference is now significantly smaller. However, it was shown through further numerical simulations covering a longer test time that, as expected from analytical examination, both methods were energy stable for the linear system. The central difference method rather produces a higher amplitude response than the integral form method.

Numerical simulations were carried out on the steel column system as the stiffness is near constant. Using the measured stiffness, including the same mass and exposing the sample to the same ground motion as in the pseudodynamic tests, the response was obtained numerically using the same time integration methods. The response obtained can be seen in figures 7.12 and 7.13 using the integral form and the central difference methods respectively.

The numerically generated response is near identical with that obtained pseudodynamically, for both schemes and for all time step sizes, comparing figures 7.10 and 7.11 with 7.12 and 7.13. The small differences that exist are due to the exclusion of viscous damping in the numerical model and a small offset of the zero in the experimental tests. It can thus be confirmed that in the linear case the differences caused by the time integration scheme exist entirely on an algorithmic level.

The interpretation of the results obtained under non-linear conditions, i.e. from the tests on the reinforced concrete column is more difficult than under the linear conditions. The periodicity error can fairly easily be explained as a pure algorithmic effect as this is very similar to the period shift experienced under the linear numerical tests and can also be accounted for analytically. The existence of the amplitude related differences can also be partly explained by algorithmic effects. The increased amplitude with the central difference method is also present in the linear case, as is a somewhat damped response with the integral form method.

7.1.3.8 *Response* vs. *speed* for given *time step size* (Relationship 5)

This section aims to obtain a relationship between the response and the speed of implementation for constant time step sizes. This should theoretically give an indication of any rate effects. As the effects sought here are related to the control or the material behaviour of the specimen, the results are based only on the data from tests with the integral form method. Displayed below are results using the 0.004, 0.04, 0.08 and 0.16s time step sizes in figures 7.14 to 7.17, respectively. Each graph displays responses obtained with a range test durations. In all cases, the SDOF concrete structure has been subjected to the artificial accelerogram. However, it should be noted that the tests were carried out under slightly different conditions, so the response obtained with different time step sizes cannot be directly compared.

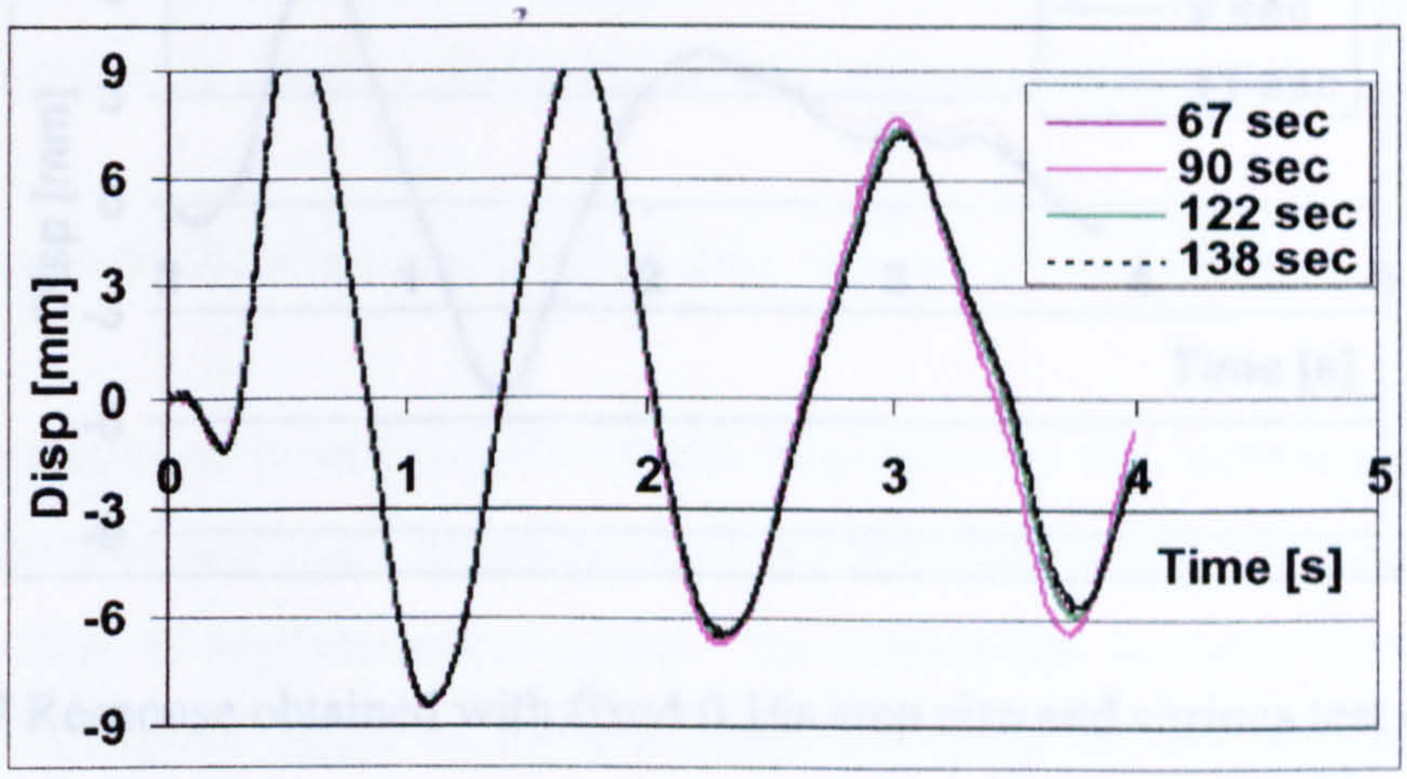


Fig. 7.14 Response obtained with fixed 0.004s step size and various test durations

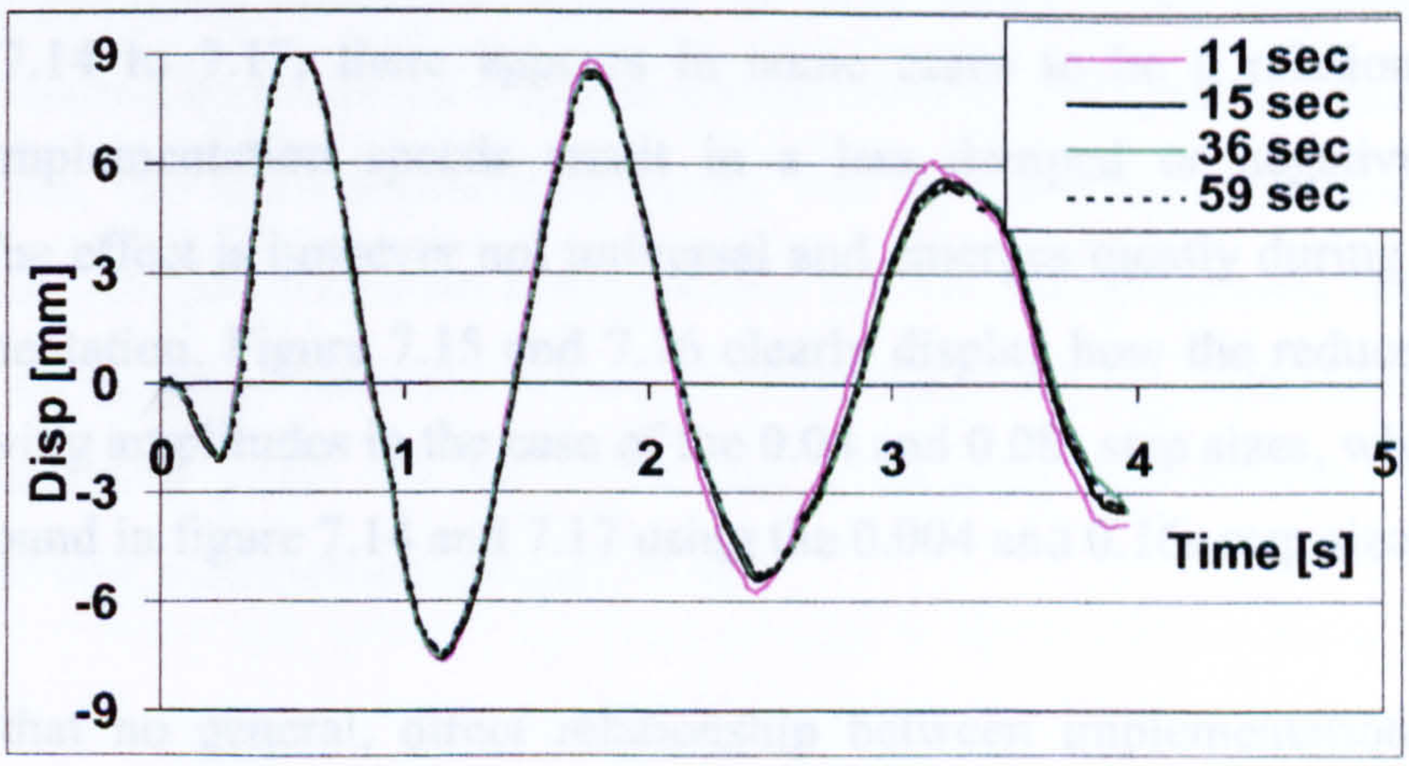


Fig. 7.15 Response obtained with fixed 0.04s step size and various test durations

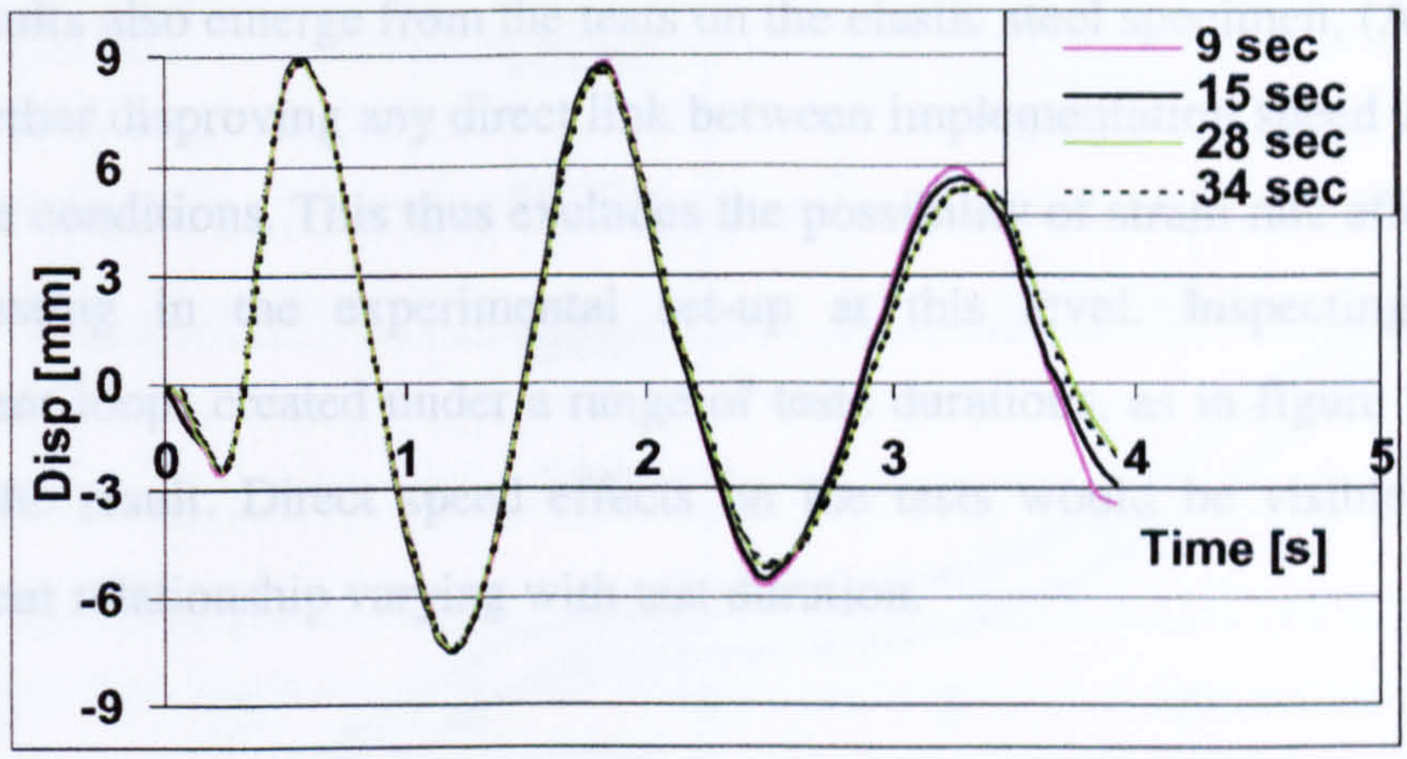


Fig. 7.16 Response obtained with fixed 0.08s step size and various test durations

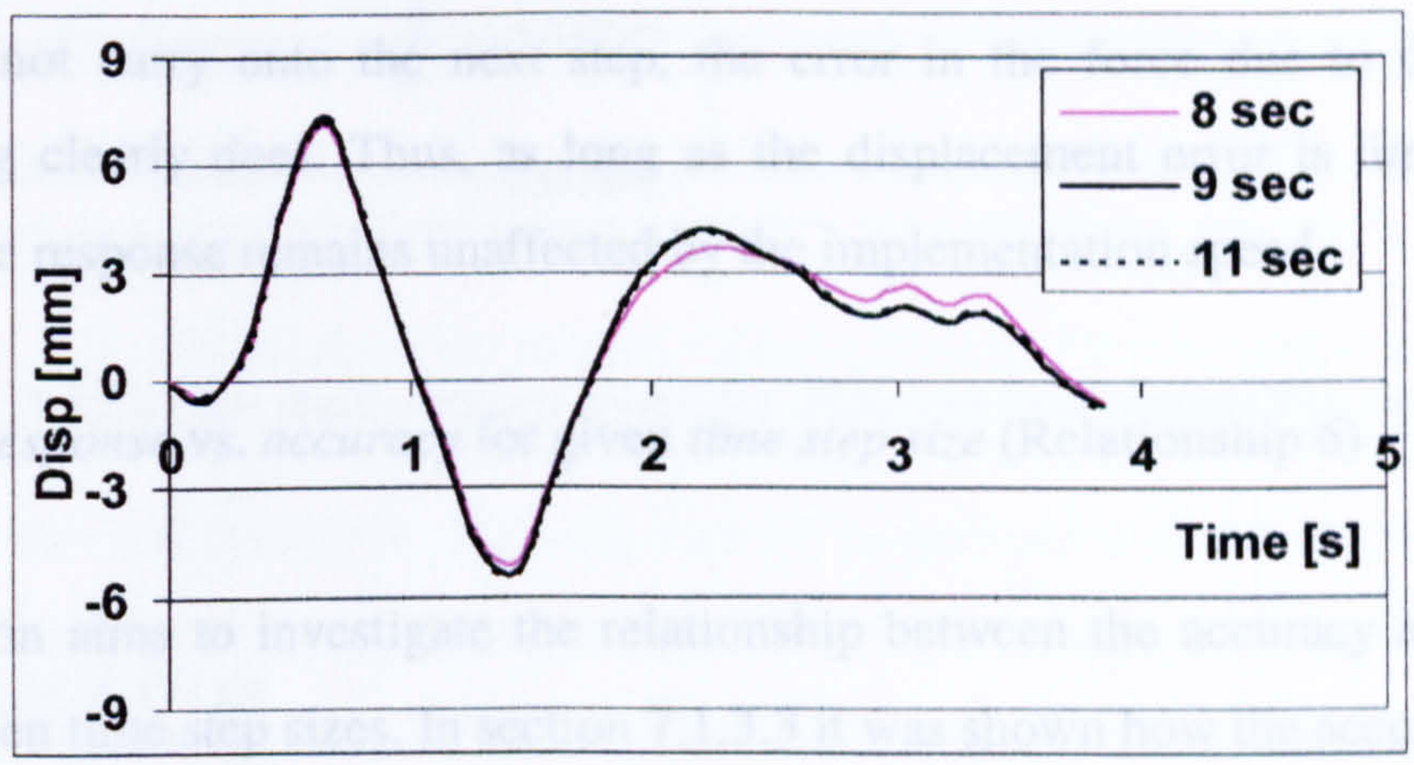


Fig. 7.17 Response obtained with fixed 0.16s step size and various test durations

7.1.3.9 Analysis: *speed - response* relationship

In figures 7.14 to 7.17, there appears in some cases to be a relationship where increased implementation speeds result in a less damped or negatively damped response. The effect is however not universal and emerges mostly during particularly fast implementation. Figure 7.15 and 7.16 clearly display how the reduced durations lead to growing amplitudes in the case of the 0.04 and 0.08s step sizes, while the same cannot be found in figure 7.14 and 7.17 using the 0.004 and 0.16s step sizes.

It appears that no general, direct relationship between implementation speed and response can be established. This can be seen for example in figures 7.14 and 7.17, where varying test durations do not result in systematic variation in system responses.

Similar results also emerge from the tests on the elastic steel specimen, (Algaard *et al.* 2001a), further disproving any direct link between implementation speed and response under these conditions. This thus excludes the possibility of strain rate effects and real inertia existing in the experimental set-up at this level. Inspecting the force-displacement loops created under a range of tests durations, as in figure 7.18, further confirms the result. Direct speed effects on the tests would be visible as a force-displacement relationship varying with test duration.

When in some instances the response appears to be affected by the implementation speed, as for example in the case of the 0.08s time step in figure 7.16, this is in fact due to the loss of control accuracy through the fast tests. Although the displacement errors do not carry onto the next step, the error in the force due to the incorrect positioning clearly does. Thus, as long as the displacement error is limited, say to 0.2mm, the response remains unaffected by the implementation speed.

7.1.3.10 *Response* vs. *accuracy* for given *time step size* (Relationship 6)

This section aims to investigate the relationship between the accuracy and response for the given time step sizes. In section 7.1.3.3 it was shown how the accuracy appears to be a function of implementation speed. The relationship between the response and accuracy is therefore likely to be essentially the same as the response vs. speed relationship presented in the previous section.

In order to positively determine whether the differences in the response are due to inaccuracy caused by increased implementation speeds, differential strain rates or real inertia effects, the force-displacement loops can be inspected. If strain rate or real inertia effects are present, these should affect the force-displacement relationship by displaying an apparent increased resistance with increasing implementation speeds. A force-displacement graph for the 0.08s test, displaying two extreme runs and an intermediate one, is presented in figure 7.18.

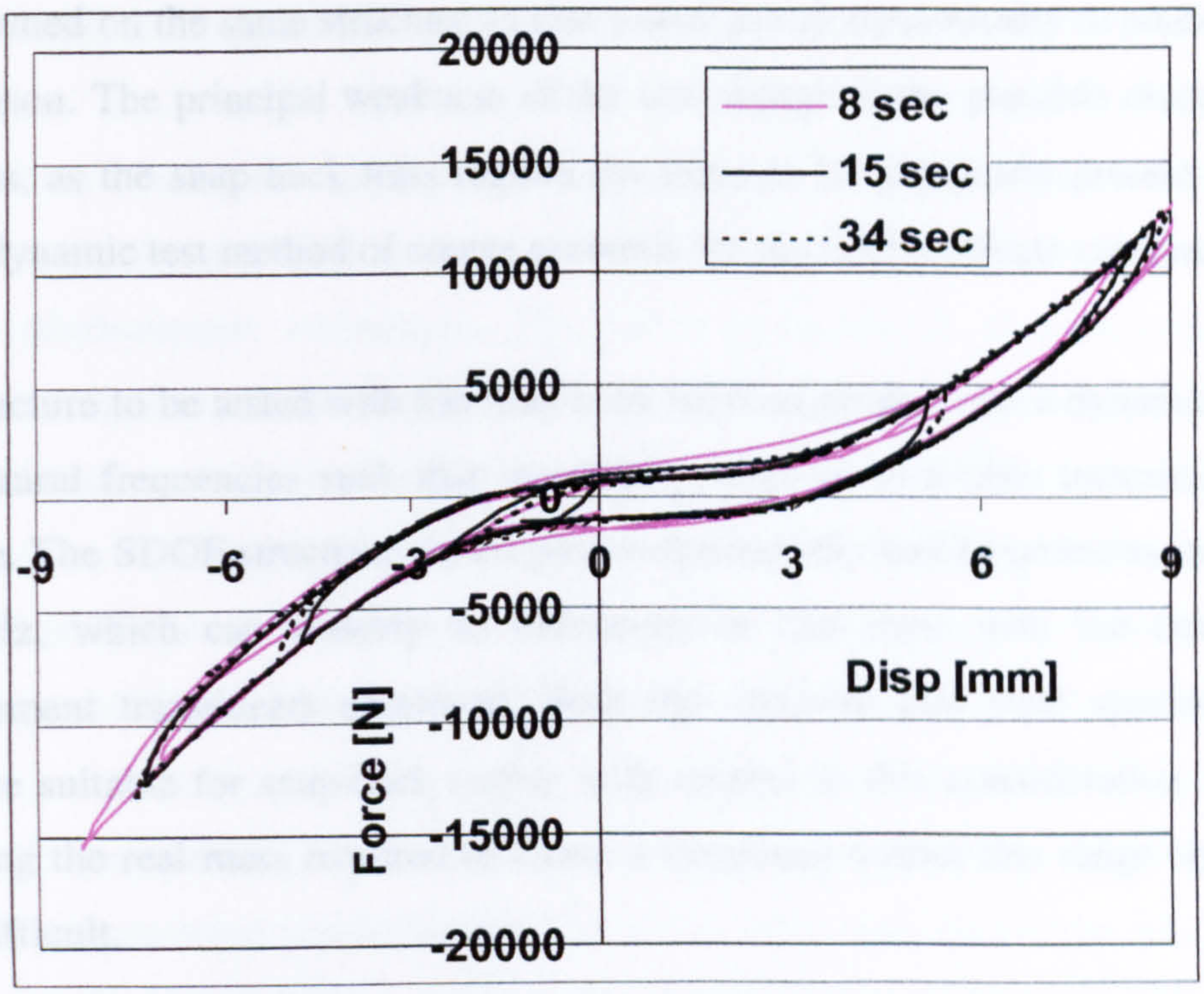


Fig. 7.18 Force displacement loops for different implementation speeds

7.2 VERIFICATION

This section is concerned with the verification of the complete implementation system developed for pseudodynamic testing and documented throughout this thesis. The verification consists of two main parts, the so-called "snap-back" test which obtains a genuine dynamic response and an evaluation based on a comparison with the numerically obtained data. The objective of both of these studies is to provide a reference solution with which the pseudodynamically obtained response may be assessed.

7.2.1 Snap-back test

A snap-back test is a test where a dynamic system is displaced to a given position and then released to vibrate freely to produce a response to the initial displacement. The method is commonly used in conjunction with pseudodynamic tests both for verification and calibration, as for example described by Negro (1997). Here, the test

is performed on the same structure as that tested pseudodynamically to enable a direct comparison. The principal weakness of the test though is the possible discrepancy in the mass, as the snap back tests require the mass to be physically present while the pseudodynamic test method of course accounts for the inertia effects computationally.

The structure to be tested with the snap-back method needs to be a dynamic structure with natural frequencies such that monitoring with the available instrumentation is possible. The SDOF structures tested pseudodynamically had frequencies in the range of 1-3Hz, which can feasibly be monitored in real time with the conventional displacement transducers employed. Both the concrete and steel specimens were therefore suitable for snap-back testing with respect to this consideration. However, providing the real mass required to create a frequency within this range could prove more difficult.

During the pseudodynamic tests, the virtual mass assigned to the concrete column was no less than 48600kg, which would be near impossible to reproduce physically. In the event that this could be done, the axial force in the column would introduce a range of new variables, rendering the comparison with the pseudodynamic tests less realistic. However, with the steel column, being significantly more flexible, the required virtual mass of around 250kg was sufficient to produce a frequency in the desired range of 1-3Hz. This mass could fairly easily be attached at the top of the structure, as described in section 3.3.3.2 without altering the properties of the column itself.

The snap-back test aimed to obtain the free vibration response of the steel column (with the 250kg mass on top) to an initial displacement of up to 50mm. The initial displacement was not imposed by the hydraulic actuator, as it would be difficult to instantaneously disconnect it from the specimen. Instead, the displacements were imposed by manually exciting the structure to oscillations of a 50mm magnitude and then allowing it to vibrate freely. At peak displacements, the velocity of course equals zero, so the fact that the structure was not stopped and released should not influence the results.

7.2.1.1 Results from snap-back test

As described above, the actual snap-back response was obtained by exciting the structure manually and tracing the time history of the displacement by means of an external displacement transducer. For direct comparison, a pseudodynamically generated response obtained under the same conditions was required. As the pseudodynamic test aimed only to provide as accurate a response as possible, a small time step size was employed. For completeness, two time integration schemes were used to reveal the existence of any algorithmic effects, in this case the central difference and Newmark Implicit – Integral Form methods.

The pseudodynamic response was obtained using the two schemes with a time step of 0.01s. This represented approximately 7% of the theoretical maximum time step size for the explicit central difference method. Such time steps should only produce small algorithmic effects. The virtual mass was set to 249kg based on preliminary experiments gauging the real frequency of the structure. No viscous damping was included, as the hysteretic damping in the specimen should be identical under both real dynamic and pseudodynamic oscillations.

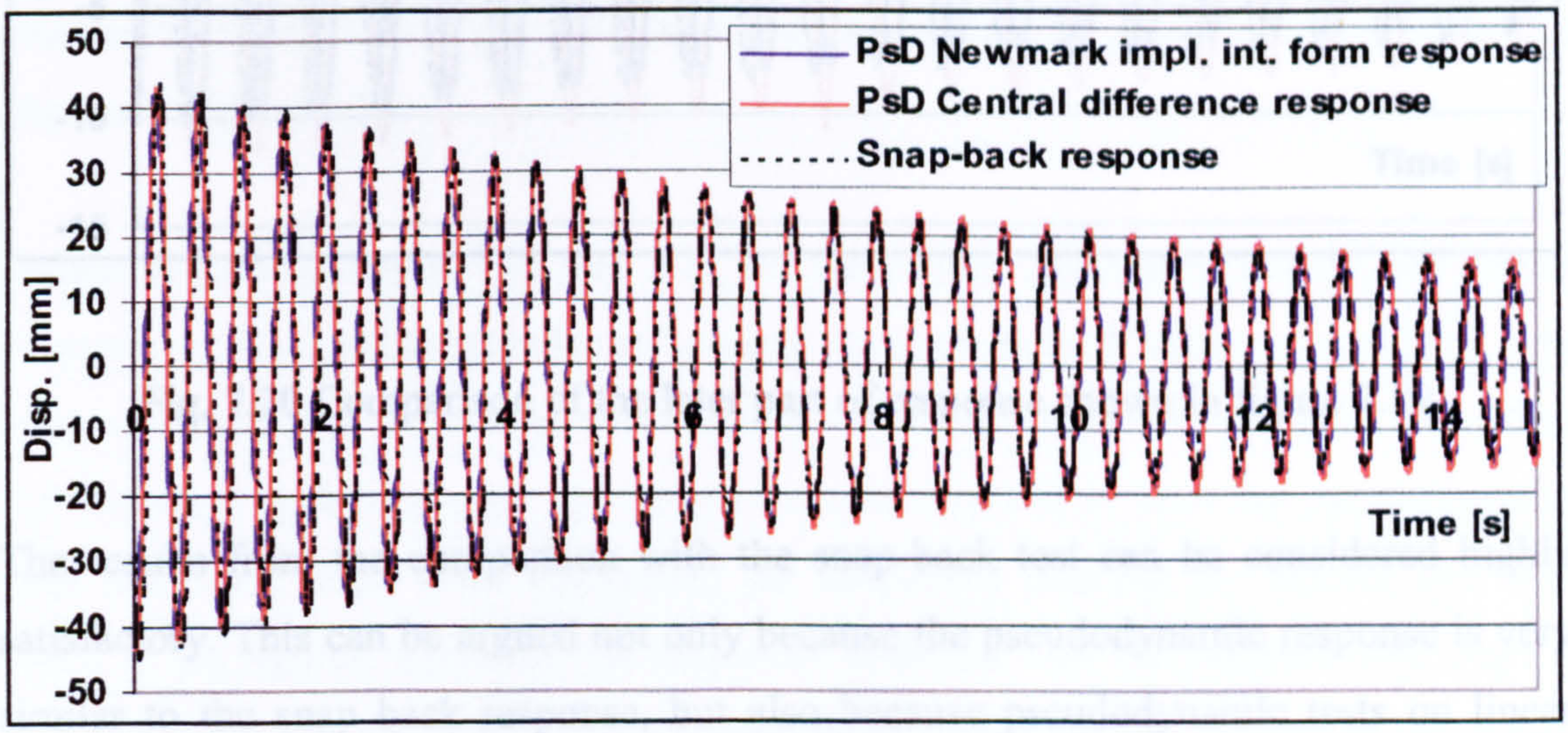


Fig. 7.19 Comparison between snap-back and pseudodynamic responses

Figure 7.19 shows the response generated during the snap-back and the two pseudodynamic tests. It is immediately apparent that the three system responses are

essentially the same, showing the first 15s of a very lightly damped response to a 45mm initial displacement. Some minor differences in the response can however be identified, especially when studying longer durations.

When studying figure 7.20, the previously identified algorithmic properties of the central difference and Newmark Implicit – Integral Form schemes can be seen. The central difference method creates a somewhat higher amplitude and higher frequency response than the integral form algorithm while the snap-back frequency appears to be higher than in both the pseudodynamic cases. The frequency difference between the snap-back and pseudodynamic responses can naturally be explained by an imperfectly chosen virtual mass. In terms of amplitude, the snap-back response falls somewhere in between the two pseudodynamic ones.

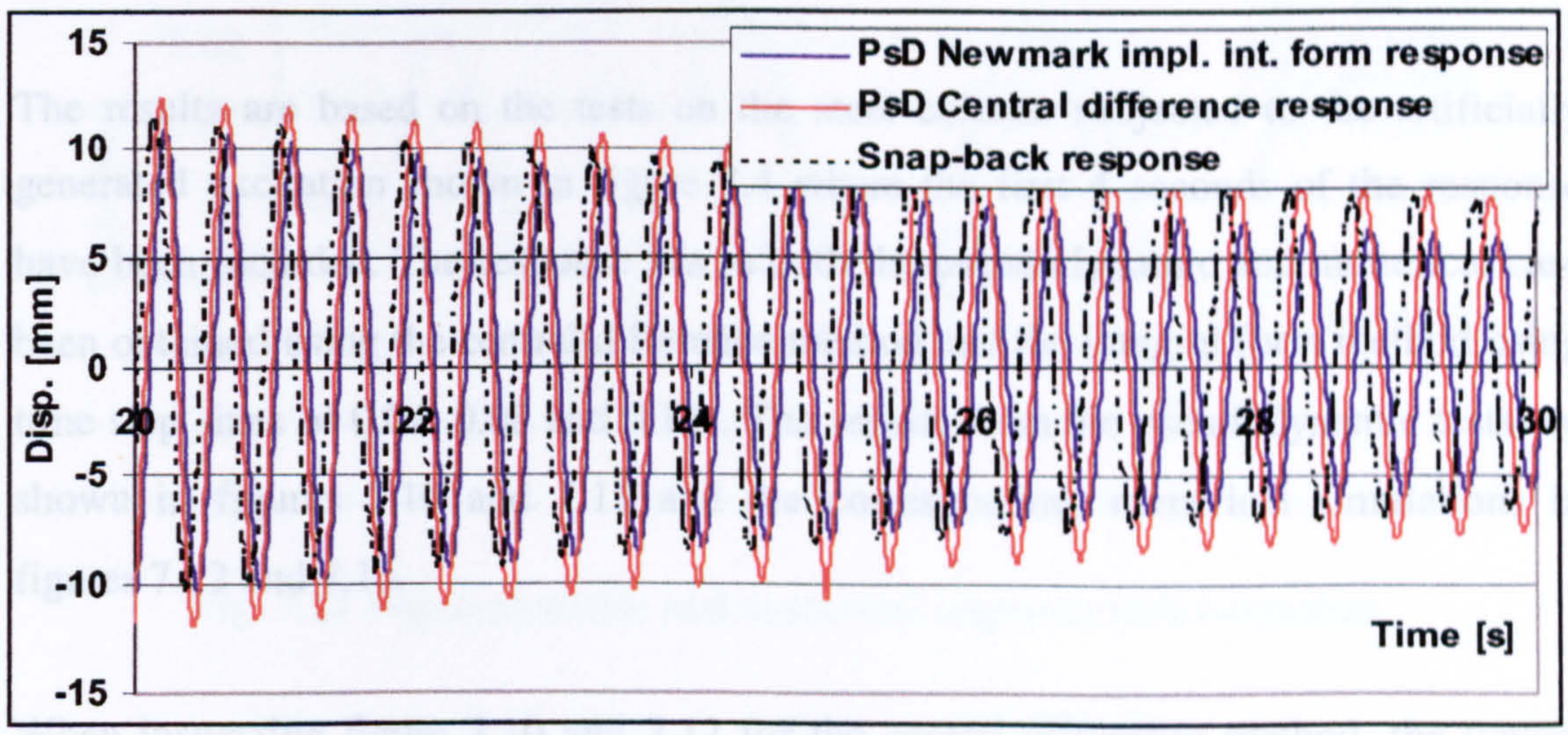


Fig. 7.20 Comparison of the later part of response shown in figure 7.19

The results from the comparison with the snap-back test can be considered highly satisfactory. This can be argued not only because the pseudodynamic response is very similar to the snap back response, but also because pseudodynamic tests on linear structures are considered particularly difficult as there is no stabilising structural damping.

7.2.2 Numerical verification study

As already indicated in section 7.1.3.6, some experiments were carried out both numerically and pseudodynamically. This was done in order to attempt to isolate the causes of some of the differences in the response identified in the sensitivity study. However, the comparison with the numerical data also provided good opportunity for verification of the implementation system.

The numerical simulations were all carried out on the steel structure as this was found to be approximately linear during the pseudodynamic tests. The numerical simulations would reproduce any algorithmic effects in the implementation, but not introduce any effects related to the control. The control component would not be present and all simulated measurements would be exact.

The results are based on the tests on the steel column subjected to the artificially generated excitation shown in figure 7.4 where the first 4 seconds of the response have been included. The response has in both the pseudodynamic and numerical case been obtained using the central difference method and the integral form method using time step sizes of 0.04, 0.08 and 0.16s. The results from the pseudodynamic tests are shown in figures 7.10 and 7.11 and the corresponding numerical simulations in figures 7.12 and 7.13.

When inspecting figure 7.10 and 7.12 for the central difference method, the typical features of the integration scheme become apparent in both cases – increasing time step size results in period shortening and a somewhat amplified response. The effect of the larger time step sizes appears to be near identical in both the pseudodynamic and numerical tests, when comparing the difference between the dashed and solid lines in the figures. However, the responses obtained with the smallest time steps in the two cases are not quite alike; the pseudodynamic response appears to be slightly damped and has a neutral position a small distance below the zero displacement line. These differences can be explained by the omission of any viscous damping in the numerical model and a small offset of the zero position in the pseudodynamic test.

Considering then figures 7.11 and 7.13, the responses obtained with the integral form time stepping scheme, the effects with the increased time step sizes again seem to be equal in both the pseudodynamic and numerical tests. The 0.16s time steps display period elongation of the same order as with the central difference method. The response obtained using the largest time steps may also appear somewhat reduced in amplitude, but this is only the case for the first few oscillations. The principal observation though is that the effects of the increasing time step sizes are of the same nature with both the pseudodynamic and numerical tests. The small differences in the response generated with small time step sizes can again be explained by the viscous damping and the offset of the zero line.

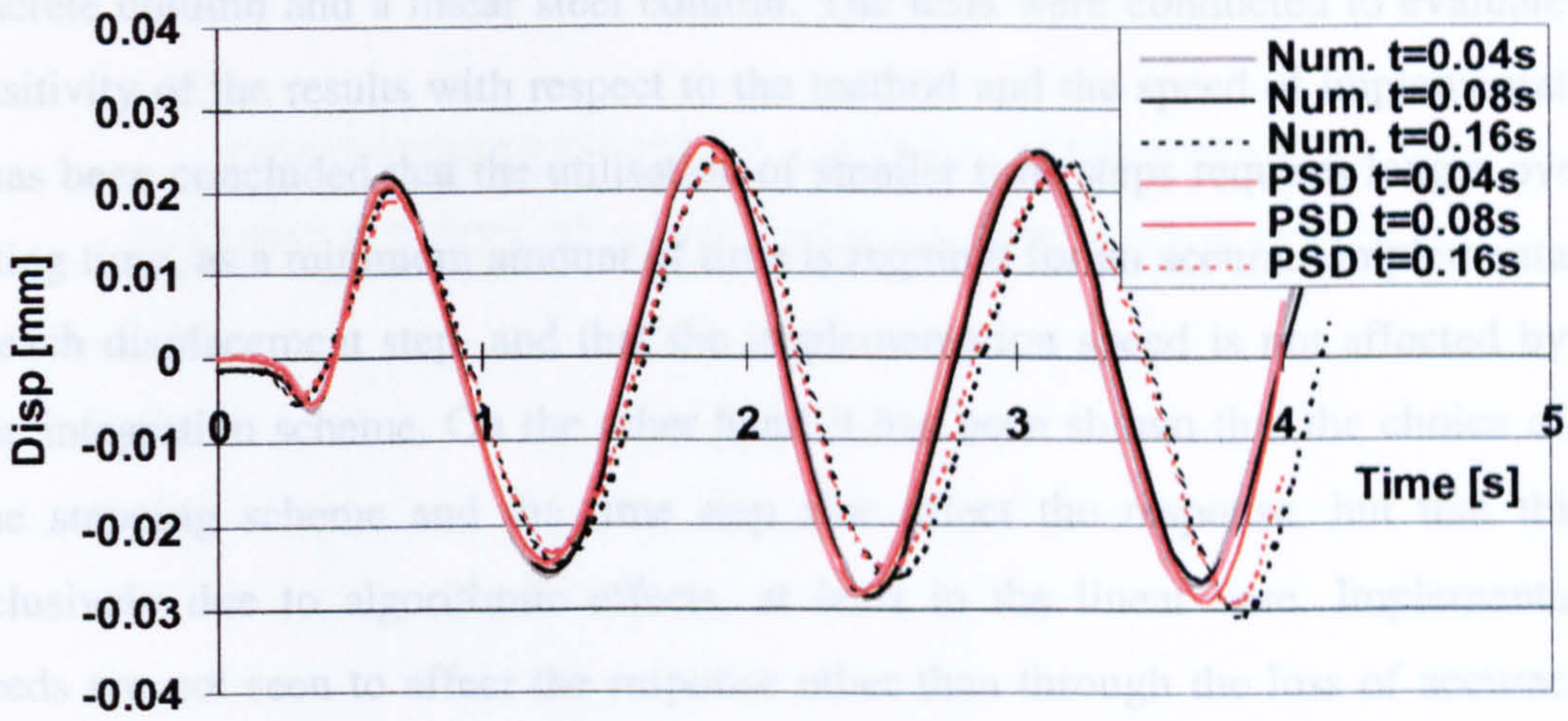


Fig. 7.21 Pseudodynamic and numerical response with correction

In order to confirm the cause of the small differences in the observed response, some further studies were carried out with the numerical tests. Variable levels of viscous damping were included and the effect of a shift in the neutral position investigated. It was found that by including approximately 1% of critical damping and an offset of the zero position by 1.5mm, the pseudodynamic and numerical response for all time step sizes became practically indistinguishable. This can be seen in figure 7.21, where the experimentally obtained data is red and the numerical results are in black, both using the integral form method.

From the comparison between the pseudodynamically and numerically obtained responses, it can be seen that only minor differences exist for the respective time step

sizes. It can thus be concluded that any differences that exist in the response obtained using different time step sizes are due to algorithmic effects only. Furthermore, the similarity in the responses obtained using the range of schemes and time step sizes for the pseudodynamic and numerical tests presents constructive support towards verifying the implementation system in general.

7.3 CONCLUSIONS

The performance of the pseudodynamic implementation system has been investigated by carrying out tests on two different dynamic systems: a highly non-linear reinforced concrete column and a linear steel column. The tests were conducted to evaluate the sensitivity of the results with respect to the method and the speed of implementation. It has been concluded that the utilisation of smaller time steps requires longer overall testing time, as a minimum amount of time is required for an accurate implementation of each displacement step, and that the implementation speed is not affected by the time integration scheme. On the other hand it has been shown that the choice of the time stepping scheme and the time step size affect the response, but that this is exclusively due to algorithmic effects, at least in the linear case. Implementation speeds are not seen to affect the response other than through the loss of accuracy in the fastest test.

Chapter VIII:

EXPERIMENTAL APPLICATION: REPEATED EARTHQUAKE LOADING

This chapter is concerned with the application of the pseudodynamic implementation system documented throughout this thesis. The implementation system is used to carry out an experimental study on the structural effects of repeated exposure to earthquake excitations. The study aims to evaluate the ability of structures to withstand further excitation after initial damage by tracing the changes in structural capacity and earthquake demands. The tests consider the changes in elastic and dissipative properties of the structure to investigate the performance and include extensive monitoring of energy changes.

8.1 BACKGROUND

In contrast to the particularly detailed and exact nature of pseudodynamic testing, seismic evaluation of structures is often carried out using simplified linear and nonlinear analysis procedures. In the linear case, *elastic response spectrum analyses* can provide an accurate picture of the maximum displacements, velocities and accelerations resulting from a particular excitation. However, as most structures are expected to undergo considerable inelastic deformation under design level ground motion acceleration, the application is sometimes limited.

An alternative method that takes account of the stiffness loss and inelastic deformation is the *pushover* analysis (ATC 1996). The results from these analyses can be combined with the elastic response spectrum of the acceleration to represent an approximate nonlinear analysis. In this case, the greatest uncertainty lies in the method of combining the elastic response spectrum and the inelastic pushover properties. This is of course due to the fact that the elastic response spectrum can only be obtained for linear elastic structures, which is not necessarily appropriate for the structures concerned. To account for the increased dissipative abilities of the structure, a reduced capacity spectrum is often used (ATC 1996) or, alternatively, an elastic spectrum with an increased level of viscous damping.

In the context of the simplified nonlinear analytical procedures, the terms *earthquake demand* and *structural capacity* are commonly used. These terms relate to the elastic response spectrum and pushover results, respectively, and will be explained below.

8.1.1 Earthquake demand

The earthquake demand is often defined in terms of displacement and is a function of both the ground motion and the structure concerned. With a ground motion accelerogram, a *demand spectrum* can be produced. This indicates the maximum amplitude for structures or structural modes with any vibration frequencies resulting from exposure to that particular accelerogram.

The demand spectrum is in theory produced by considering a number of linear elastic SDOF structures covering the entire applicable frequency range. Each of these structures are in turn exposed to the accelerogram and the response computed. From the response, the maximum occurring displacement can be found. This process is repeated for a range of structural frequencies until a continuous curve of the maximum displacements as a function of the frequency or period can be drawn. Such a curve will typically look like the one in figure 8.1 below. Similarly, the maximum occurring velocities and accelerations can be found the same way to obtain the spectral velocity and acceleration, yielding similar curves.

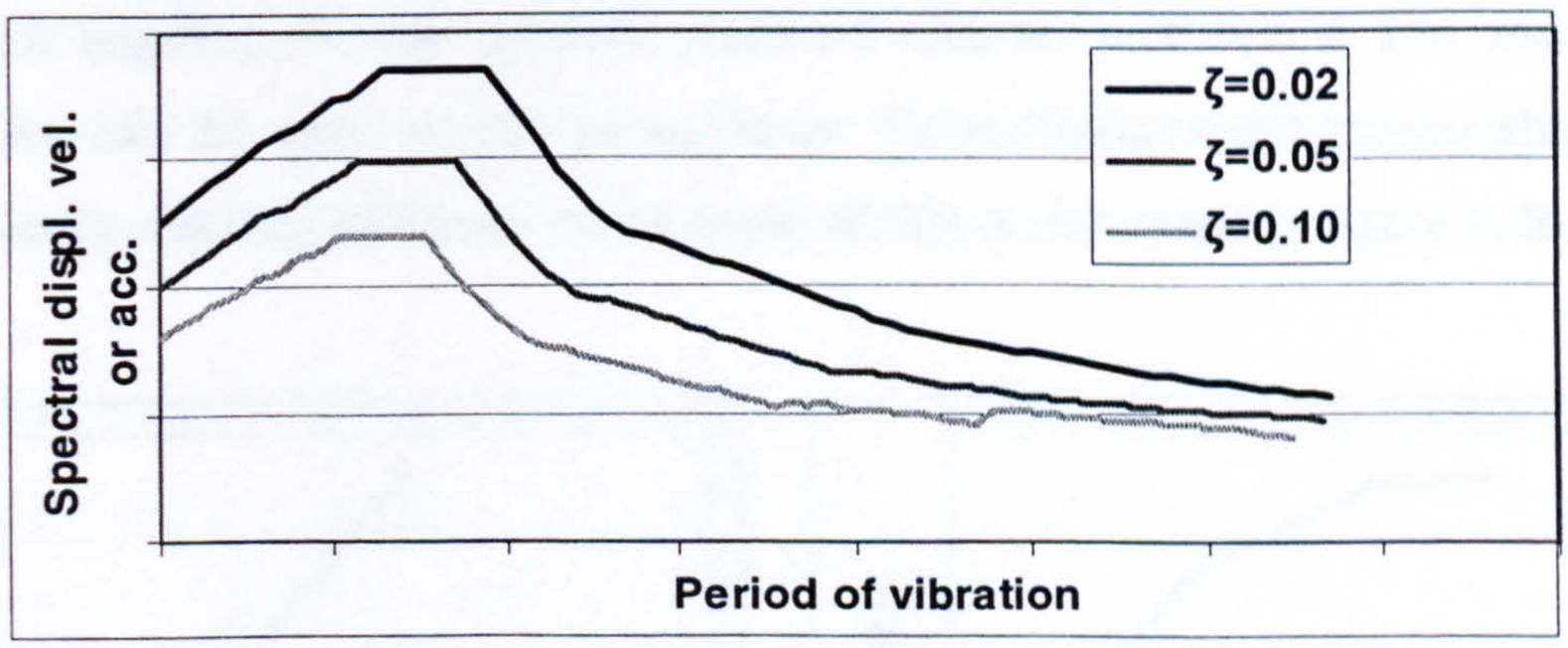


Fig. 8.1 Spectral displacement, velocity or acceleration as function of vibration period

The actual earthquake demand depends on the demand spectrum as well as the properties of the structure concerned. The curves shown above apply to structures with three levels of viscous damping. When the damping is higher, the curve will be lower indicating smaller displacement or accelerations. Additionally, for MDOF structures the earthquake demand for a certain mode shape of the structure will only

represent a single point on the curve. Furthermore, the situation is complicated by the fact that during damage, the fundamental frequency of the structure will shift and a different point on the curve should apply.

8.1.2 Structural capacity

The capacity of a structure refers to its ability to resist loads and the displacements resulting from these. Effectively, the structural capacity describes the force-displacement relationship in a structure. It is thus a function of the strength, stiffness and deformation characteristics of a structure.

Structural capacity is often described in terms of *roof displacements* and *base shear*. This gives an indication to the structure's overall ability to withstand lateral loads. For a linear elastic structure, the relationship will yield a linear function as shown in figure 8.2a. However, more realistically, structures will lose stiffness as damage accumulates and displacements increase. This can be idealised as happening in steps as plastic hinges develop within the structure (an example of a simple nonlinear model). To obtain the capacity curve beyond the elastic limit, a form of nonlinear analysis is required. For this purpose, pushover tests are well suited. The results from such tests can be seen as piecewise linear force-displacement relationships with continuously reducing stiffness. An example of this is illustrated in figure 8.2b.

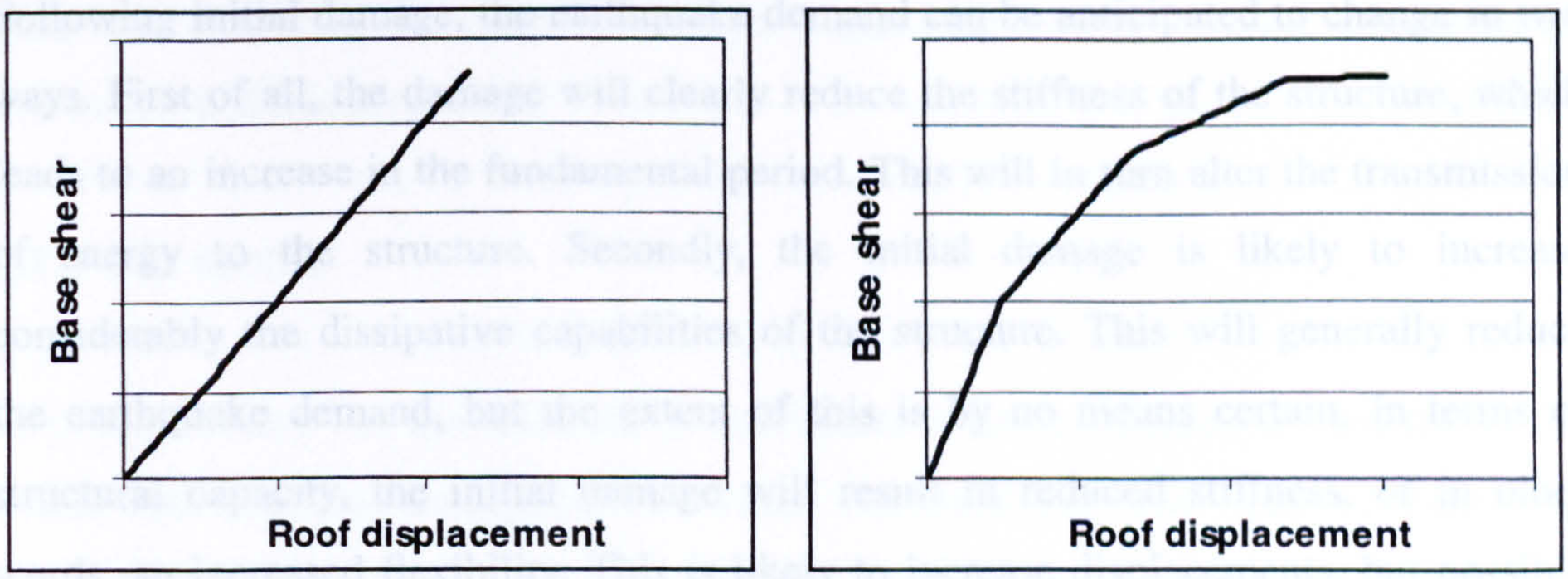


Fig. 8.2a & Fig 8.2b Linear and nonlinear (pushover) capacity curves, respectively

8.1.3 Pseudodynamic context

While it is possible to compute the earthquake demand for a linear elastic structure and approximate this for a nonlinear structure, some amount of uncertainty will always be introduced when applying this to a real structure. Likewise, the capacity curve of a structure only goes a limited distance in describing the structural behaviour as damage takes place. Additionally, the approximation of the capacity curve is often fairly crude.

The main objective in this chapter is to evaluate the effect of repeated earthquake excitation of structures. Following initial damage, both the earthquake demand and structural capacity will change. However, it is not straightforward to evaluate the structures' ability to handle another excitation, as the combined effect of the changes in not obvious. This can be seen in the context of retrofitting, which again may change both the demand and capacity in various ways.

In order to accurately assess the ability of withstanding repeated seismic excitations, a pseudodynamic test programme has been formed. A pseudodynamic test will be able to determine the exact amount of energy transmitted to a structure as well as demonstrating the structure's detailed response to this. It does not require any approximation technique to determine the earthquake demand.

Following initial damage, the earthquake demand can be anticipated to change in two ways. First of all, the damage will clearly reduce the stiffness of the structure, which leads to an increase in the fundamental period. This will in turn alter the transmission of energy to the structure. Secondly, the initial damage is likely to increase considerably the dissipative capabilities of the structure. This will generally reduce the earthquake demand, but the extent of this is by no means certain. In terms of structural capacity, the initial damage will result in reduced stiffness, or in other words, an increased flexibility. This is likely to increase displacements, but possibly reduce the maximum accelerations taking place.

Depending on the changes within both the earthquake demands and structural capacities following an initial exposure to a ground motion, subsequent exposure may

have more or less detrimental effects. One can envisage that repeated exposure leads to further and further damage taking place through an increased susceptibility to the ground acceleration. On the other hand, a possible scenario is that the increased vibration period and the increased dissipative capabilities of the structure following the initial damage will attract less seismic energy and overall provide an improved resistance to the excitation.

By carrying out a pseudodynamic test programme investigating the effects of repeated earthquake excitation, one will assess the structures ability to resist such exposure. Additionally, one will gain valuable information relevant to the issue of retrofitting damaged structures. This will extend to suggest whether increased stiffness or increased strength will assist in withstanding further excitations, or if the softer, more dissipative structure is already superior in coping with any repeated exposure to ground motion.

8.2 TEST PROGRAMME

The objective of the test programme was to systematically investigate the effects of repeated ground motion accelerations on a simple reinforced concrete structural component. It aimed not only to evaluate the ability to resist such repeated exposure, but to examine the specific changes in earthquake demand and structural capacity. This was done in the pseudodynamic framework, which should enable exact evaluation of these properties.

8.2.1 Test facility

In order to carry out the tests, the pseudodynamic testing facility developed by the author and described throughout this thesis was employed. The facility consisted of the SDOF experimental set-up described in Chapter III. Comprising a single double-acting 50kN capacity actuator with a ±50mm stroke with an internal LVDT and a specially designed load-cell. The actuator was controlled by a remote servo-valve, in turn controlled by the software controller described in Chapter V. The implementation system incorporating the controller was created within the LabView environment and

enabled semi-continuous and reasonably fast and accurate execution of the tests. This system is described in detail in Chapter VI.

In terms of time integration, the novel Newmark Implicit – Integral Form method, discussed in Chapter IV, was employed. This was combined with a fairly small time step size to ensure minimal introduction of algorithmic errors. The implementation system and time stepping scheme had been thoroughly validated prior to any testing, and a discussion on this can be found in Chapter VII.

The pseudodynamic execution system described in Chapter VI was specially adapted for the repeated tests and evaluation of the demands and capacities. These adaptations included introduction of a system for monitoring the energy balance during the tests. The external, kinetic and strain energies were evaluated at each time step and continuously integrated to determine the dissipated energy. This energy monitoring system was coded for use with both the Newmark Implicit – Integral Form algorithm and the classic central difference method.

8.2.2 Specimen structure

During the development of the pseudodynamic implementation system, two main specimen types were tested. These were the reinforced concrete and steel columns, described in section 3.3.3. Essentially, the short reinforced concrete column was expected to undergo significant inelastic deformation and accumulate substantial damage under the displacement stroke of the actuator. The long and slender steel column on the other hand, behaved in an almost perfect linear manner.

The nature of the repeated earthquake exposure is such that the initial damage from the first excitation will influence the transmission of forces and structural behaviour during further excitation. It is therefore essential that damage actually occurs and that the structural properties change. This kind of behaviour would not be expected from the steel column but had already been observed during sensitivity and verification tests carried out on the reinforced concrete column. As this specimen was also specifically designed for the capacity of the experimental facility, it was elected for the repeated earthquake excitation tests as well.

The concrete column specimen is described in detail in section 3.3.3.1, but will be described briefly here. Essentially, it consists of a 100x200mm section reinforced by 4x12mm high yield reinforcing bars. During testing, the column is strained in the strong direction. The column is only 600mm long and built into a rigid base. This ensures that critical damage takes place within the capacity and stroke of the actuator.

The structure was in all cases provided with a mass at the top of the column to yield an inverted pendulum system in dynamic terms. In pseudodynamics, the mass is of course only virtual as all inertia is accounted for numerically. This implies that the mass can readily be changed and assigned any value. Generally speaking, the mass is defined such that the fundamental natural frequency of the system represents a realistic structure. For a regular reinforced concrete frame structure, this could be of the order of 1-3Hz. Verification and sensitivity test carried out at an earlier stage (Chapter VII) suggested an initial stiffness of the reinforced concrete specimen of around 3000kN/m. The required mass to produce a frequency of 2Hz would therefore be around 20t.

During the repeated tests described in this chapter, it was desirable to identify changes in the earthquake loading (demand) as a result of the change in fundamental frequency of the structure. At this stage, two scenarios can be envisaged. Either the frequency of the structure is initially higher than the prominent earthquake excitation frequency and through damage will be reduced to approach that of the acceleration, or the initial frequency of the structure is already lower than the prevailing acceleration and will only shift further away. The effect of the two alternative situations will of course be opposite, and both are worthwhile investigating.

In order to enable the possibility of the two situations described occurring, two systems with different frequencies have been tested. In one case, a frequency of approximately 4Hz was aimed for, to allow for the possibility of a frequency decrease to approach the predominant seismic excitation frequency. In the other case, a frequency of around 2Hz was desired so that a decreasing frequency will increase the difference from the excitation frequency. Both these values were based on a prevailing frequency in the ground motion of 3Hz.

8.2.3 Excitation

During the tests with repeated exposure to seismic excitation, the objective is to identify changes in both earthquake demand and structural capacity from one exposure to the next as well as evaluating the overall ability to withstand the repeated loading. This makes certain ground motion acceleration histories more suitable than others and a few factors have to be considered before selecting an accelerogram.

Although it is essential to induce the damage, it is also important that the specimen is not critically damaged during the first excitation, as this would naturally extinguish the prospect of further tests on the same specimen. On the other hand, it has to be ensured that the ground motion causes some damage in the specimen, or the test could be repeated an infinite number of times without any development. The scaling of the accelerogram is thus crucial, but also the frequency content and duration have to be considered. If the frequency content is very wide ranging, with no predominant excitation frequency in the vicinity of the natural frequency of the specimen, it can be difficult to expect a significant change in the energy transmitted to the specimen through the period elongation taking place within the specimen structure. The duration may have to be considered in the light that a long excitation could inflict all potential damage on a specimen in a single exposure.

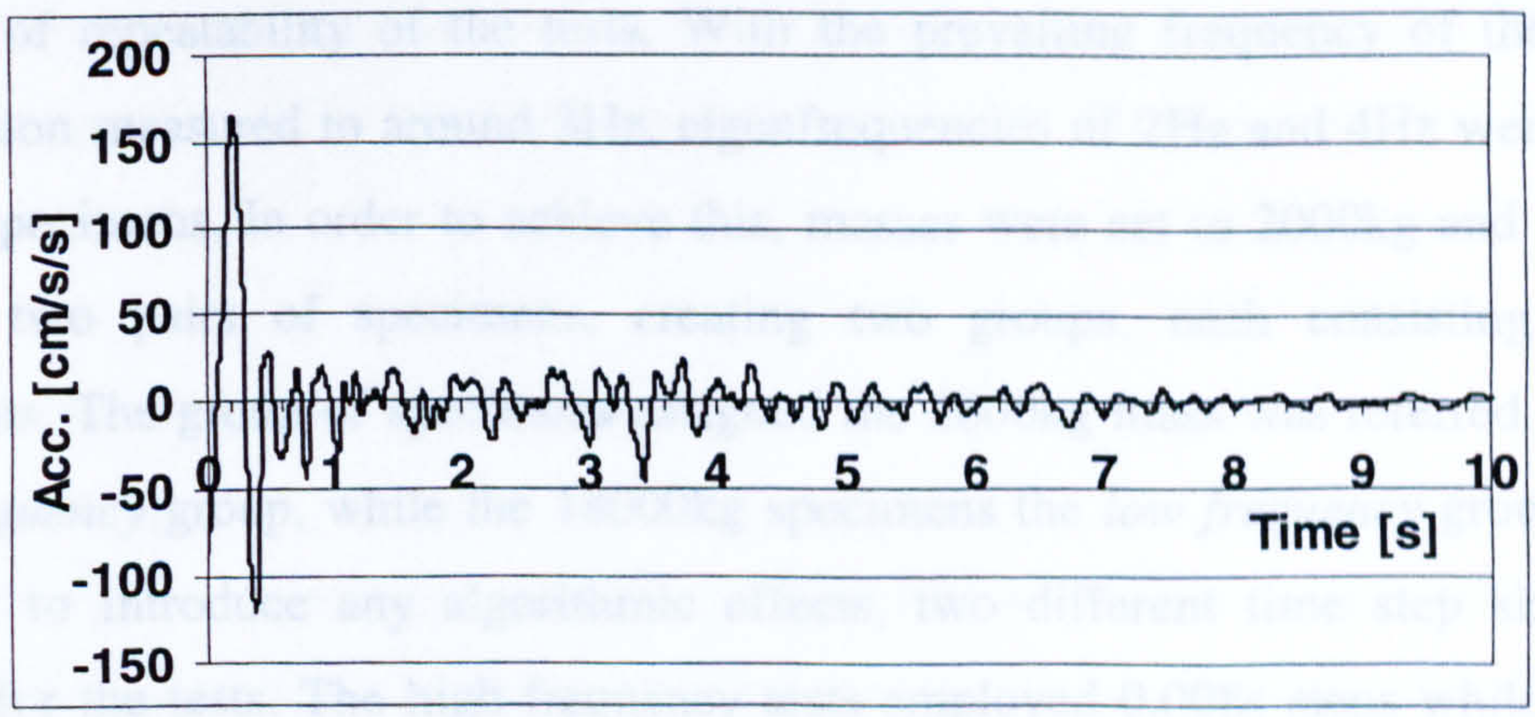


Fig. 8.3 The N-S 1957 Port Hueneme accelerogram

The North-South component of the 1957 Port Hueneme excitation was considered to satisfy the above criteria fairly well. As it principally consists of a single excitation

cycle, it is certainly not of a too long duration to display progressive damage from exposure to exposure. The predominant excitation frequency can be reasonably well defined as being around 3Hz, which enables the possibility of changes in the energy transfer to structures with frequencies near this. The accelerogram can be seen in figure 8.3.

8.2.4 Test programme

In this investigation of the effects of repeated exposure to earthquake excitation, two main scenarios were sought: A general *increase* in the susceptibility of the structure to the excitation through a *reduction* in the difference between the eigenfrequency of the structure and the prevailing frequency of the excitation through damage, or a general *decrease* in the susceptibility through an *increase* in the difference between the two frequencies through damage. The two situations could be created by either keeping the structural properties constant, i.e. constant eigenfrequency, but altering the time scale of the accelerogram, or by keeping the frequency content of the accelerogram constant, but altering the structural properties. The second option was opted for, hence assigning different masses allowed for setting the frequencies to the desired values.

Four reinforced concrete specimens were created, all to the same specification detailed in section 3.3.3.1. The specimens were assigned masses in pairs, to include an element of repeatability of the tests. With the prevailing frequency of the ground acceleration measured to around 3Hz, eigenfrequencies of 2Hz and 4Hz were sought for the specimens. In order to achieve this, masses were set to 2000kg and 18000kg for the two pairs of specimens, creating two groups, each consisting of two specimens. The group of specimens assigned the 2000kg mass was referred to as the *high frequency* group, while the 18000kg specimens the *low frequency* group. To be sure not to introduce any algorithmic effects, two different time step sizes were utilised for the tests. The high frequency tests employed 0.008s steps while the low frequency group 0.04s steps. This allowed approximately the same number of steps per oscillation. In both cases 500 time steps were computed, providing 20s and 4s testing time for the low and high frequency systems, respectively.

The scaling of the ground motion excitation required to focus on the desired effects was not straightforward. Too strong an excitation could inflict critical damage at the first exposure, whilst too weak excitations could result in little or no change in the structural characteristics. This could indicate that weaker, rather than stronger excitations should be employed at the start. On the other hand, it was desired that the initial exposure represented a substantial earthquake acting on an undamaged structure, so smaller excitation amplitudes prior to the actual event should be avoided. In order to attempt to correctly scale the initial excitation, numerical simulations were carried out on an equivalent linear system. The simulations were however not expected to provide an accurate representation of the expected highly non-linear behaviour of the reinforced concrete structure. The uncertainties introduced by this and other potential experimental difficulties indicated that high-quality results should not necessarily be expected from the first tests in each specimen group. While the initial tests in each group were still expected to display a general trend consistent with the later tests, the first tests might or might not allow for the systematic testing desired.

8.3 RESULTS

This section presents the results from the four reinforced concrete specimens subjected to repeated exposures of the Port Hueneme ground acceleration. The results have been presented for each structure category, i.e. the high and low frequency groups.

8.3.1 The low frequency structures

The low frequency specimens are referred to as *LF1* and *LF2* throughout this section. Preliminary numerical simulations indicated that a scale factor of unity for the ground motion should result in suitable initial amplitudes, so it was decided to scale the peak acceleration to 1.67m/s^2. However, prior to this, non-damaging elastic tests were carried out to determine the undamaged properties. This was successfully done for both specimens by obtaining the response from 3.6-6.0mm initial displacements for LF2 and LF1, respectively. Typical force-displacement loops and response plots can

be seen in figures 8.4 and 8.5, respectively, both showing the results from LF2. Figure 8.4 indicates an approximate initial stiffness of 3.6kN/mm, and figure 8.5 a fundamental frequency of approximately 1.9Hz.

During the full-scale low frequency tests, the specimens were subjected to a Port Hueneme accelerogram, first with a scale factor of unity, then a scale factor of 1.5. This was the case for both LF1 and LF2, but only the results for LF2 are presented, as the testing of this was more successful. During the testing, the structure was exposed to the ground motion in alternating directions. The ground motion with a scale factor of +1 is referred to as *positive*, while with the scale factor of –1 as *negative*.

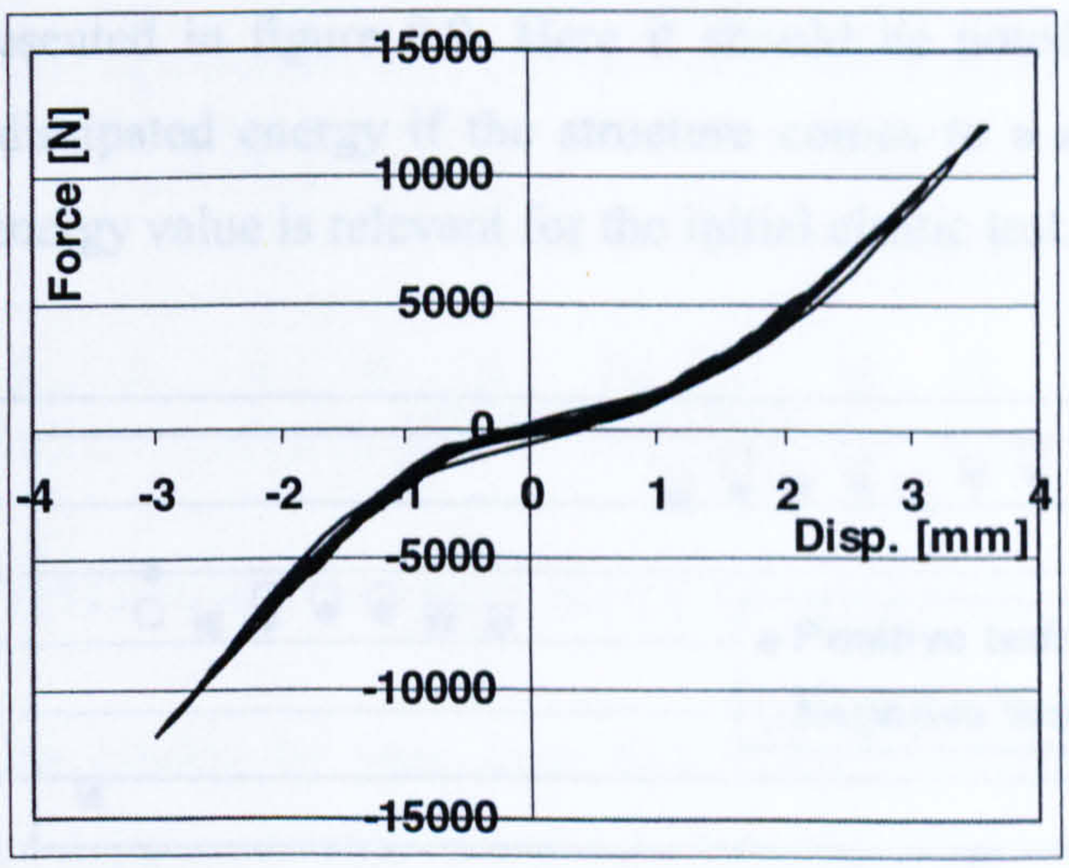


Fig. 8.4 Initial force-displacement loops for LF2

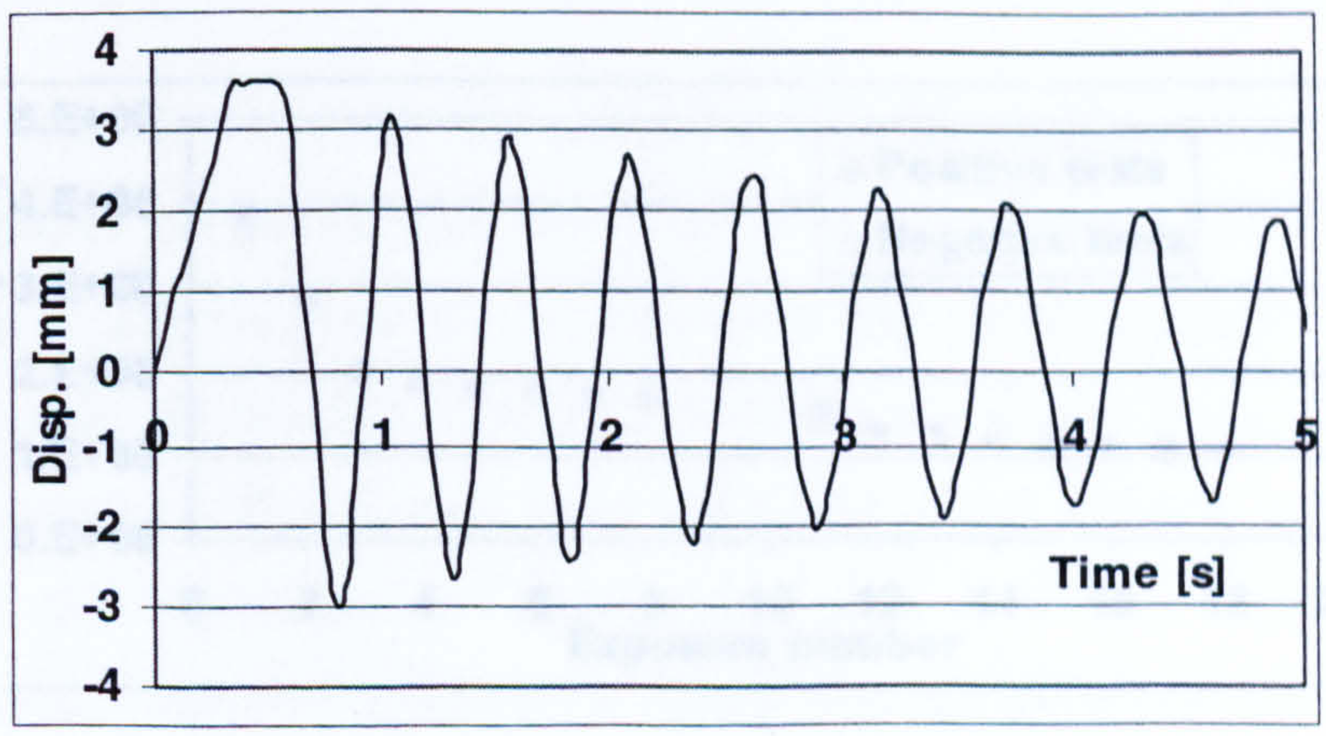


Fig. 8.5 Elastic response to a 3.6mm displacement

As the tests aim to investigate the overall behaviour and the change in earthquake demand and structural capacity, a selection of results are presented below. First, in figure 8.6, the peak amplitudes are displayed as a function of exposure number. Here, it should be noted that the first exposure represents an elastic test, exposures 2-8 the exposures to the unit scale factor accelerogram and exposures 11-18 the exposures to the 1.5 scale factor accelerogram. The exposure to the unit accelerogram is referred to as *Stage 1*, while to the 1.5 scale factor accelerogram *Stage 2*. In figures 8.7 and 8.8 the secant stiffness and resulting frequency are presented. The secant frequency is a measure of the approximate natural frequency of the structure for the maximum amplitudes encountered during that exposure based on the secant stiffness. The exposure numbers correspond to those in figure 8.6. The attracted energy for number of exposures is presented in figure 8.9. Here it should be noted that the attracted energy equals the dissipated energy if the structure comes to a rest within the test period and that no energy value is relevant for the initial elastic test.

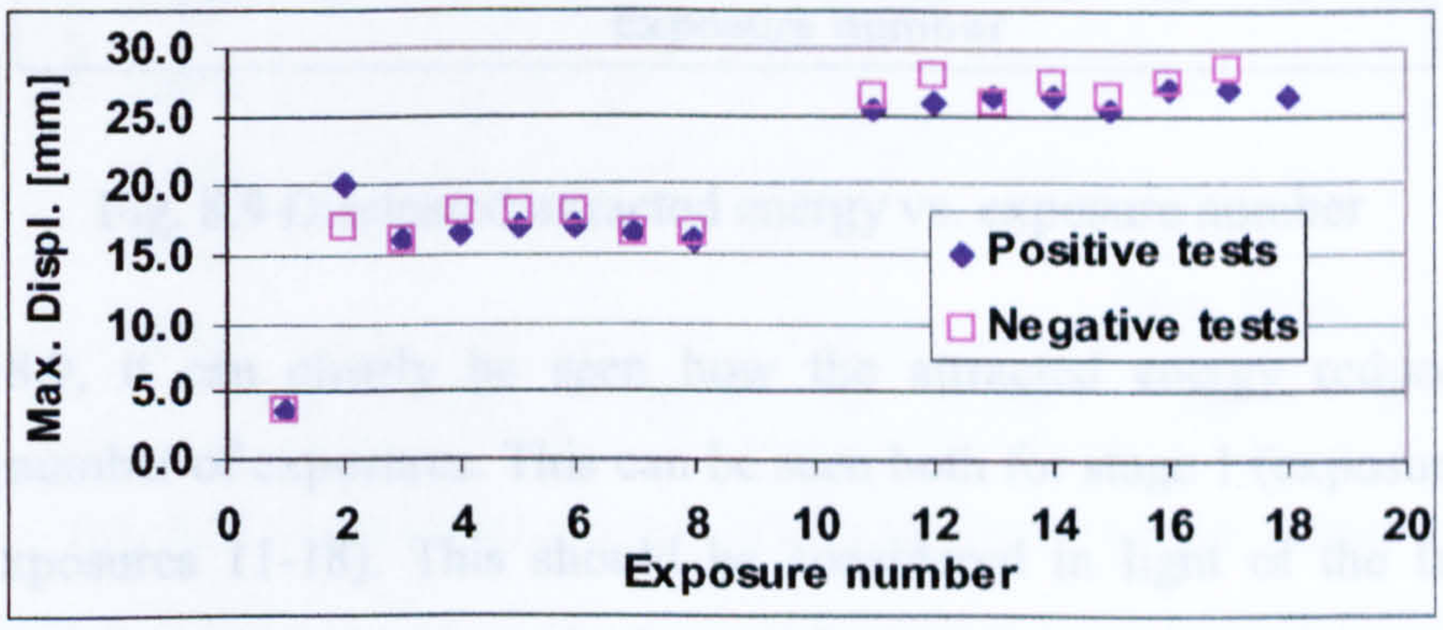


Fig. 8.6 Peak amplitude vs. exposure number

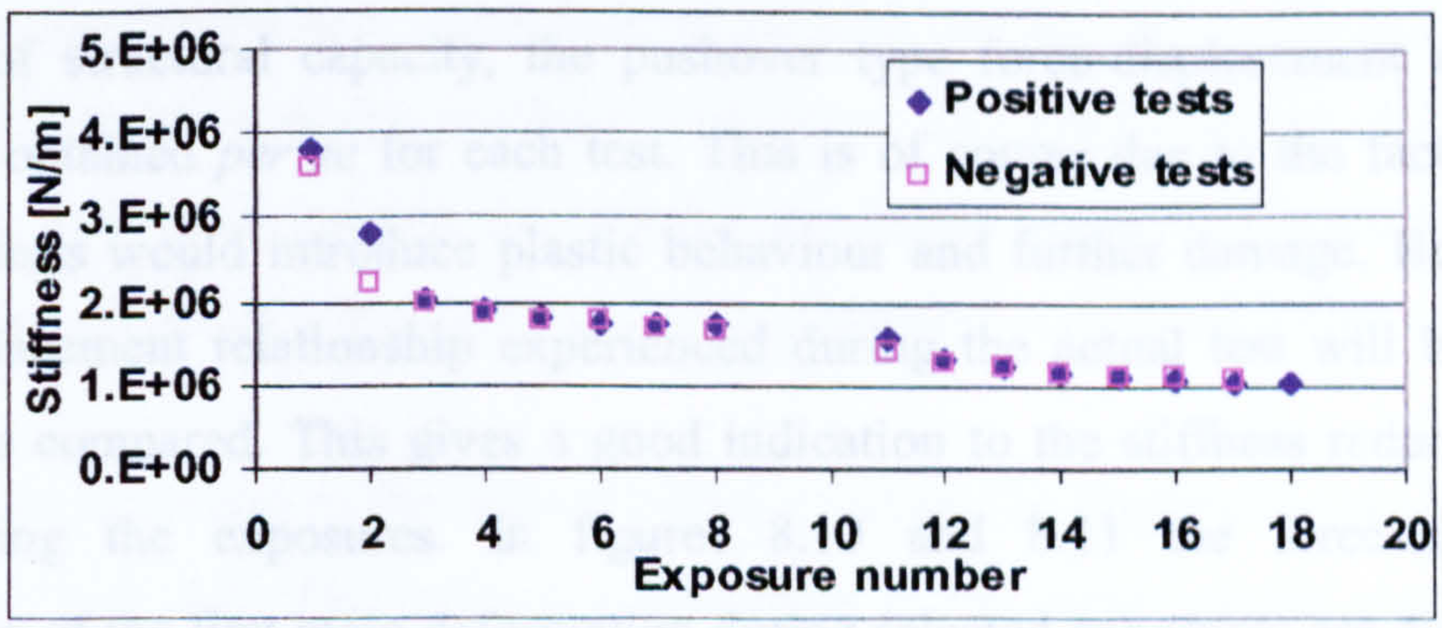


Fig. 8.7 Stiffness vs. exposure number

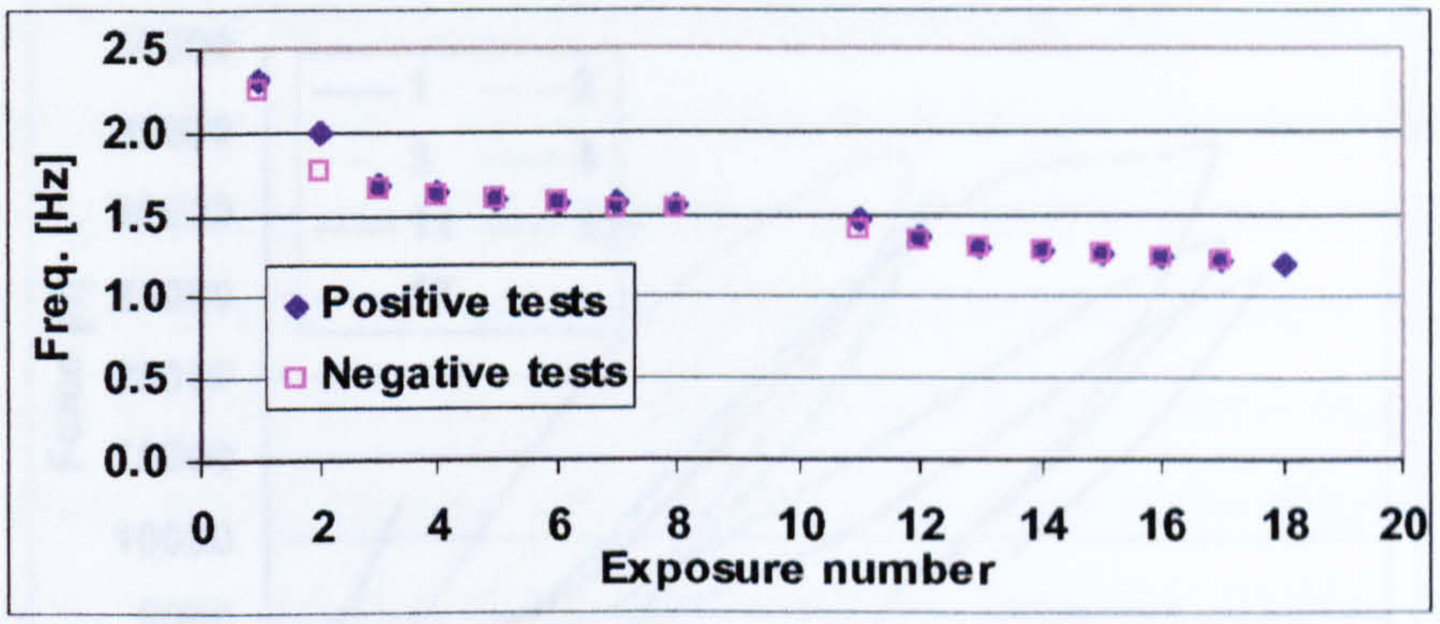


Fig. 8.8 Frequency vs. exposure number

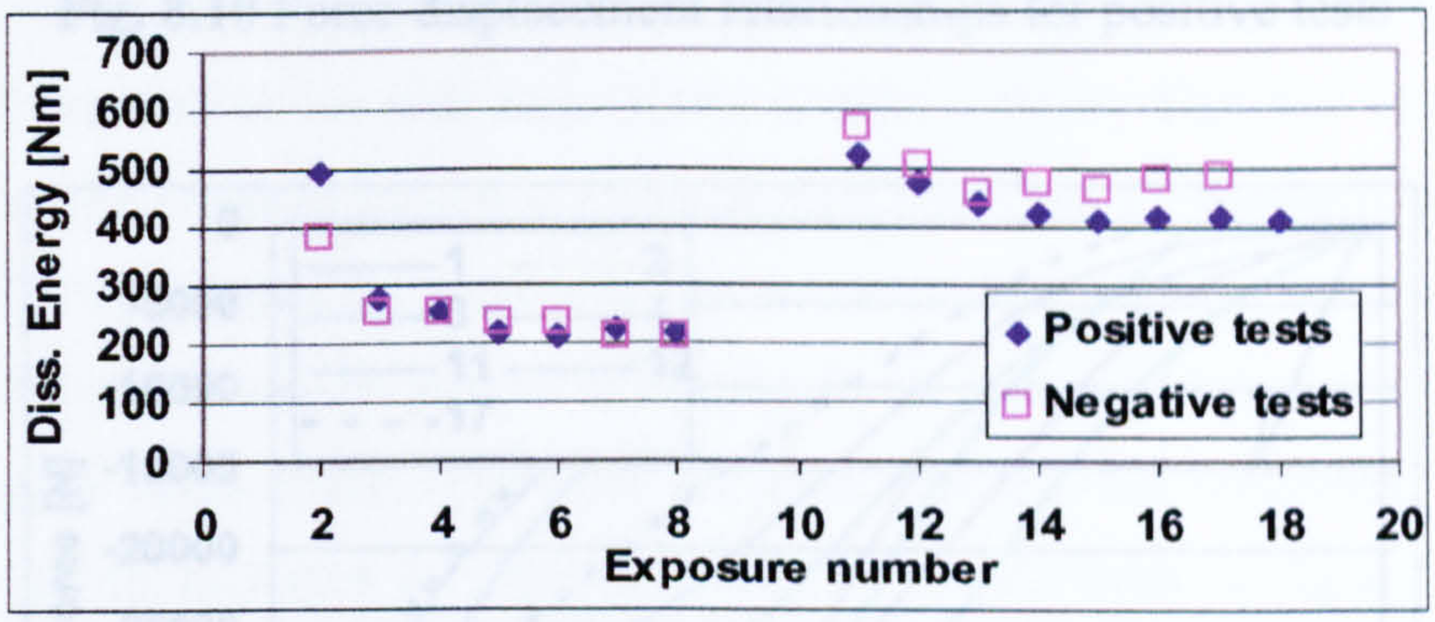


Fig. 8.9 Dissipated/attracted energy vs. exposure number

In figure 8.9, it can clearly be seen how the attracted energy reduces with an increasing number of exposures. This can be seen both for stage 1 (exposures 2-8) and stage 2 (exposures 11-18). This should be considered in light of the fact that the secant frequency reduces consistently from around 2.5Hz to 1.5Hz during stage 1 and continues to decrease during stage 2.

In terms of structural capacity, the pushover type force-displacement relationship cannot be obtained *per se* for each test. This is of course due to the fact that a full pushover tests would introduce plastic behaviour and further damage. However, the force-displacement relationship experienced during the actual test will be available and can be compared. This gives a good indication to the stiffness reduction taking place during the exposures. In figures 8.10 and 8.11 the force-displacement relationship of the first main deformation during selected exposures are displayed, for the positive and negative tests, respectively. The numbers correspond to the exposure numbers.

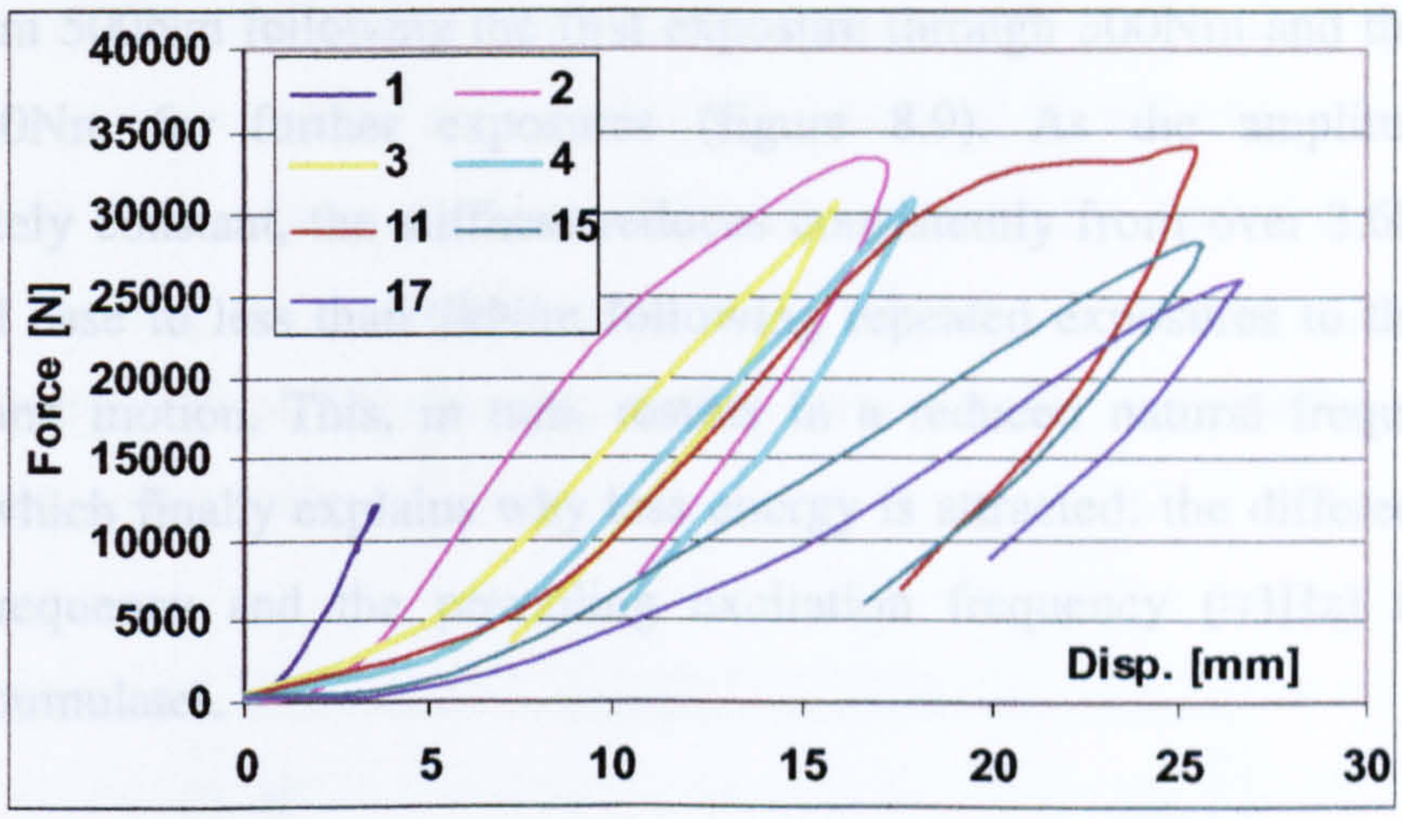


Fig. 8.10 Force-displacement relationships for positive tests

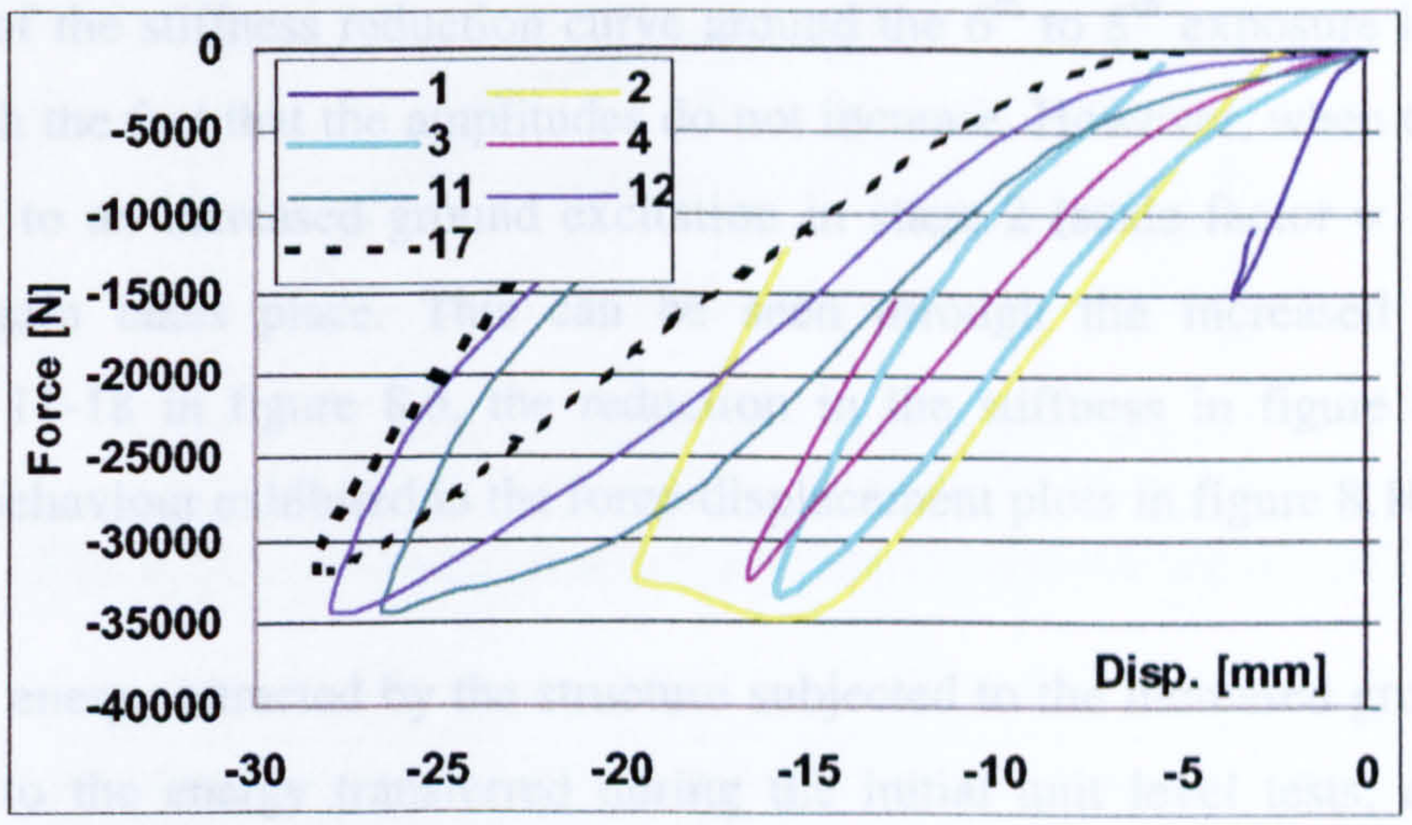


Fig. 8.11 Force-displacement relationships for negative tests

8.3.1.1 Discussion

During the low-level elastic snap-back tests, with amplitudes in the case of LF2 to ±3.6mm, no visible concrete cracking took place. The elastic nature of the response can also be confirmed through the closed force-displacement loop seen in figure 8.4. The response indicated a frequency of 1.9Hz, close to the targeted 2.0Hz.

The first two exposures to the unit scale factor resulted in amplitudes of 17 and 20mm as seen in figure 8.6. The following exposures to the same ground motion did not result in an increase in the amplitude, indicating that the system did not attract more energy. In fact, when inspecting figure 8.9 it can be seen that the attracted energy falls

sharply from 500Nm following the first exposure through 300Nm and then settles at around 200Nm for further exposures (figure 8.9). As the amplitude remains approximately constant, the stiffness reduces consistently from over 3.6kN/m in the undamaged case to less than 2kN/m following repeated exposures to the unit scale factor ground motion. This, in turn, results in a reduced natural frequency of the structure, which finally explains why less energy is attracted: the difference between the eigenfrequency and the prevailing excitation frequency (≈3Hz) increases as damage accumulates.

The structure appears to approach a point where no further damage takes place when it is again exposed to the unit scale factor ground motion. This can be seen in the flattening of the stiffness reduction curve around the 6^{th} to 8^{th} exposure in figure 8.7 and through the fact that the amplitudes do not increase. However, when the structure is exposed to an increased ground excitation in stage 2 (scale factor = 1.5), further damage again takes place. This can be seen through the increased amplitudes, exposures 11-18 in figure 8.6, the reduction in the stiffness in figure 8.7 and the softening behaviour exhibited in the force-displacement plots in figure 8.10 and 8.11.

The initial energy attracted by the structure subjected to the increased ground motion is similar to the energy transferred during the initial unit level tests, around 500-600Nm. This immediately indicates that the structure is now less susceptible to that particular earthquake as the ground motion acceleration is 50% higher. During repeated exposures to the 1.5 scale factor acceleration though, the structure further reduces it susceptibility by attracting less energy, now around 400-500Nm as seen in figure 8.9. At this point, the natural frequency of the structure has almost reduced to 1Hz, with a secant stiffness of only 1kN/mm. The attracted energy and amplitudes appears to flatten out, indicating that the structure again has reached a point where no further damage takes place.

8.3.2 The high frequency structure

The high frequency specimens are referred to as *HF1* and *HF2* throughout this section. Preliminary numerical simulations indicated that a scale factor of as much as 10 was required to obtain suitable initial displacements. This was based on linear

analyses using the undamaged and slightly reduced stiffnesses. But, again prior to exposing the structure to the main excitation, low-level elastic tests were required to determine the undamaged properties. This was successfully carried out for both the specimens yielding very similar properties. Figures 8.12 and 8.13 show the undamaged force-displacement relationship and elastic response to the 3.6mm displacement, respectively.

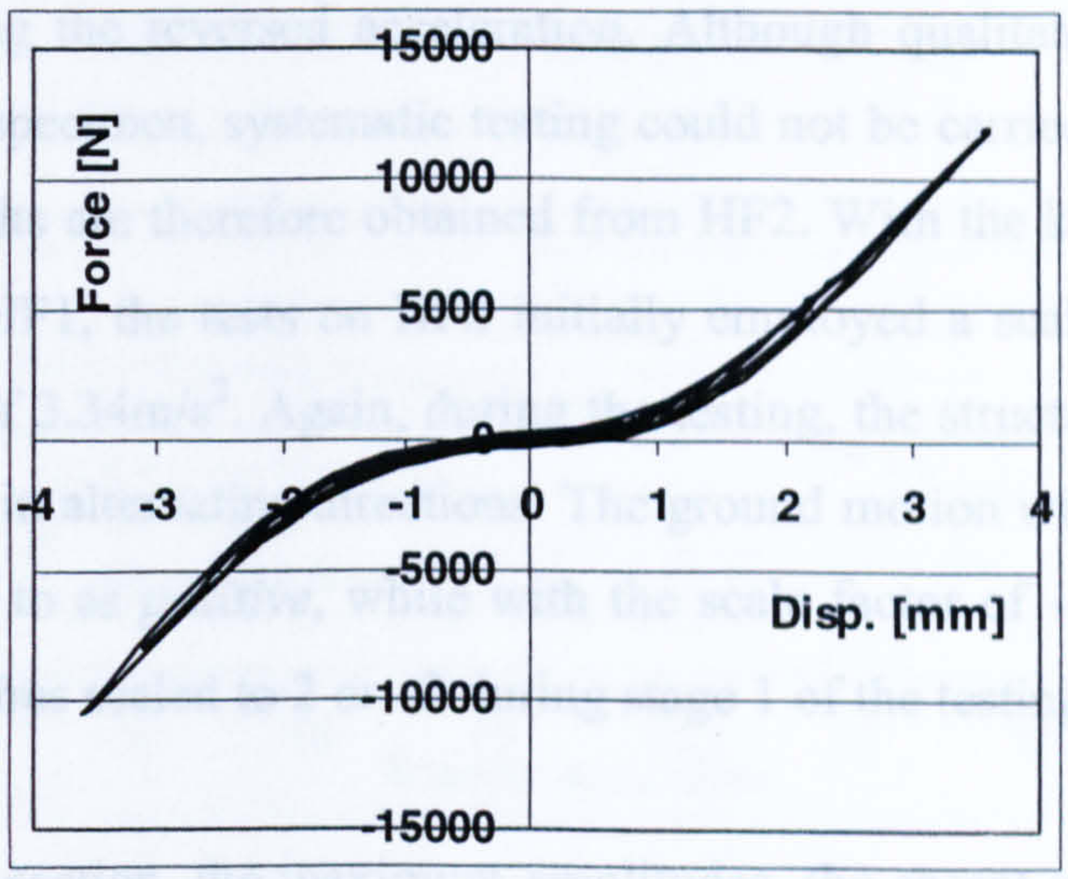


Fig. 8.12 Undamaged force-displacement relationship for HF2

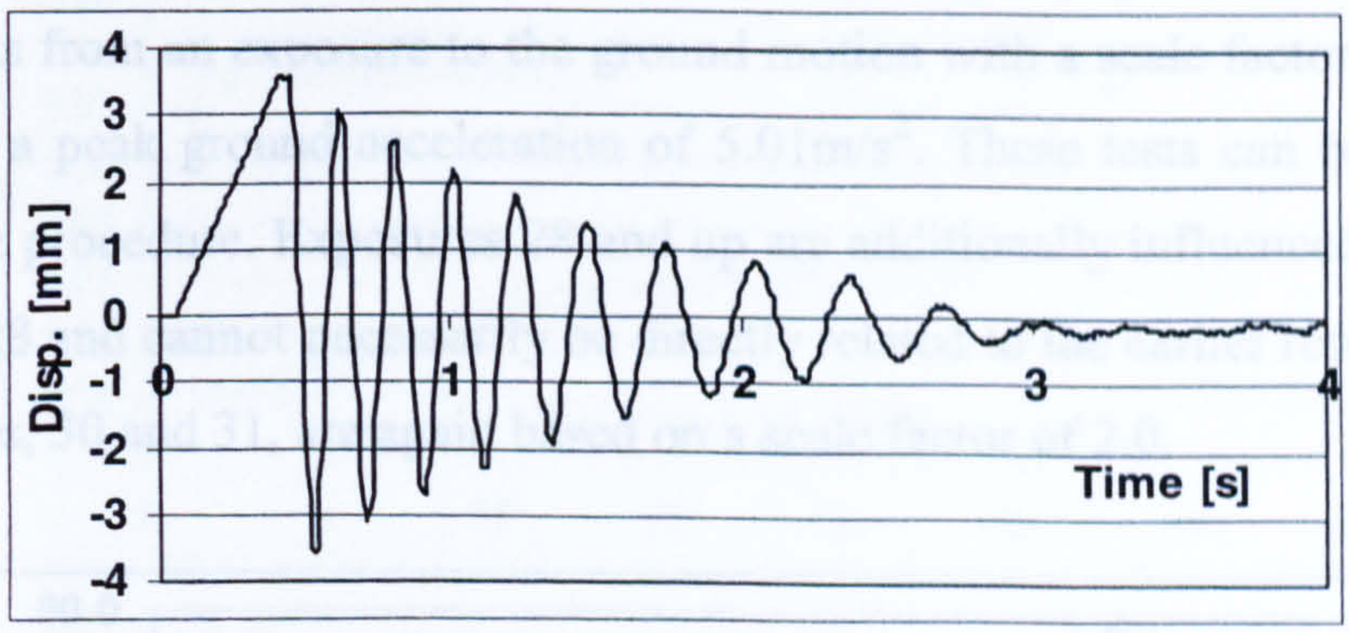


Fig. 8.13 Elastic response to initial 3.6mm displacement

The force-displacement relationship in figure 8.12 is near identical with that in figure 8.4 as the structural stiffness properties are essentially the same. However, the dynamic properties are completely different as the assigned mass is now only 2000kg as opposed to 18000kg in the earlier tests. The response curve in figure 8.13 indicates a frequency of over 5Hz during the first cycles. This can be contrasted to the 1.9Hz exhibited by the low frequency specimen. The response curve in figure 8.13 compared

with figure 8.5 also indicates that there is effectively more damping present in the high-frequency system.

The first full-scale high frequency test on HF1 consisted of an exposure to the Port Hueneme acceleration scaled to 16.7m/s^2. This acceleration turned out to produce higher displacements than anticipated through the numerical simulations, possibly because damage taking place in the first half-wave resulted in highly increased susceptibility during the reversed acceleration. Although qualitative tests were still carried out on the specimen, systematic testing could not be carried out satisfactorily. The following results are therefore obtained from HF2. With the knowledge obtained from the tests on HF1, the tests on HF2 initially employed a scale factor of 2, or a peak acceleration of 3.34m/s^2. Again, during the testing, the structure was exposed to the ground motion in alternating directions. The ground motion with a scale factor of +2 is now referred to as *positive*, while with the scale factor of –2 as *negative*. The accelerogram was thus scaled to 2 or –2 during stage 1 of the testing.

As in the previous section, the maximum amplitudes, the secant stiffness, the secant frequency and attracted energies are plotted as a function of exposure numbers. In this case, exposure 1 again equals the low-level elastic test. However, exposures 25 to 29 display results from an exposure to the ground motion with a scale factor of 3.0, or in other words, a peak ground acceleration of 5.01m/s^2. These tests can be considered stage 2 of the procedure. Exposures 28 and up are additionally influenced by a failure in exposure 28 and cannot necessarily be directly related to the earlier results. The last two exposures, 30 and 31, are again based on a scale factor of 2.0.

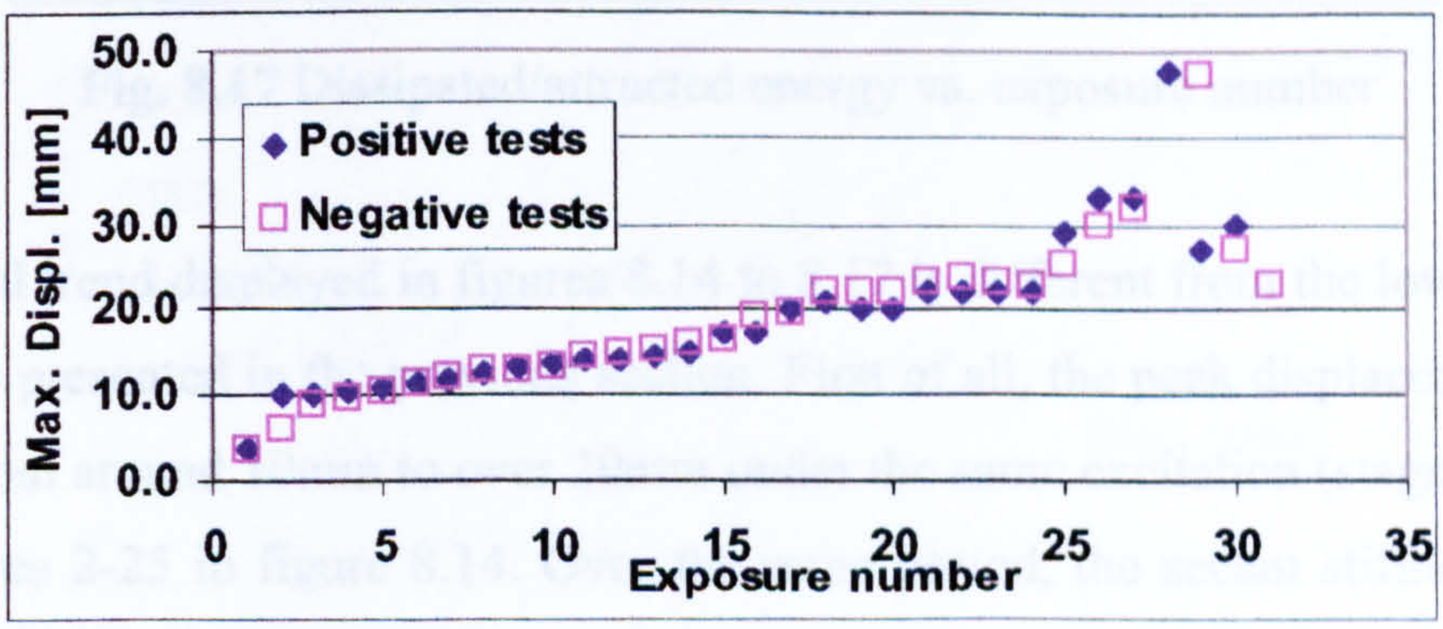


Fig. 8.14 Peak amplitude vs. exposure number

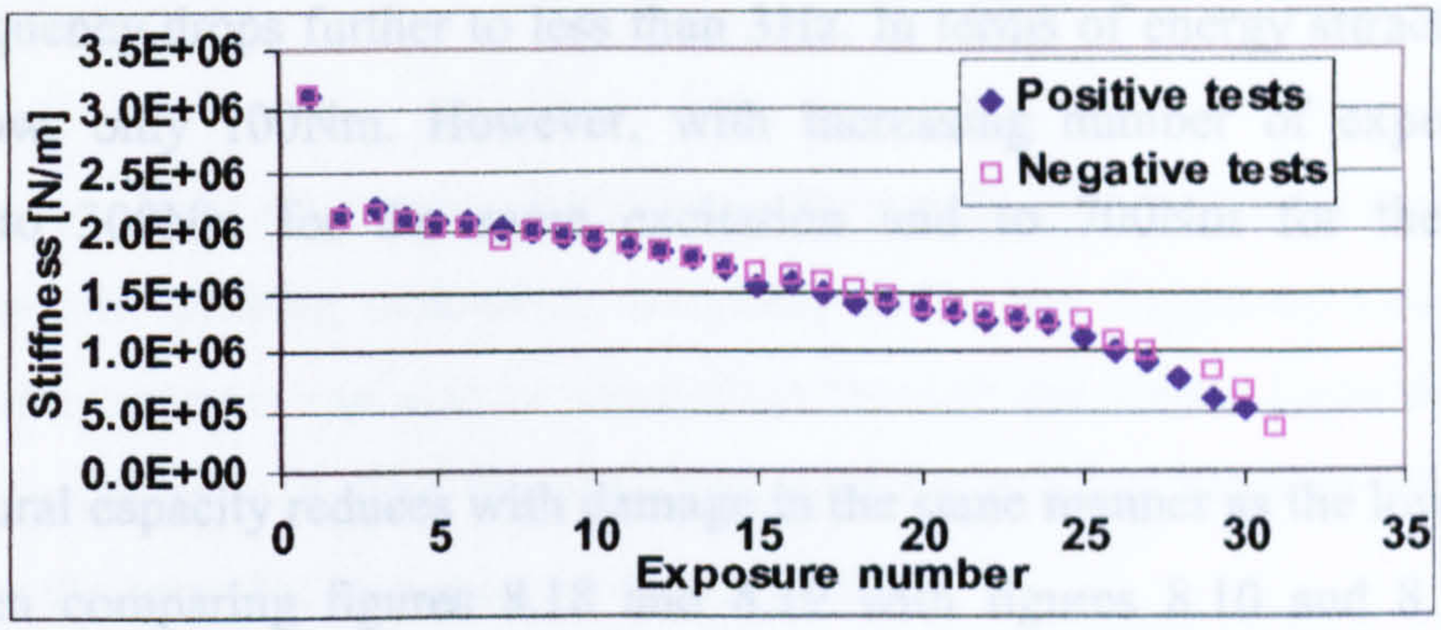


Fig. 8.15 Stiffness vs. exposure number

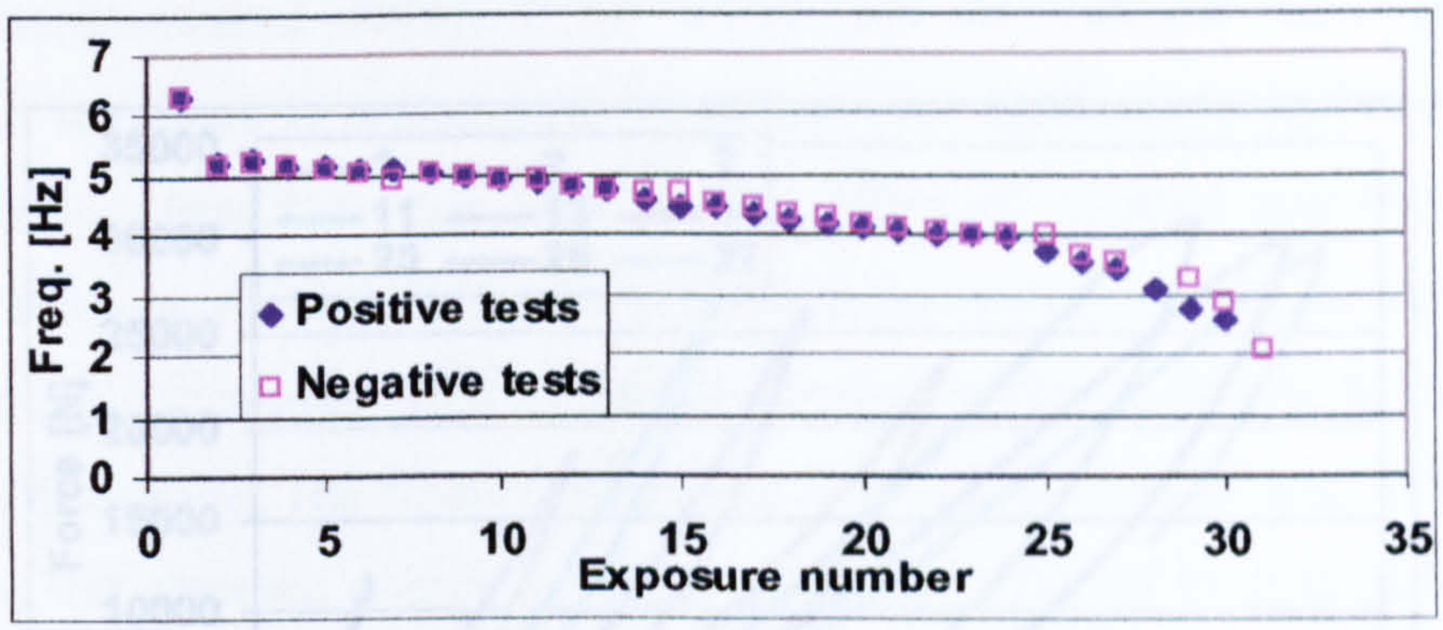


Fig. 8.16 Frequency vs. exposure number

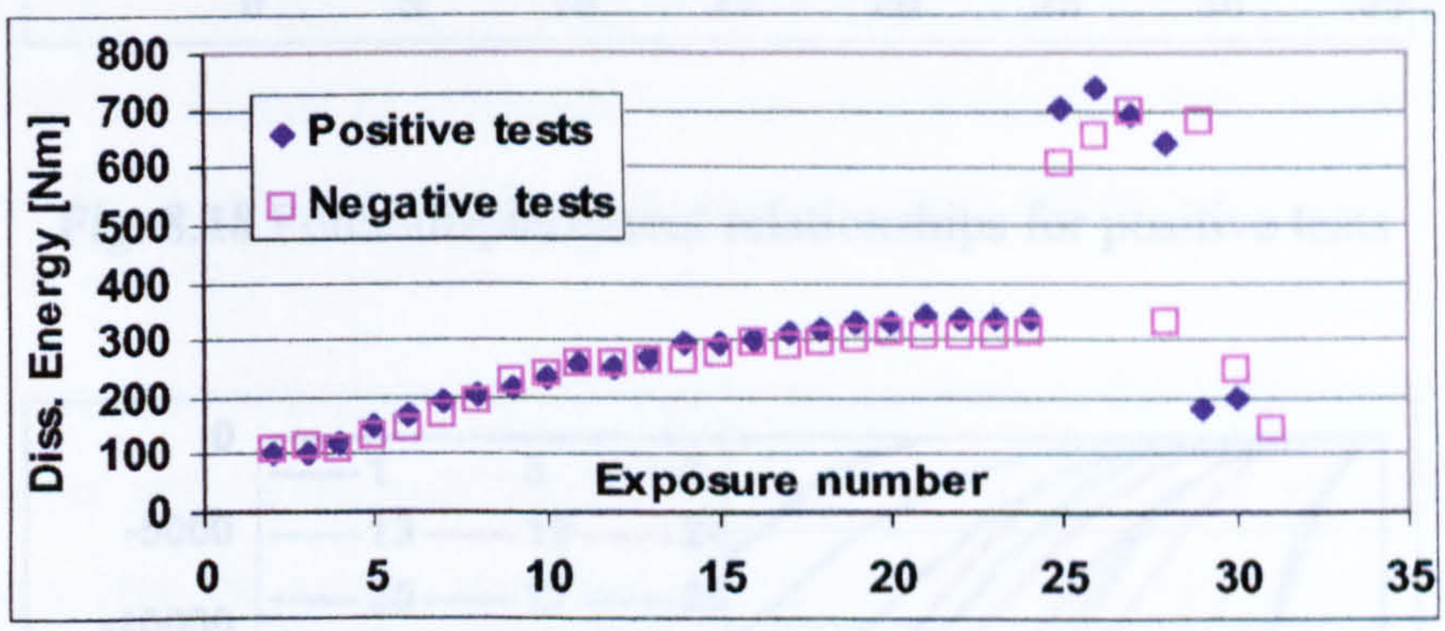


Fig. 8.17 Dissipated/attracted energy vs. exposure number

The general trend displayed in figures 8.14 to 8.17 is different from the low frequency test results presented in the previous section. First of all, the peak displacements grow steadily from around 10mm to over 20mm under the same excitation (stage 1), as seen in exposures 2-25 in figure 8.14. Over the same period, the secant stiffness reduces from over 3kN/mm to less than 1kN/mm with the corresponding drop in frequency from over 6Hz to around 4Hz. Following exposure to the increased excitation (stage

2), the frequency drops further to less than 3Hz. In terms of energy attraction, this is initially low, only 100Nm. However, with increasing number of exposures, this increases to 300Nm for the same excitation and to 700Nm for the increased excitation.

The structural capacity reduces with damage in the same manner as the low frequency tests. When comparing figures 8.18 and 8.19 with figures 8.10 and 8.11, similar softening behaviour can be observed. This should be expected, as the relationship is independent of the mass and dynamic properties.

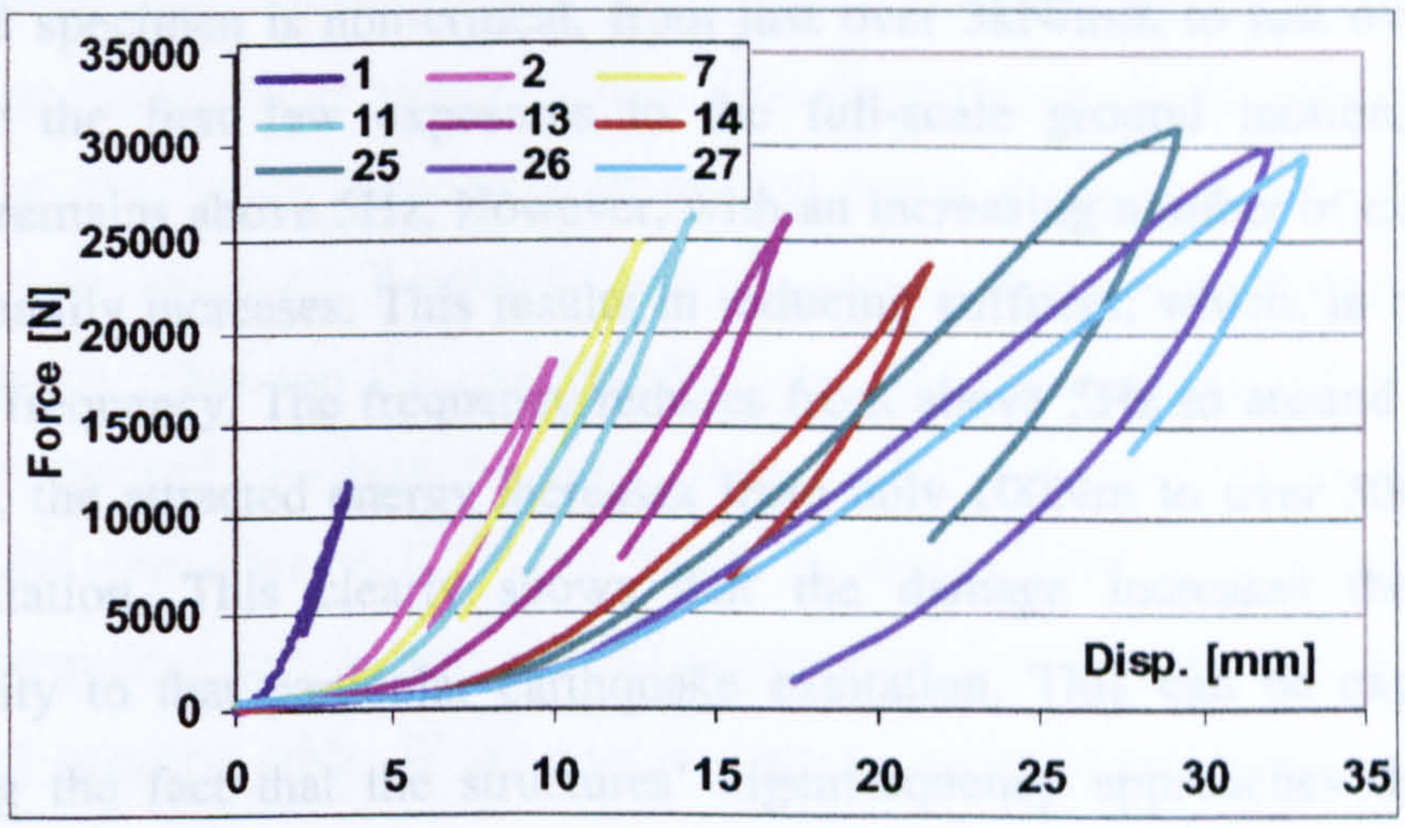


Fig. 8.18 Force-displacement relationships for positive tests

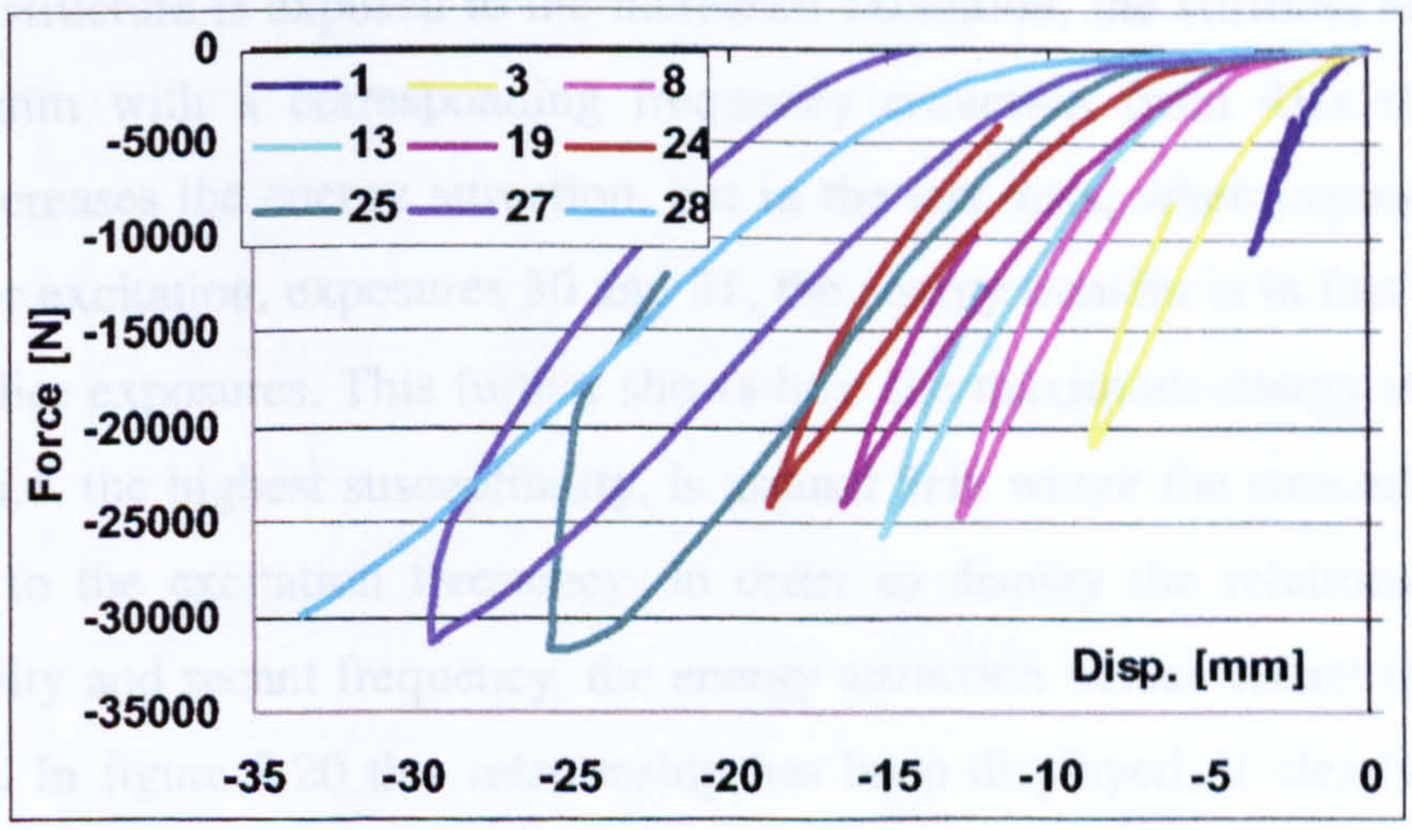


Fig. 8.19 Force-displacement relationships for negative tests

8.3.2.1 Discussion

During the low-level elastic snap-back tests, again with amplitudes up to ±3.6mm, no visible concrete cracking took place. Similarly to the low frequency test, the elastic nature of the response can also be confirmed through the closed force-displacement loop seen in figure 8.12. The response indicated frequencies between 4Hz and 5Hz, suitable for the intended frequency of 4Hz or above.

With the first exposure to the full-scale ground excitation, the initial damage is limited. Displacements remain below 10mm and the stiffness reduction from the undamaged specimen is non-critical, from just over 3kN/mm to just over 2kN/mm. Even after the first few exposures to the full-scale ground motion, the secant frequency remains above 5Hz. However, with an increasing number of exposures, the damage steadily increases. This results in reducing stiffness, which, in turn, reduces the secant frequency. The frequency reduces from above 5Hz to around 4Hz. At the same time, the attracted energy increases from only 100Nm to over 300Nm for the same excitation. This clearly shows that the damage increases the structure's susceptibility to that particular earthquake excitation. This can be expected when considering the fact that the structures' eigenfrequency approaches the excitation frequency. It can further be contrasted to the low frequency tests where the damage rather distances the eigenfrequency from the excitation frequency.

When the structure is exposed to the increased excitation, the stiffness reduces down to 0.5kN/mm with a corresponding frequency reduction from 4Hz to 2Hz. This initially increases the energy attraction, but in the last tests, when exposed to the 2.0 scale factor excitation, exposures 30 and 31, the energy transfer is in fact smaller than during earlier exposures. This further shows how the maximum energy transfer to the structure, i.e. the highest susceptibility, is around 3Hz where the structural frequency is closest to the excitation frequency. In order to display the relationship between susceptibility and secant frequency, the energy attraction versus secant frequency can be plotted. In figure 8.20 this relationship has been displayed. It clearly shows how the energy attraction increases as the frequency decreases down towards 3Hz. Once passed, the energy transfer can be interpreted as decreasing again. In figure 8.20, all the plot points result from the 2.0 scale factor excitations.

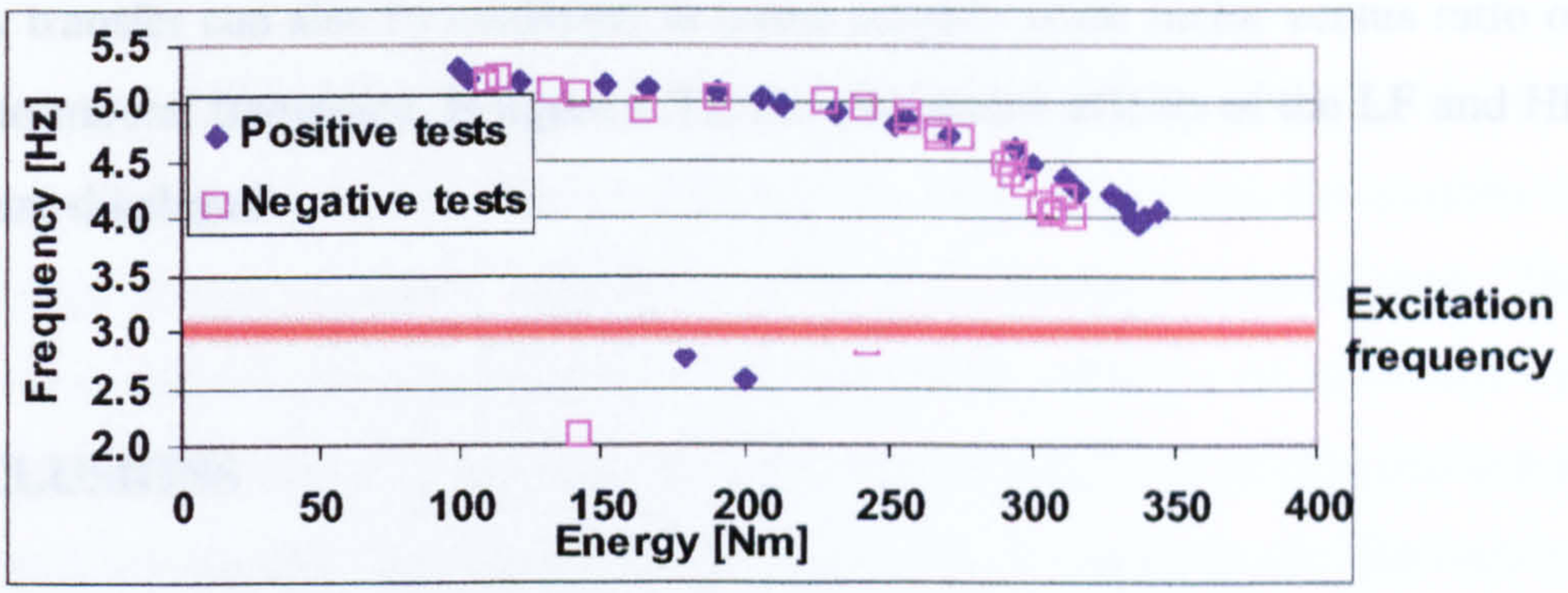


Fig. 8.20 Dissipated/attracted energy vs. frequency

In figure 8.20 above, it appears that the maximum energy transfer does not exactly coincide with the 3Hz excitation frequency. It seems that the energy transfer is highest with a secant frequency of around 3.5Hz. This discrepancy can be explained through the fact that the secant frequency is not an exact representation of the natural frequency of the structure. It tends to suggest a somewhat higher frequency than that actually present. This can for example be seen with the undamaged HF specimen, where the measured vibration frequency was around 5Hz and the secant frequency just over 6Hz (see figures 8.13 and 8.16). In figure 8.20, it is therefore not the excitation frequency that is inaccurately represented, but the plot points for both the positive and negative tests. If the energy transfer were to be plotted against actual natural frequency, one can envisage that the frequency values of all the points would be reduced by between 0Hz and 1Hz. The maximum energy transfer would thus coincide better with 3Hz excitation frequency.

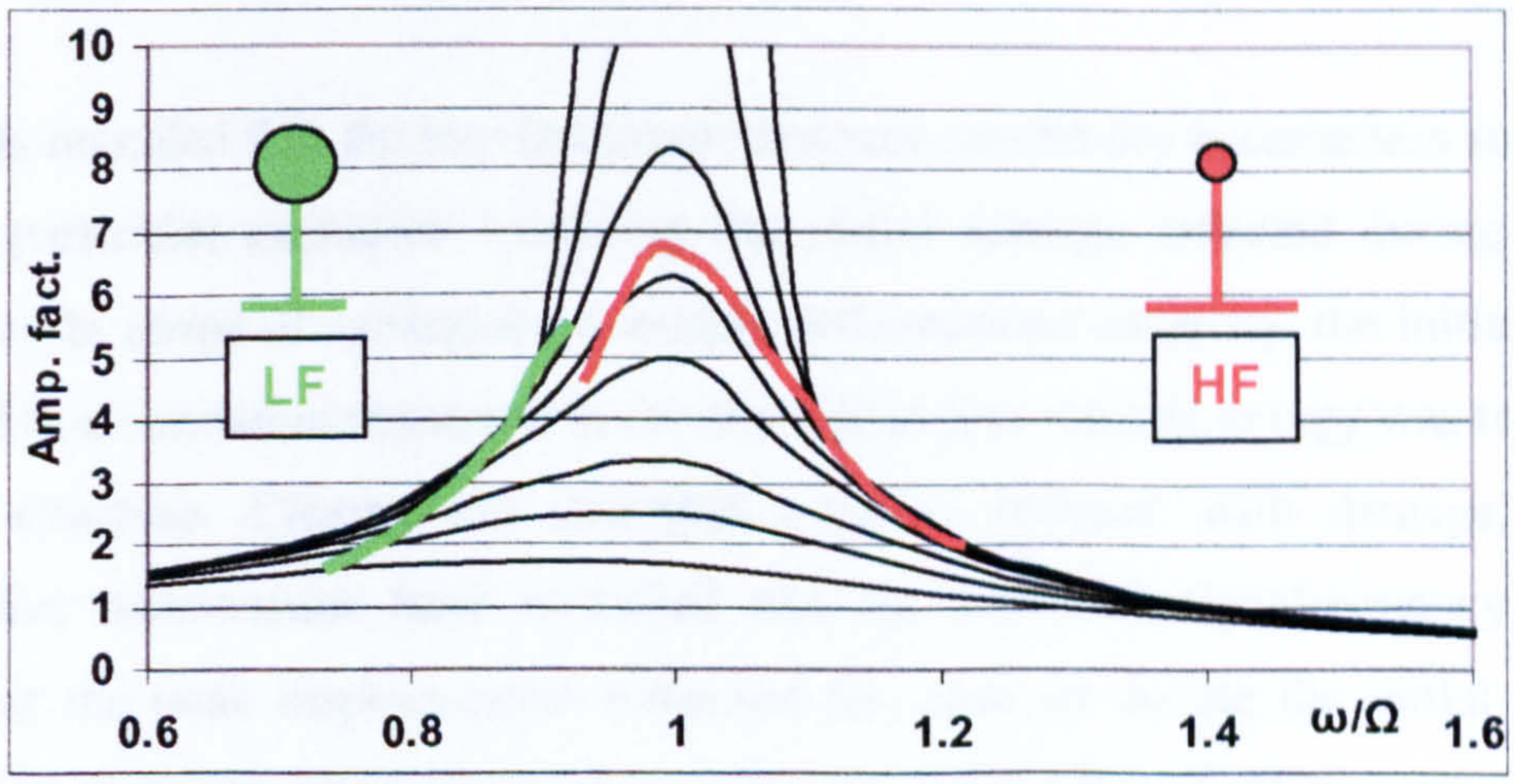


Fig. 8.21 Structural behaviour for LF and HF in terms of resonance

The energy transfer can also be exhibited in terms amplification factor versus ratio of excitation to natural frequency. In figure 8.21, the qualitative effects of the LF and HF structures are displayed.

8.4 CONCLUSIONS

The objectives of this chapter were twofold: to demonstrate the applicability of the pseudodynamic test system documented throughout this thesis and to investigate potential effects of repeated earthquake loading on a structure. Considering first the application value of the test system developed, it has been shown that the fully software based semi-continuous control-execution system, employing the novel Newmark Implicit – Integral Form time stepping scheme, is well suited for systematic testing of SDOF reinforced concrete structures. The adaptability of the implementation software was also exhibited by the incorporation of energy monitoring and tracing systems.

The tests further successfully evaluated the response of two dynamic systems to repeated Port Hueneme earthquake excitations. The two SDOF systems comprised a low frequency structure (LF), initially displaying an eigenfrequency of around 2Hz and a high frequency structure (HF), with an initial eigenfrequency of around 4Hz. The 1957 Port Hueneme accelerogram consists of one main acceleration cycle with a predominant excitation frequency of around 3Hz.

The tests revealed that the low frequency structure essentially became less susceptible to that particular excitation following the initial damage inflicted during the first exposure. In terms of earthquake demands and structural capacity, the initial damage resulted in a consistent reduction in the demand as less seismic energy was transferred to the structure. Clearly, the structural capacity reduced with damage, but the dissipative mechanisms have increased and the structural eigenfrequency changed such that the peak displacements remained the same as during the initial exposure stage. The structure appeared to reach a "steady state" where further exposures to the same excitation did not result in any further damage. This is clearly a result of the structural eigenfrequency drifting further away from the excitation frequency.

The high frequency structure displayed behaviour significantly different from the low frequency one. Following limited damage during the initial exposures, the transfer of seismic energy to the structure steadily increased as damage accumulated. The earthquake demand was thus increasing as the structural stiffness reduced and the eigenperiod extended. The structural capacity naturally reduced in a similar fashion to the low frequency structure, clearly resulting in an unstable scenario. As the damage became so great that the stiffness was reduced to 1/6 of the original, the eigenfrequency of the structure was such that the energy transfer again started reducing. In effect, the structure passed through the resonance response correlated with the predominant excitation frequency. However, the reduction in the energy transfer did not occur until the structure was beyond the point at which it could be defined as failed. It can thus be concluded that the high frequency structure is less resistant than the low frequency structure to the repeated earthquake excitations of the particular kind (predominant single frequency shock) employed in these tests.

Chapter IX:

CONCLUSIONS

The research documented in this thesis pertains to the development of a SDOF pseudodynamic facility for testing structural components under seismic loading. The pseudodynamic test method is a hybrid technique in which the inertia and viscous damping components are modelled computationally while the non-linear structural restoring forces are obtained experimentally. The research has considered all the aspects related to pseudodynamic implementation: the servo-hydraulic experimental testing system, the digital control of this, the computerised execution system and of course the time stepping scheme, which forms a central part of the pseudodynamic formulation.

This concluding chapter aims to highlight the principal achievements and novel developments attained during this research on the pseudodynamic test method and to make some suggestions to future work. With the method being of a hybrid type, the research has targeted both the experimental and computational aspects of the method, with key innovations realised within both parts. The following sections will summarize the main contributions from each chapter, presented in order of appearance in this thesis.

- A complete experimental facility for SDOF pseudodynamic testing has been designed the by the author. The construction of this, as well as the fabrication of all the required structural components and instrumentation, has also been supervised by the author.

- Substantial work has been carried out on integral form time stepping algorithms, leading to the development of the novel Newmark Implicit – Integral Form time stepping scheme. The scheme constitutes an improvement to the earlier proposed explicit versions of the algorithm. This alternative formulation not only improves the dissipative characteristics of the algorithm, but also ensures unconditional stability. The improvements have been shown both through numerical examples and analytically, and systems for implementation into the pseudodynamic framework have been suggested. The improvements to the method now enable the general advantages of the integral

form formulation to be combined with the unlimited time step size associated with the unconditional stability offered with the implicit version.

- A digital control formulation has been devised, which employs proportional and integral control modes and includes a system for compensation of the dead-zone present within the valve. The control formulation also considers the intra-step actuator motion by taking the information for the next time step into account. It further includes systems for compensating for transducer non-linearities and an algorithm for numerically integrating the restoring force over each time step.

- A computational implementation system has been developed within the LabView environment. The LabView environment is a graphical programming environment, specifically designed for hardware communication and also offers excellent graphical output. The implementation system encompasses an execution scheme, which ensures overall control of the pseudodynamic tests. The scheme offers interactive input and output within a graphical environment for setting up and monitoring tests and also contains the main time stepping algorithm. Furthermore, it offers the possibility for manual control of the experimental system and is also responsible for data logging. Uniquely, the valve controller is coded as a nested loop within the main execution algorithm. This implies that the controller exists entirely on a software level, still within the LabView environment. Such a set-up offers an excellent opportunity for an integration of the control and time stepping routines. This has been fully taken advantage of through the development of the semi-continuous implementation system, which ensures continuous actuator motion from one time step to another. The semi-continuous motion enables increased actuator speeds while retaining high accuracy. Further combined with the integral form time stepping scheme, the potential effects of actuator overshoot are minimised.

- The complete implementation system has been tested and verified through an extensive verification and sensitivity study. These studies have investigated the algorithmic and control related effects on the pseudodynamic generation of

the seismic response of a structure to a ground excitation. By considering both linear and non-linear structures, the sensitivities and inter-related effects of choice of time stepping scheme, time step size, implementation speed and accuracy have been examined. It has been concluded that in this set-up, higher implementation speeds can be maintained with larger time step sizes, but that the size of time step affects the response more than the speed of implementation. Furthermore, the choice of the time integration algorithm and the time step size may affect the response significantly, but that secondary effects on the control are not significant.

- The devised pseudodynamic testing system has been applied in an experimental study of potential effects of repeated earthquake loading on a single structure. The investigation considered the repeated exposure of two different reinforced concrete structural components to scaled 1957 Port Hueneme excitations. The study concluded that if the structure displayed an eigenfrequency already below the prevailing frequency of the excitation, the structure effectively became less susceptible to that particular excitation through repeated exposure. It could be argued that the structure reached a steady state where further exposures inflicted no additional damage. On the other hand, if the structure initially exhibited an eigenfrequency higher than the prevailing excitation frequency, damage and stiffness reduction effectively increased the susceptibility to that excitation, eventually leading to failure.

In terms of future work, this can be considered along two main routes. The possibility of expanding the test facility could be considered or one could concentrate wholly on application of the existing system for new problems. Both routes offer a number of possibilities, and some options are suggested below.

For expansion of the test facility, the prospect of upgrading both the experimental and computational hardware should be considered. A larger actuator, with higher capacity and stroke, would enable a substantially larger range of specimen types to be tested. Combined with upgraded computational hardware, this could provide the means for further research on high-speed or real-time SDOF implementation. Alternatively, the system could be expanded with one or more DOFs to constitute a MDOF test facility.

This would require substantial expansion of both the experimental hardware and the control system. Such an expansion would enable the both Newmark Implicit – Integral Form formulation and the LabView execution and control systems to be tested under much more demanding conditions.

Further research could alternatively be carried out with the existing test facility. This could be in terms of the computational execution/control or time integration system or purely as an application. Following the application of the system for investigating effects of repeated earthquake loading on a single structure, this could be expanded to consider a range of base excitations and a range of structures. Rather than the essentially single shock Port Hueneme excitation, accelerograms with wider frequency content could be applied. If the test facility were to be expanded to a MDOF system as well, the structural behaviour would be much more complex and difficult to predict through simplified analyses.

REFERENCES

Aktan, H.M. and Hashish, A.A. (1985), "Pseudodynamic testing of structural components", *Dynamic Response of Structures*, Proc. 3rd conference by Engineering Mechanics Division, ASCE, Los Angeles, pp. 131-137

Algaard, W., Agar, A. and Bićanić, N. (2000), "Computational aspects of pseudodynamic testing", *Proc. 8th Ann. Conf. of the Association for Computational Mechanics in Engineering (ACME)*, Greenwich, UK, April 2000, pp. 12-15

Algaard, W., Bićanić, N. and Agar, A. (2001a), "Assessment of algorithmic and control sensitivities in pseudodynamic testing", *Proc. 10th Int. Conf. on Computational Methods and Experimental Measurements (CMEM)*, Alicante, Spain, June 2001, pp. 879-888

Algaard, W., Agar, A. and Bićanić, N. (2001b), "Enhanced integral form of the Newmark time stepping scheme for pseudodynamic testing", *Engineering Computations*, Vol. 18, pp. 676-689

Algaard, W., Agar, A. and Bićanić, N. (2001c), "Algorithmic improvements of pseudodynamic testing", *Proc. 9th Ann. Conf. of the Association for Computational Mechanics in Engineering (ACME)*, Birmingham, April 2001, pp. 89-94

ATC, Applied Technology Council (1996), "Seismic evaluation and retrofit of concrete buildings", Report No. SSC 96-01, Redwood City, California, 1996

Bateson, R.N. (1996), "Introduction to control system technology", Prentice-Hall, Inc., New Jersey, 1996

Bathe, K.-J. and Wilson, E.L. (1976), "Numerical methods in finite element analyses", Prentice-Hall, Inc., New Jersey, USA

Beck, J.L. and Jayakumar, P. (1986), "System identification applied to pseudodynamic test data: A treatment of experimental errors", *Dynamic Response of Structures*, Proc. 3rd conference by Engineering Mechanics Division, ASCE, Los Angeles, USA, pp. 505-512

Buchet, P. and Pegon, P. (1994), "PSD testing with substructuring: Implementation and use", JRC Ispra Special Publication no: I.94.25, Ispra, Italy, 1994

Bursi. O.S., Shing, P.B. and Radakovic-Guzina, Z. (1994), "Pseudodynamic testing of strain-softening systems with adaptive time steps", *Earthquake Engineering and Structural Dynamics*, Vol. 23, pp. 745-760

Chang, S-Y., Tsai, K-C. and Chen, K-C. (1998), "Improved time integration for pseudodynamic tests", *Earthquake Engineering and Structural Dynamics*, Vol. 27, pp. 711-730

Chang, S-Y. (1997), "Improved numerical dissipation for explicit methods in pseudodynamic tests", *Earthquake Engineering and Structural Dynamics*, Vol. 26, pp. 917-929

Chung, W-J., Yun, C-B., Kim, N-S. and Seo, J-W. (1999), "Shaking table and pseudodynamic tests for the evaluation of the seismic performance of base-isolated structures", *Engineering Structures*, Vol. 21, pp. 365-379

Combescure, P. and Pegon, P. (1997), "α-Operator splitting time integration technique for Pseudodynamic testing Error propagation analysis", *Soil Dynamics and Earthquake Engineering*, Vol. 16, pp. 427-443

Cuadra, C. and Ogawa, J. (2001), "Experimental errors in pseudodynamic test using conventional testing devices", , *Proc. 10th Int. Conf. on Computational Methods and Experimental Measurements (CMEM)*, Alicante, Spain, June 2001, pp. 869-876

Darby, A.P., Blakeborough, A. and Williams, M.S. (1999), "Real-time substructuring tests using hydraulic actuator", *Journal of Engineering Mechanics*, Vol. 125, pp. 1133-1139

Géradin, M. and Rixen, D. (1994), "Mechanical Vibrations – Theory and application to structural dynamics" Wiley, Masson, Paris, France 1994

Hilber, H.M., Hughes, T.J.R., Taylor, R.L. (1977), "Improved numerical dissipation for time integration algorithms in nonlinear transient analyses.", *Earthquake Engineering and Structural Dynamics*, Vol. 5, pp. 283-292

Horiuchi, T., Inoue, M., Konno, T. and Namita, Y. (1999), "Real-time hybrid experimental system with actuator delay compensation and its application to a piping system with energy absorber", *Earthquake Engineering and Structural Dynamics*, Vol. 28, pp. 1121-1141

Kumar, S., Itoh, Y., Saizuka, K. and Usami, T. (1997), "Pseudodynamic testing of scaled models", *Journal of Structural Engineering*, Vol. 123, pp. 524-526

Magonette, G.E., Pegon, P., Molina, J. and Buchet, P. (1998), "Development of fast continuous Pseudodynamic substructuring tests", *In Proc. of 2nd World Conference on Structural Control*, June 1998, Kyoto, Japan

Magonette, G.E., Pegon, P. and Pinto, A.V. (2000), "The new ELSA laboratory", *European Commission, Joint Research Centre, ELSA Laboratory*, Ispra, Italy

Mahin, S.A. and Shing, P.B. (1985), "Pseudodynamic method for seismic testing", *Journal of Structural Engineering*, Vol. 111, pp. 1482-1503

Mahin, S.A., Shing, P.B., Thewalt, C.R., Hanson, R.D. (1989), "Status and future directions" *Journal of Structural Engineering*, Vol. 115, pp. 2113-2128

Nakashima, M. and Masaoka, N. (1999), "Real-time on-line test for MDOF systems", *Earthquake Engineering and Structural Dynamics*", Vol. 28, pp. 393-420

Negro, P. (1997), "Combined experiment and computation for dynamic testing of structures", *Ph.D. thesis*, University of Wales, Swansea, UK

Negro, P., Pinto, A.V., Verzeletti, G and Magonette, G.E. (1996), "PsD test on four-story R/C buildings designed according to Eurocodes", *Journal of Structural Engineering*, Vol. 122, pp. 1409-1417

Peek, R. and Yi, W. (1990a), "Error analysis for pseudodynamic test method. I: Analysis", *Journal of Engineering Mechanics*, Vol. 116, pp. 1618-1637

Peek, R. and Yi, W. (1990b), "Error analysis for pseudodynamic test method. II: Application", *Journal of Engineering Mechanics*, Vol. 116, pp. 1638-1658

Pegon, P. and Pinto, A.V. (2000), "Pseudo-dynamic testing with substructuring at the ELSA laboratory", *Earthquake Engineering and Structural Dynamics*, Vol. 29, pp. 905-925

Shing, P.B., Bursi, O.S. and Vannan, M.T. (1994), "Pseudodynamic tests of a concentrically braced frame using substructuring techniques", *Journal of Constructional Steel Research*, Vol. 29, pp. 121-148

Shing, P.B. and Mahin, S.A. (1986), "Cumulative experimental errors in pseudodynamic tests", *Earthquake Engineering and Structural Dynamics*, Vol. 15, pp. 409-424

Shing, P.B. and Mahin, S.A. (1987), "Elimination of spurious higher-mode response in pseudodynamic tests", *Earthquake Engineering and Structural Dynamics*, Vol. 15, pp. 425-445

Shing, P.B. and Mahin, S.A. (1988), "Rate of loading effects on PsD tests", *Journal of Structural Engineering*, Vol. 114, pp. 2403-2420

Shing, P.B. and Vannan, M.T. (1990), "On the accuracy of implicit time integration for pseudodynamic tests", *Earthquake Engineering and Structural Dynamics*, Vol. 19, pp. 632-651

Shing, P.B., Vannan, M.T. and Cater, E. (1991), "Implicit time integration for pseudodynamic tests", *Earthquake Engineering and Structural Dynamics*, Vol. 20, pp. 551-576

Thewalt, C.R. and Mahin, S.A. (1995), "An unconditionally stable hybrid pseudodynamic algorithm", *Earthquake Engineering and Structural Dynamics*, Vol. 24, pp. 723-731

Thewalt, C.R. and Roman, M. (1994), "Performance parameters for pseudodynamic tests", *Journal of Structural Engineering*, Vol. 120, pp. 2768-2781

Vannan, M.T. (1991), "The pseudodynamic test method with substructuring applications", *Ph.D. thesis*, University of Colorado, Boulder, USA

Williams, M.S. and Blakeborough, A. (1998), "Real-time hybrid dynamic testing of structures", *Department of Engineering Science, University of Oxford*, http://www-civils.eng.ox.ac.uk/research/structdyn.html

Williams, M.S., Blakeborough, A., Clement, D., Darby, A.P., Williams, D.M. and Woodward, N. (1999), "Development and simulation of a real-time dynamic testing procedure for structures", *Development in analysis and design using Finite Element Analysis*, Civil-Comp Press, Edinburgh, pp. 133-140

Yamazaki, Y., Nakashima, M. and Kaminosono, T. (1986), "Earthquake response simulation capacity of pseudodynamic testing – Experimental demonstration and analytical evaluation", *Dynamic Response of Structures*, Proc. 3rd conference by Engineering Mechanics Division, ASCE, Los Angeles, pp. 446-453

BIBLIOGRAPHY

Bursi, O.S. and Shing, P.B. (1996), "Evaluation of some implicit time-stepping algorithms for pseudodynamic tests", *Earthquake Engineering and Structural Dynamics*", Vol. 25, pp. 333-355

Buonopane, S.G. and White, R.N. (1999), "Pseudodynamic testing of masonry infilled reinforced concrete frame", *Journal of Structural Engineering*, Vol. 125, pp. 578-589

Chen, C.-C. and Robinson, A.R. (1993), "Improved time-history analysis for structural dynamics, I: Treatment of rapid variation of excitation and material non-linearity", *Journal of Engineering Mechanics*, Vol. 119, pp. 2496-2513

Chen, C.-C. and Robinson, A.R. (1993), "Improved time-history analysis for structural dynamics, II: Reduction of effective number of degrees of freedom", *Journal of Engineering Mechanics*, Vol. 119, pp. 2514-2530

Cho, H. and Bai, E.-W. (1998), "Convergence results for an adaptive dead zone inverse", *International Journal of Adaptive Control and Signal Processing*, Vol. 12, pp. 451-466

Dimig, J., Shield, C., French, C., Bailey, F. and Clark, A. (1999), "Effective force testing: A method of seismic simulation for structural testing", *Journal of Structural Engineering*, Vol. 125, pp. 1028-1037

Golley, B.W. and Amer, M. (1999), "An unconditionally stable time-stepping procedure with algorithmic damping: A weighted integral approach using two general weight functions", *Earthquake Engineering and Structural Dynamics*, Vol. 28, pp. 1345-1360

Gutiérrez, E. and Zaldivar, J.M. (2000), "The application of Karhunen-Loéve, or principal component analysis method, to study the non-linear seismic response of structures", *Earthquake Engineering and Structural Dynamics*, Vol. 29, pp. 1261-1296

Kim, N.-S. and Lee, D.-G. (1995), "Pseudodynamic test for evaluation of seismic performance of base-isolated liquid storage tanks", *Engineering Structures*, Vol. 17, pp. 198-208

De Luca, A., Mele, E., Molina, J., Verzeletti, G. and Pinto, A.V. (2001), "Base isolation for retrofitting historic buildings: Evaluation of seismic performance through experimental investigation", *Earthquake Engineering and Structural Dynamics*, Vol. 30, pp. 1125-1145

Mo, Y.L. and Wang, S.J. (2000), "Seismic behaviour of RC columns with various tie configurations", *Journal of Structural Engineering*, Vol. 126, pp. 1122-1130

Molina, J., Verzeletti, G., Magonette, G., Buchet, P. and Géradin, M. (1999), "Bi-directional pseudodynamic test of a full size three-storey building", *Earthquake Engineering and Structural Dynamics*, Vol. 28, pp. 1541-1566

Mosalam, K.M., White, R.N. and Ayala, G. (1998), "Response of infilled frames using pseudo-dynamic experimentation", *Earthquake Engineering and Structural Dynamics*, Vol. 27, pp. 589-608

Parra-Montesinos, G. and Wight, J.K. (2000), "Seismic response of exterior RC column-to-steel beam connections", *Journal of Structural Engineering*, Vol. 126, pp. 1113-1121

Saatcioglu, M., Alsiwat, J.M. and Ozcebe, G. (1992), "Hysteretic behaviour of anchorage slip in R/C members", *Journal of Structural Engineering*, Vol. 118, pp. 2439-2458

Schneider, S.P. and Roeder, C.W. (1994), "An inelastic substructure technique for the pseudodynamic test method", *Earthquake Engineering and Structural Dynamics*, Vol. 23, pp. 761-775

Shing, P.B., Nakashima, M. and Bursi, O.S. (1996), "Application of pseudodynamic test method to structural research", *Earthquake Spectra*, Vol. 12, pp. 29-56

Thiele, K (2000), "Pseudodynamische Versuche an Tragwerken mit grossen Steifigkeitsänderungen und mehreren Freiheitsgraden, *Internal publication, Institut für Baustatik und Konstruktion (IBK), Bericht Nr. 253*, Birkhäuser Verlag, Basel, Aug. 2000

Thiele, K, Wenk, T. and Bachmann, H. (2000), "Versuche an Stahlbetontragwänden unter pseudodynamischer Einwirkung, *Internal publication, Institut für Baustatik und Konstruktion (IBK), Bericht Nr. 257*, Birkhäuser Verlag, Basel, Nov. 2000

Usami, T. and Kumar, S. (1996), "Damage evaluation in steel box columns by pseudodynamic tests", *Journal of Structural Engineering*, Vol. 122, pp. 635-642

Wagg, D.J. and Stoten, D.P. (2001), "Substructuring of dynamic systems via the adaptive minimal control synthesis algorithm", *Earthquake Engineering and Structural Dynamics*, Vol. 30, pp. 865-877

Appendix A1:

STABILITY OF THE NEWMARK EXPLICIT – INTEGRAL FORM w/o RECALCULATED v_{n+1}

Note: $s = \int d \, \mathrm{dt}$

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t(v_n + v_{n+1}) = d_n + \Delta t v_n + \frac{1}{2}\Delta t \Delta v_{n+1}$$

$$v_{n+1} = v_n + M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta \bar{r}_{n+1}\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)$$

$$v_{n+1} = v_n + M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta \bar{r}_{n+1}\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta t}{2}(r_n + r_{n+1})\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta t k}{2}\left(d_n + d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)\right)\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n + \frac{1}{2}\Delta t^2 \frac{k}{m}\left(-\Delta t d_n - \frac{1}{2}(\Delta t)^2 v_n\right)$$

$$\Delta t^2 v_{n+1} = \Delta t^2 v_n + \Delta t^2 \frac{k}{m}\left(-\frac{\Delta t}{2}\left(d_n + d_n + \Delta t v_n + \frac{1}{2}\Delta t \frac{k}{m}\left(-\Delta t d_n - \frac{1}{2}(\Delta t)^2 v_n\right)\right)\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n + \frac{1}{2}\Omega^2\left(-\Delta t d_n - \frac{1}{2}(\Delta t)^2 v_n\right)$$

$$\Delta t^2 v_{n+1} = \Delta t^2 v_n + \Omega^2\left(-\Delta t d_n - \frac{1}{2}\Delta t^2 v_n - \frac{1}{4}\Omega^2\left(-\Delta t d_n - \frac{1}{2}(\Delta t)^2 v_n\right)\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n - \frac{1}{2}\Omega^2 \Delta t d_n - \frac{1}{4}\Omega^2 (\Delta t)^2 v_n$$

$$\Delta t^2 v_{n+1} = \Delta t^2 v_n - \Omega^2 \Delta t d_n - \frac{1}{2}\Omega^2 \Delta t^2 v_n + \frac{1}{4}\Omega^4 \Delta t d_n + \frac{1}{8}\Omega^4 (\Delta t)^2 v_n$$

****

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} s_{n+1} \\ \Delta t d_{n+1} \\ (\Delta t)^2 v_{n+1} \end{Bmatrix} = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2} & 1-\frac{\Omega^2}{4} \\ 0 & -\Omega^2 + \frac{\Omega^4}{4} & 1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8} \end{bmatrix} \begin{Bmatrix} s_n \\ t d_n \\ (\Delta t)^2 v_n \end{Bmatrix}$$

****

$$\begin{Bmatrix} s_{n+1} \\ \Delta t d_{n+1} \\ (\Delta t)^2 v_{n+1} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2} & 1-\frac{\Omega^2}{4} \\ 0 & -\Omega^2 + \frac{\Omega^4}{4} & 1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8} \end{bmatrix} \begin{Bmatrix} s_n \\ t d_n \\ (\Delta t)^2 v_n \end{Bmatrix}$$

****

$$\begin{Bmatrix} s_{n+1} \\ \Delta t d_{n+1} \\ (\Delta t)^2 v_{n+1} \end{Bmatrix} = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2} & 1-\frac{\Omega^2}{4} \\ 0 & -\Omega^2 + \frac{\Omega^4}{4} & 1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8} \end{bmatrix} \begin{Bmatrix} s_n \\ t d_n \\ (\Delta t)^2 v_n \end{Bmatrix}$$

****

$$A = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2} & 1-\frac{\Omega^2}{4} \\ 0 & -\Omega^2 + \frac{\Omega^4}{4} & 1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8} \end{bmatrix}$$

****

$$|A-\lambda| = \begin{vmatrix} 1-\lambda & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2}-\lambda & 1-\frac{\Omega^2}{4} \\ 0 & -\Omega^2+\frac{\Omega^4}{4} & 1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8}-\lambda \end{vmatrix}$$

****

$$|A-\lambda| = (1-\lambda)\left\{\left(1-\frac{\Omega^2}{2}-\lambda\right)\left(1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8}-\lambda\right)-\left(1-\frac{\Omega^2}{4}\right)\left(-\Omega^2+\frac{\Omega^4}{4}\right)\right\}$$

****

$$|A-\lambda| = (1-\lambda)\left\{1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8}-\lambda-\frac{\Omega^2}{2}\left(1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8}-\lambda\right)-\lambda\left(1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8}-\lambda\right)-\left(-\Omega^2+\frac{\Omega^4}{4}\right)+\frac{\Omega^2}{4}\left(-\Omega^2+\frac{\Omega^4}{4}\right)\right\}$$

****

$$|A-\lambda| = (1-\lambda)$$
$$\left\{1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8}-\lambda-\frac{\Omega^2}{2}+\frac{\Omega^4}{4}-\frac{\Omega^6}{16}+\frac{\lambda\Omega^2}{2}-\lambda+\frac{\lambda\Omega^2}{2}-\frac{\lambda\Omega^4}{8}+\lambda^2+\Omega^2-\frac{\Omega^4}{4}-\frac{\Omega^4}{4}+\frac{\Omega^6}{16}\right\}$$

$$|A-\lambda| = (1-\lambda)$$
$$\left\{\lambda^2-2\lambda+\lambda\Omega^2-\frac{\lambda\Omega^4}{8}-\frac{\Omega^4}{8}-1\right\}$$

****

$$|A-\lambda| = (1-\lambda)\left\{\lambda^2+\lambda\left(-2+\Omega^2-\frac{\Omega^4}{8}\right)-\frac{\Omega^4}{8}-1\right\}$$

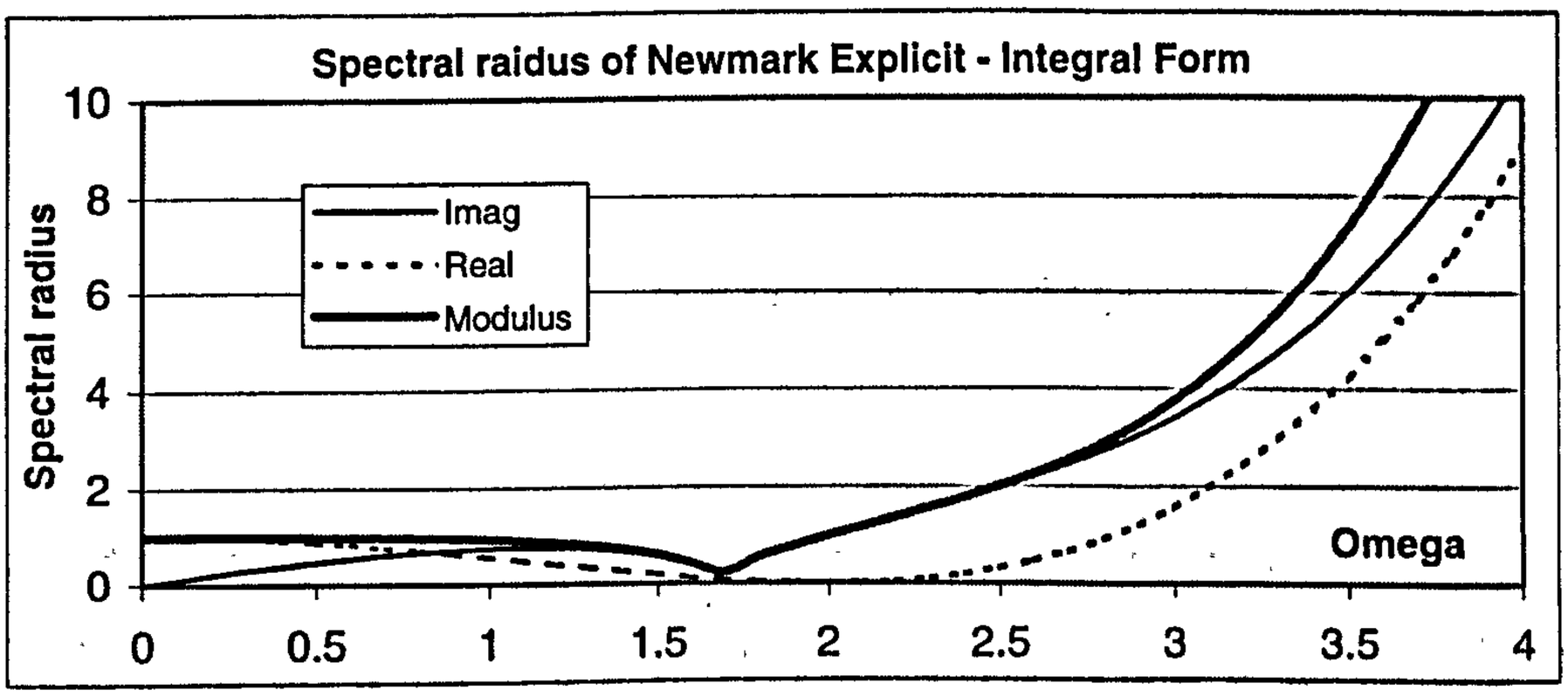
$\lambda_1=1$

$$\lambda_{2,3} = \frac{-b\pm\sqrt{b^2-4ac}}{2a}$$

$$\lambda_{2,3} = \frac{-\left(-2+\Omega^2-\frac{\Omega^4}{8}\right)\pm\sqrt{\left(-2+\Omega^2-\frac{\Omega^4}{8}\right)^2-4\left(-\frac{\Omega^4}{8}-1\right)}}{2}$$

$$\lambda_{2,3} = \frac{\left(2-\Omega^2+\frac{\Omega^4}{8}\right) \pm \sqrt{-4-4\Omega^2+\frac{3\Omega^4}{2}-\frac{\Omega^6}{4}+\frac{\Omega^8}{64}+\frac{\Omega^4}{2}+4}}{2}$$

$$\lambda_{2,3} = \frac{\left(2-\Omega^2+\frac{\Omega^4}{8}\right) \pm \sqrt{-4\Omega^2+2\Omega^4-\frac{\Omega^6}{4}+\frac{\Omega^8}{64}}}{2}$$



Appendix A2:

STABILITY OF THE NEWMARK EXPLICIT – INTEGRAL FORM w/o RECALCULATED v_{n+1}, ALTERNATIVE $\int r_{n+1} dt$

Note: $s = \int d \, dt$

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t(v_n + v_{n+1}) = d_n + \Delta t v_n + \frac{1}{2}\Delta t \Delta v_{n+1}$$

$$v_{n+1} = v_n + M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta \bar{r}_{n+1}\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)$$

$$v_{n+1} = v_n + M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta \bar{r}_{n+1}\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta t}{2}(r_n + r_{n+1})\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta t k}{2}(d_n + d_{n+1})\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n + \frac{1}{2}\Delta t^2 \frac{k}{m}\left(-\Delta t d_n - \frac{1}{2}(\Delta t)^2 v_n\right)$$

$$\Delta t^2 v_{n+1} = \Delta t^2 v_n + \Delta t^2 \frac{k}{m}\left(-\frac{\Delta t}{2}(d_n + d_{n+1})\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n + \frac{1}{2}\Omega^2\left(-\Delta t d_n - \frac{1}{2}(\Delta t)^2 v_n\right)$$

$$\Delta t^2 v_{n+1} = \Delta t^2 v_n + \Omega^2\left(-\frac{\Delta t}{2}(d_n + d_{n+1})\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n - \frac{1}{2}\Omega^2 \Delta t d_n - \frac{1}{4}\Omega^2 (\Delta t)^2 v_n$$

$$\Delta t^2 v_{n+1} = \Delta t^2 v_n - \Omega^2 \frac{\Delta t d_n}{2} - \Omega^2 \frac{\Delta t d_{n+1}}{2}$$

****

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \Omega^2/2 & 1 \end{bmatrix} \begin{Bmatrix} s_{n+1} \\ \Delta t d_{n+1} \\ (\Delta t)^2 v_{n+1} \end{Bmatrix} = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2} & 1-\frac{\Omega^2}{4} \\ 0 & -\Omega^2/2 & 1 \end{bmatrix} \begin{Bmatrix} s_n \\ t d_n \\ (\Delta t)^2 v_n \end{Bmatrix}$$

****

$$\begin{Bmatrix} s_{n+1} \\ \Delta t d_{n+1} \\ (\Delta t)^2 v_{n+1} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\Omega^2/2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2} & 1-\frac{\Omega^2}{4} \\ 0 & -\Omega^2/2 & 1 \end{bmatrix} \begin{Bmatrix} s_n \\ t d_n \\ (\Delta t)^2 v_n \end{Bmatrix}$$

****

$$\begin{Bmatrix} s_{n+1} \\ \Delta t d_{n+1} \\ (\Delta t)^2 v_{n+1} \end{Bmatrix} = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2} & 1-\frac{\Omega^2}{4} \\ 0 & -\frac{\Omega^2}{2}\left(1-\frac{\Omega^2}{2}\right)-\frac{\Omega^2}{2} & 1-\frac{\Omega^2}{2}\left(1-\frac{\Omega^2}{4}\right) \end{bmatrix} \begin{Bmatrix} s_n \\ t d_n \\ (\Delta t)^2 v_n \end{Bmatrix}$$

****

$$A = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1-\frac{\Omega^2}{2} & 1-\frac{\Omega^2}{4} \\ 0 & -\Omega^2+\frac{\Omega^4}{4} & 1-\frac{\Omega^2}{2}+\frac{\Omega^4}{8} \end{bmatrix}$$

Appendix B:

STABILITY OF THE NEWMARK EXPLICIT – INTEGRAL FORM w/ RECALCULATED v_{n+1}

Note: $s = \int d \, dt$

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t(v_n + v_{n+1}) = d_n + \frac{1}{2}\Delta t v_n + \frac{1}{2}\Delta t v_{n+1}$$

$$v_{n+1} = v_n + M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta \bar{r}_{n+1}\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t v_n + \frac{1}{2}\Delta t v_{n+1}$$

$$v_{n+1} = v_n + M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta \bar{r}_{n+1}\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t v_n + \frac{1}{2}\Delta t v_{n+1}$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta t}{2}\left(r_n + r^*_{n+1}\right)\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t v_n + \frac{1}{2}\Delta t v_{n+1}$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta t k}{2}\left(d_n + d^*_{n+1}\right)\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t v_n + \frac{1}{2}\Delta t v_{n+1}$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta t k}{2}\left(2d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d^*_n - \frac{k}{2}(\Delta t)^2 v_n\right)\right)\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t v_n + \frac{1}{2}\Delta t v_{n+1}$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta t k}{2}\left(2d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k\left(d_{n-1} + \Delta t v_{n-1} + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d^*_{n-1} - \frac{k}{2}(\Delta t)^2 v_{n-1} \, etc....\right)\right) - \frac{k}{2}(\Delta t)^2 v_n\right)\right)\right)$$

This method is implemented using the restoring force resulting from the explicit displacement predictor, and not the implicit corrected displacement value. The reason for this is that the restoring force resulting from the implicit displacement target, calculated after predictor step has been imposed, cannot be obtained pseudodynamically without a double step implementation. In other words, r_n is unavailable at the start of the step as the corresponding d_n has not been imposed on the structure. This is due to the fact that d_n is recalculated after completion of the step, and the restoring force caused by it thus unknown. The method has to be implemented using the restoring force at the end of the previous step unless a kind of double step is utilised. This would require the recalculated d_{n+1} to be imposed separately once calculations have finished and the applicable restoring force then measured. Using the r_n from the explicit target instead results in some negative damping and a reduced stability limit, but the recurrent relationship between the state vectors cannot be expressed in terms of time=t_n and t_{n+1} only.

Appendix C:

STABILITY OF THE NEWMARK IMPLICIT – INTEGRAL FORM

Note: $s = \int d \, dt$

$$s_{n+1} = s_n + \Delta t d_n + \left(\frac{1}{2} - \beta\right)(\Delta t)^2 v_n + \beta(\Delta t)^2 v_{n+1}$$
$$d_{n+1} = d_n + (1-\gamma)\Delta t v_n + \gamma \Delta t v_{n+1}$$
$$v_{n+1} = v_n + M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta \bar{r}_{n+1}\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \left(\frac{1}{2} - \beta\right)(\Delta t)^2 v_n + \beta(\Delta t)^2 \left(v_n + \Delta v^*_{n+1}\right)$$
$$d_{n+1} = d_n + (1-\gamma)\Delta t v_n + \gamma \Delta t \left(v_n + \Delta v^*_{n+1}\right)$$
$$v_{n+1} = v_n + M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta \bar{r}_{n+1}\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n + \beta(\Delta t)^2 M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta t r_n - \frac{k'}{2}(\Delta t)^2 v_n\right)\left(1 + M^{-1}\frac{k'}{4}(\Delta t)^2\right)^{-1}$$
$$d_{n+1} = d_n + \Delta t v_n + \gamma \Delta t M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta t r_n - \frac{k'}{2}(\Delta t)^2 v_n\right)\left(1 + M^{-1}\frac{k'}{4}(\Delta t)^2\right)^{-1}$$
$$v_{n+1} = v_n + M^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - k' \Delta s_{n+1}\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n + \beta(\Delta t)^2 M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)\left(1 + M^{-1}\frac{k}{4}(\Delta t)^2\right)^{-1}$$
$$d_{n+1} = d_n + \Delta t v_n + \gamma \Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)\left(1 + M^{-1}\frac{k}{4}(\Delta t)^2\right)^{-1}$$
$$v_{n+1} = v_n + M^{-1}\left(-k\Delta s_{n+1}\right)$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n + \beta(\Delta t)^2\left(-\frac{\Delta t k d_n}{m} - \frac{k}{2m}(\Delta t)^2 v_n\right)\left(1 + \frac{k}{4m}(\Delta t)^2\right)^{-1}$$
$$d_{n+1} = d_n + \Delta t v_n + \gamma \Delta t\left(-\frac{\Delta t k d_n}{m} - \frac{k}{2m}(\Delta t)^2 v_n\right)\left(1 + \frac{k}{4m}(\Delta t)^2\right)^{-1}$$
$$v_{n+1} = v_n - \frac{k(s_{n+1} - s_n)}{m}$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n + \beta \Delta t \left(-\Omega^2 d_n - \frac{\Omega^2}{2} \Delta t v_n \right) \left(1 + \frac{\Omega^2}{4} \right)^{-1}$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n + \gamma \Delta t \left(-\Omega^2 d_n - \frac{\Omega^2}{2} \Delta t v_n \right) \left(1 + \frac{\Omega^2}{4} \right)^{-1}$$

$$(\Delta t)^2 v_{n+1} = (\Delta t)^2 v_n - \Omega^2 s_{n+1} + \Omega^2 s_n$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n - \beta \Omega^2 \Delta t d_n \left(1 + \frac{\Omega^2}{4} \right)^{-1} - \frac{1}{2} \beta \Omega^2 (\Delta t)^2 v_n \left(1 + \frac{\Omega^2}{4} \right)^{-1}$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n - \gamma \Omega^2 \Delta t d_n \left(1 + \frac{\Omega^2}{4} \right)^{-1} - \frac{1}{2} \gamma \Omega^2 (\Delta t)^2 v_n \left(1 + \frac{\Omega^2}{4} \right)^{-1}$$

$$(\Delta t)^2 v_{n+1} = (\Delta t)^2 v_n - \Omega^2 s_{n+1} + \Omega^2 s_n$$

****

$$s_{n+1} = s_n + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n - \beta \frac{\Omega^2}{1+\Omega^2/4} \Delta t d_n - \frac{1}{2} \beta \frac{\Omega^2}{1+\Omega^2/4} (\Delta t)^2 v_n$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n - \gamma \frac{\Omega^2}{1+\Omega^2/4} \Delta t d_n - \frac{1}{2} \gamma \frac{\Omega^2}{1+\Omega^2/4} (\Delta t)^2 v_n$$

$$\Omega^2 s_{n+1} + (\Delta t)^2 v_{n+1} = \Omega^2 s_n + (\Delta t)^2 v_n$$

****

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \Omega^2 & 0 & 1 \end{bmatrix} \begin{Bmatrix} s_{n+1} \\ \Delta t d_{n+1} \\ (\Delta t)^2 v_{n+1} \end{Bmatrix} = \begin{bmatrix} 1 & 1-\beta \frac{\Omega^2}{1+\Omega^2/4} & \frac{1}{2} - \frac{1}{2} \beta \frac{\Omega^2}{1+\Omega^2/4} \\ 0 & 1-\gamma \frac{\Omega^2}{1+\Omega^2/4} & 1-\frac{1}{2} \gamma \frac{\Omega^2}{1+\Omega^2/4} \\ \Omega^2 & 0 & 1 \end{bmatrix} \begin{Bmatrix} s_n \\ t d_n \\ (\Delta t)^2 v_n \end{Bmatrix}$$

****

$$\begin{Bmatrix} s_{n+1} \\ \Delta t d_{n+1} \\ (\Delta t)^2 v_{n+1} \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\Omega^2 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1-\beta \frac{\Omega^2}{1+\Omega^2/4} & \frac{1}{2} - \frac{1}{2} \beta \frac{\Omega^2}{1+\Omega^2/4} \\ 0 & 1-\gamma \frac{\Omega^2}{1+\Omega^2/4} & 1-\frac{1}{2} \gamma \frac{\Omega^2}{1+\Omega^2/4} \\ \Omega^2 & 0 & 1 \end{bmatrix} \begin{Bmatrix} s_n \\ t d_n \\ (\Delta t)^2 v_n \end{Bmatrix}$$

****

$$\begin{Bmatrix} s_{n+1} \\ \Delta t d_{n+1} \\ (\Delta t)^2 v_{n+1} \end{Bmatrix} = \begin{bmatrix} 1 & 1-\beta\frac{\Omega^2}{1+\Omega^2/4} & \frac{1}{2}-\frac{1}{2}\beta\frac{\Omega^2}{1+\Omega^2/4} \\ 0 & 1-\gamma\frac{\Omega^2}{1+\Omega^2/4} & 1-\frac{1}{2}\gamma\frac{\Omega^2}{1+\Omega^2/4} \\ 0 & -\Omega^2\left(1-\beta\frac{\Omega^2}{1+\Omega^2/4}\right) & 1-\Omega^2\left(\frac{1}{2}-\frac{1}{2}\beta\frac{\Omega^2}{1+\Omega^2/4}\right) \end{bmatrix} \begin{Bmatrix} s_n \\ td_n \\ (\Delta t)^2 v_n \end{Bmatrix}$$

****

$$A = \begin{bmatrix} 1 & 1-\frac{\Omega^2}{4+\Omega^2} & \frac{1}{2}-\frac{\Omega^2}{8+2\Omega^2} \\ 0 & 1-\frac{\Omega^2}{2+\Omega^2/2} & 1-\frac{\Omega^2}{4+\Omega^2} \\ 0 & -\Omega^2\left(1-\frac{\Omega^2}{4+\Omega^2}\right) & 1-\Omega^2\left(\frac{1}{2}-\frac{\Omega^2}{8+2\Omega^2}\right) \end{bmatrix}$$

****

$$|A-\lambda| = \begin{vmatrix} 1-\lambda & 1-\frac{\Omega^2}{4+\Omega^2} & \frac{1}{2}-\frac{\Omega^2}{8+2\Omega^2} \\ 0 & 1-\frac{\Omega^2}{2+\Omega^2/2}-\lambda & 1-\frac{\Omega^2}{4+\Omega^2} \\ 0 & -\Omega^2\left(1-\frac{\Omega^2}{4+\Omega^2}\right) & 1-\Omega^2\left(\frac{1}{2}-\frac{\Omega^2}{8+2\Omega^2}\right)-\lambda \end{vmatrix}$$

****

$$|A-\lambda| = (1-\lambda)\left\{\left(1-\frac{\Omega^2}{2+\Omega^2/2}-\lambda\right)\left(1-\frac{\Omega^2}{2}\left(1-\frac{\Omega^2}{4+\Omega^2}\right)-\lambda\right)-\left(1-\frac{\Omega^2}{4+\Omega^2}\right)\left(-\Omega^2\left(1-\frac{\Omega^2}{4+\Omega^2}\right)\right)\right\}$$

****

$$|A-\lambda| = (1-\lambda)\left\{\left(1-\frac{2\Omega^2}{4+\Omega^2}-\lambda\right)\left(1-\frac{\Omega^2}{2}+\frac{1}{2}\frac{\Omega^4}{4+\Omega^2}-\lambda\right)-\left(1-\frac{\Omega^2}{4+\Omega^2}\right)\left(-\Omega^2+\frac{\Omega^4}{4+\Omega^2}\right)\right\}$$

****

$$|A-\lambda| = (1-\lambda)$$
$$\left\{\left(1-\frac{\Omega^2}{2}+\frac{1}{2}\frac{\Omega^4}{4+\Omega^2}-\lambda-\frac{2\Omega^2}{4+\Omega^2}+\frac{\Omega^4}{4+\Omega^2}-\left(\frac{\Omega^2}{4+\Omega^2}\right)\left(\frac{\Omega^4}{4+\Omega^2}\right)+\frac{2\Omega^2}{4+\Omega^2}\lambda-\lambda+\frac{\lambda\Omega^2}{2}-\frac{\lambda}{2}\frac{\Omega^4}{4+\Omega^2}+\lambda^2\right)\right\}$$
$$\left\{-\left(-\Omega^2+\frac{\Omega^4}{4+\Omega^2}+\frac{\Omega^4}{4+\Omega^2}-\left(\frac{\Omega^2}{4+\Omega^2}\right)\left(\frac{\Omega^4}{4+\Omega^2}\right)\right)\right\}$$

$$\xi^2 = \frac{\Omega^2}{1+\frac{1}{4}\Omega^2}$$

••••••••••••

$$|A-\lambda| = (1-\lambda)$$

$$\left\{\left(1-\frac{\Omega^2}{2}+\frac{\Omega^2}{8}\xi^2-\lambda-\frac{1}{2}\xi^2+\frac{\Omega^2}{4}\xi^2-\frac{1}{4}\xi^2\frac{\Omega^2}{4}\xi^2+\frac{1}{2}\xi^2\lambda-\lambda+\frac{\lambda\Omega^2}{2}-\lambda\frac{\Omega^2}{8}\xi^2+\lambda^2\right)\right\}=0$$

$$\left\{-\left(-\Omega^2+\frac{\Omega^2}{4}\xi^2+\frac{\Omega^2}{4}\xi^2-\frac{1}{4}\xi^2\frac{\Omega^2}{4}\xi^2\right)\right.$$

••••••••••

$$=(1-\lambda)\left\{\left(\lambda^2+\left(-2+\frac{1}{2}\xi^2+\frac{\Omega^2}{2}-\frac{\Omega^2}{8}\xi^2\right)\lambda+1+\frac{\Omega^2}{2}-\frac{1}{8}\Omega^2\xi^2-\frac{1}{2}\xi^2\right)\right\}=0$$

••••••••••••

$$=(1-\lambda)\left(\lambda^2+\left\{\left(-2+\frac{1}{2}\Omega^2\right)\left(1-\frac{1}{4}\xi^2\right)\right\}\lambda+1+\frac{\Omega^2}{2}-\frac{\Omega^2\xi^2}{8}-\frac{1}{2}\xi^2\right)=0$$

••••••••••••

$$\lambda_1=1$$

$$\lambda_{2,3}=\frac{-b\pm\sqrt{b^2-4ac}}{2a}$$

••••••••••••

$$\sqrt{b^2-4ac}=\sqrt{\left\{\left(-2+\frac{1}{2}\Omega^2\right)\left(1-\frac{1}{4}\xi^2\right)\right\}^2-4\left(1+\frac{\Omega^2}{2}-\frac{\Omega^2\xi^2}{8}-\frac{1}{2}\xi^2\right)}$$

••••••••••••

$$\sqrt{b^2-4ac}=\sqrt{\left(-2+\frac{1}{2}\xi^2+\frac{\Omega^2}{2}-\frac{\Omega^2\xi^2}{8}\right)^2-4-2\Omega^2+\frac{1}{2}\Omega^2\xi^2+2\xi^2}$$

••••••••••••

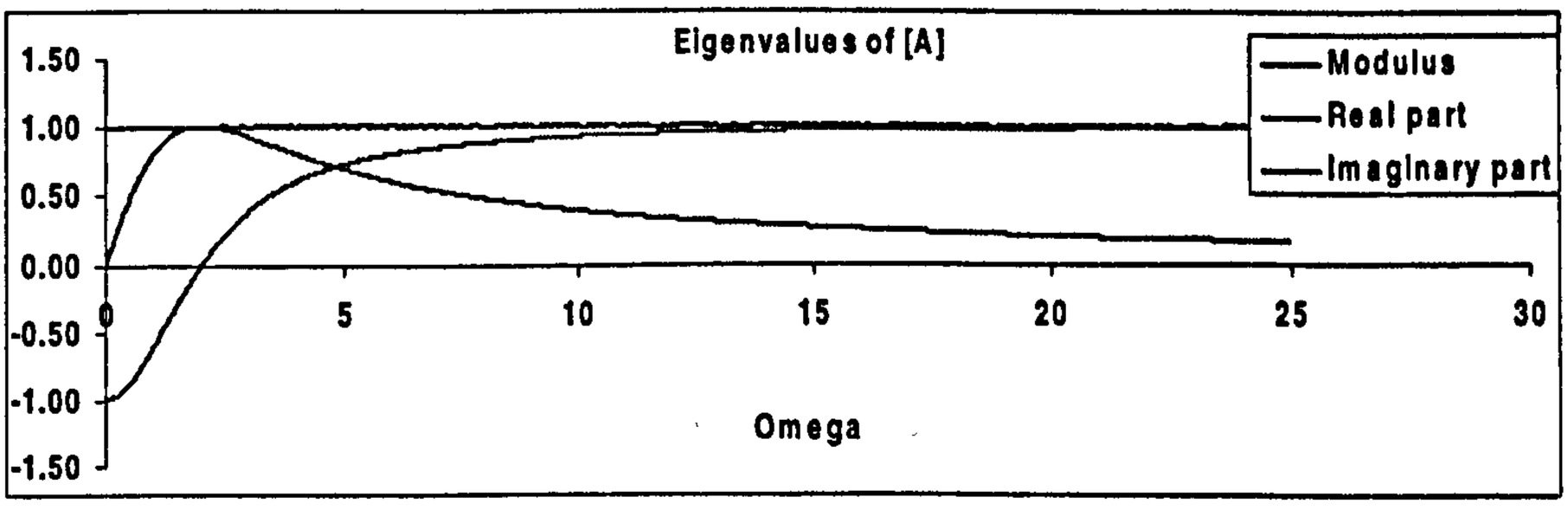
$$\sqrt{b^2-4ac}=\sqrt{-4\Omega^2+\frac{1}{4}\xi^4+\frac{1}{4}\Omega^4+\frac{3\xi^2\Omega^2}{2}-\frac{\Omega^4\xi^2}{8}-\frac{\Omega^2\xi^4}{8}+\frac{\Omega^4\xi^4}{64}}$$

••••••••••••

$$\lambda_{2,3} = \frac{\left(-2+\frac{\xi^2}{2}+\frac{\Omega^2}{2}-\frac{\Omega^2\xi^2}{8}\right) \pm \sqrt{-4\Omega^2+\frac{1}{4}\xi^4+\frac{1}{4}\Omega^4+\frac{3\xi^2\Omega^2}{2}-\frac{\Omega^4\xi^2}{8}-\frac{\Omega^2\xi^4}{8}+\frac{\Omega^4\xi^4}{64}}}{2}$$

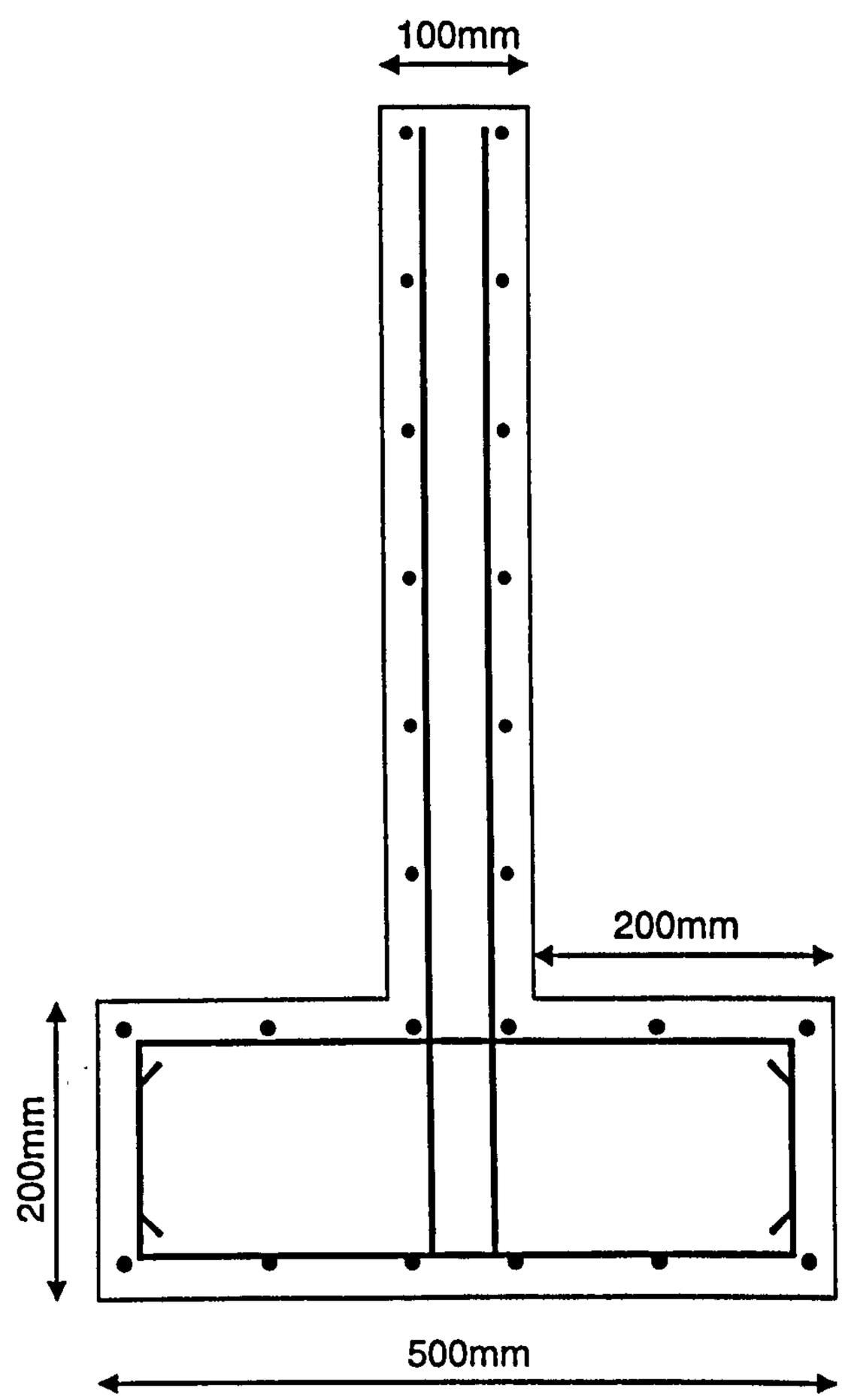
****

$\lambda_{2,3} = 1$ *for ALL* Ω



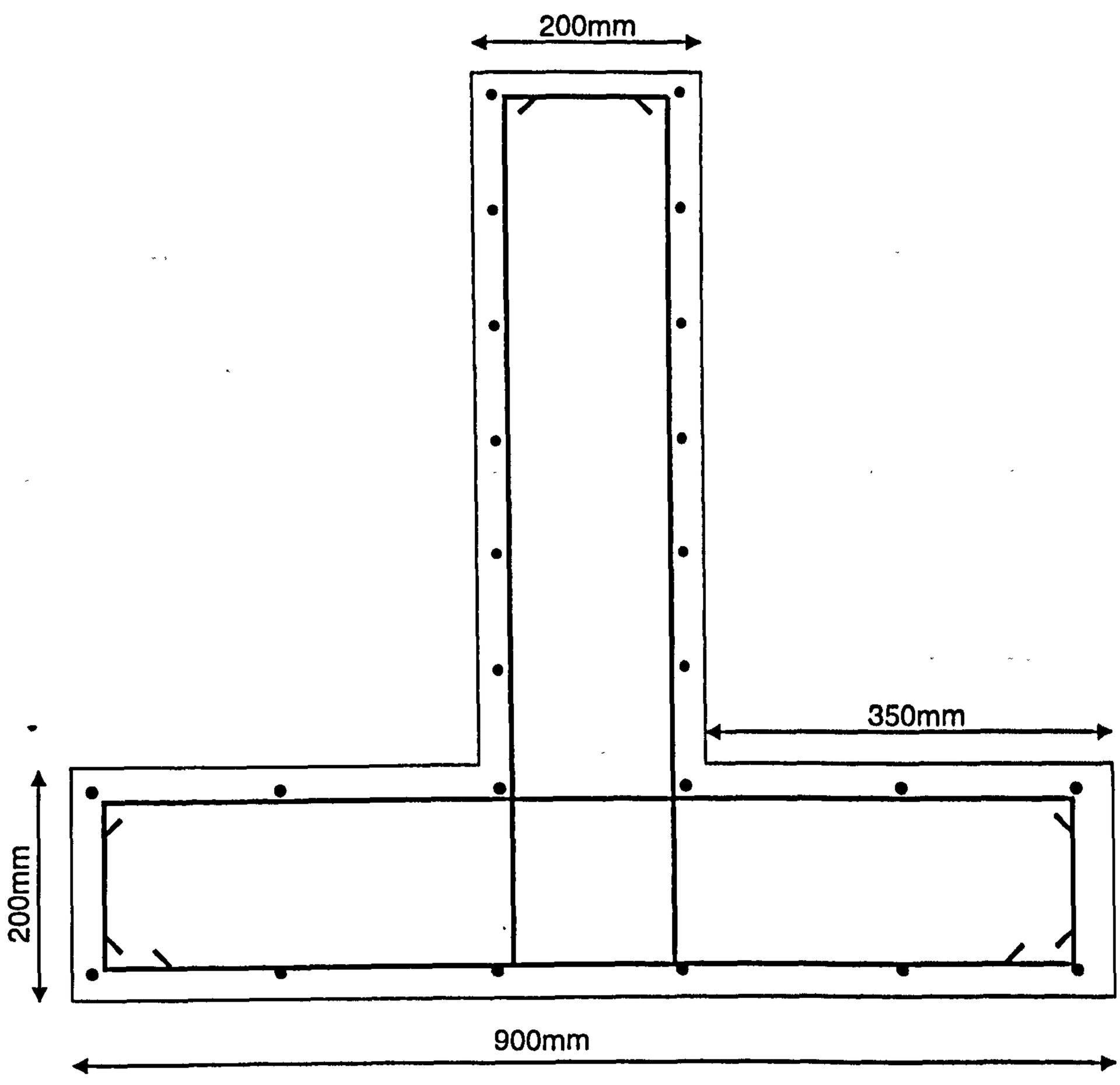
Appendix D:

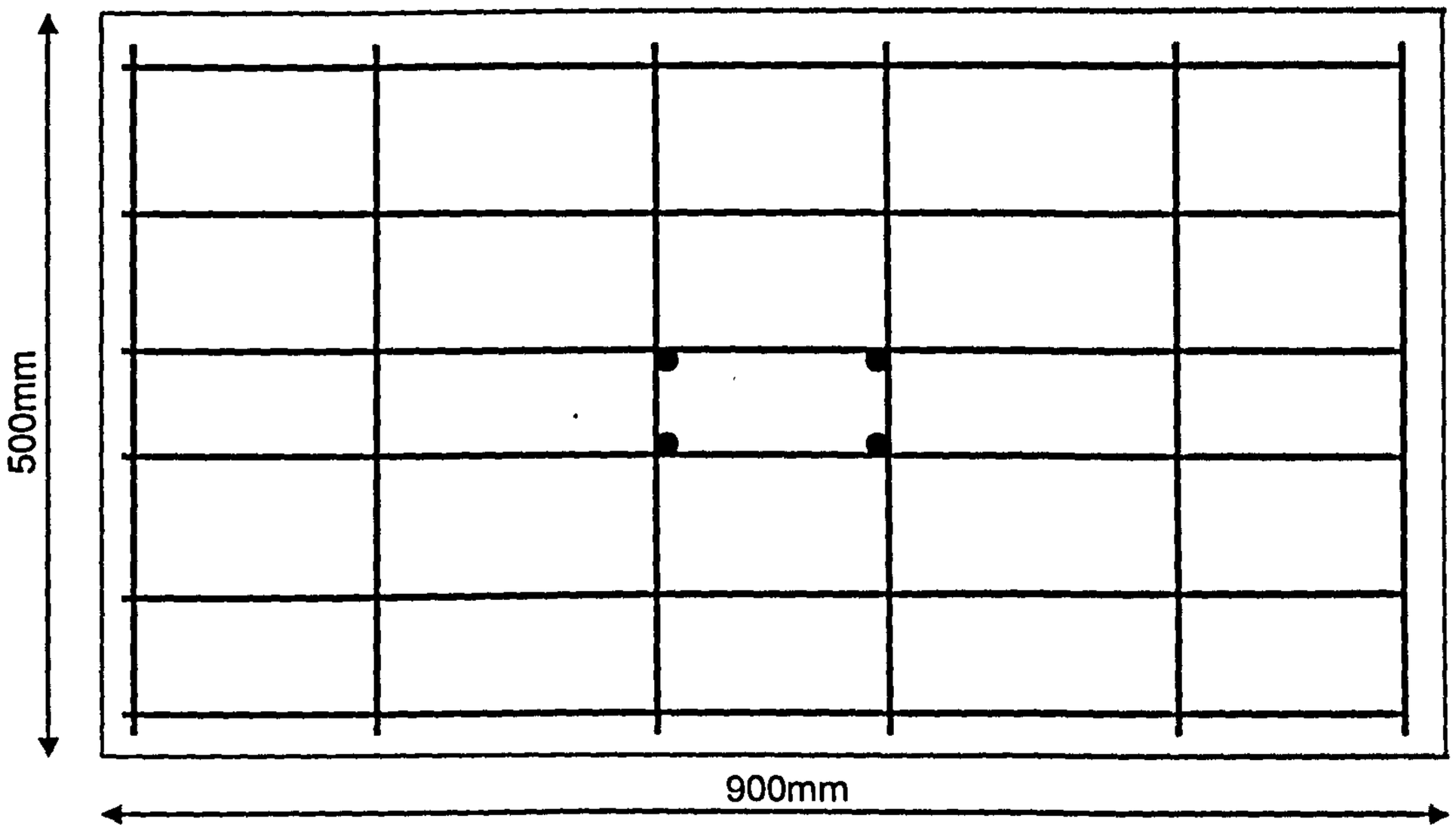
SPECIMEN DRAWINGS



Notes:

- All steel high yield (f_y=460N/mm^2) apart from links in column, which are mild steel (f_y=250N/mm^2).
- All steel 12mm Ø ribbed bars, apart from links in column, which are 6mm Ø straight bars.
- Concrete cube strength, f_{cu}=30N/mm^2
- Maximum aggregate size, d_{max}=10mm
- Cover: 20mm to links


200mm
350mm
200mm
900mm


500mm
900mm

Appendix E

ACME 2000 CONFERENCE PAPER

COMPUTATIONAL ASPECTS OF PSEUDO-DYNAMIC TESTING

W. Algaard, T.J.A. Agar, N. Bicanic
Dept. of Civil Engineering, University of Glasgow, Glasgow, G12 8LT

INTRODUCTION

Pseudodynamic testing is a combined computational/experimental technique for dynamic systems where inertia and damping are modelled computationally while the non-linear restoring forces are measured experimentally. The following equation of motion describes a non-linear system subject to an external force:

$$M\ddot{x} + C\dot{x} + r(x) = f \tag{1}$$

where M is the mass matrix, C the viscous damping matrix, $r(x)$ the non-linear structural restoring force and f the applied force. The pseudodynamic testing technique uses a time-stepping scheme to integrate equation (1) to obtain the response of the system. For each cycle, a displacement increment is computed, then imposed on the structure by servo-hydraulic actuators. Measurement of the elastic restoring force follows, which is then utilised when computing the next displacement increment.

Several time-stepping algorithms have been proposed for application to pseudodynamic testing. The majority of these are explicit due to the fact that the non-linear structural restoring forces at the end of any time-step are unknown and displacement iterations in pseudodynamic tests are undesirable. The so-called *integral form* of the Newmark Explicit method, proposed by Chang et al. [1], relies on integrating the second-order equation of motion (1) once with respect to time. It is believed that this method may exhibit improved abilities to model rapidly varying loads and stiffness. Implementation of the integral form into the pseudodynamic testing framework is however complicated by the fact that certain stiffness related terms become implicit. The possibility of further refining the method by modifying some of its aspects has been considered. Furthermore, a system for implementation into a pseudodynamic test is described.

FORMULATION

By integrating equation (1) with respect to time and expressing it in incremental form, the equation of motion for an SDOF was formulated by Chang et al. [1] as:

$$m\Delta v_{n+1} + c\Delta d_{n+1} + \Delta \bar{r}_{n+1} = \Delta \bar{f}_{n+1} \tag{2}$$

where v represents velocity, d displacement, $\bar{r}$ time-integral of restoring force, $\bar{f}$ time-integral of applied force and Δ indicates the change of variables over one time-step. This formulation can also be solved by a modified Newmark Explicit method, Chang et al. [1]:

$$\Delta s_{n+1} = (\Delta t) d_n + \frac{1}{2}(\Delta t)^2 v_n \tag{3a}$$

$$\Delta d_{n+1} = \frac{1}{2}(\Delta t)(v_n + v_{n+1}) \tag{3b}$$

where s is the time-integral of displacement. The advantage of this procedure, as compared with conventional methods is that the time integral of restoring force and external excitation is used rather than the respective values at the start of each time step. This will avoid linearisation errors of highly non-linear restoring forces and better accommodate rapidly varying dynamic loads.

The solution procedure outlined by Chang et al. suggests a predicted and experimentally imposed displacement for an SDOF system as:

$$\Delta d_{n+1} = \Delta t v_n + \frac{\Delta t}{2}\left(m + \frac{\Delta t}{2}c\right)^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right) \tag{4}$$

where Δt is the size of the time-step and k_0 the initial stiffness replacing an otherwise implicit tangent stiffness term. This expression builds on the following prediction of the integral of the restoring force over that time-step:

$$\Delta \bar{r}_{n+1} = (\Delta t) r_n + \frac{k}{2}(\Delta t)^2 v_n \tag{5}$$

where k is the tangent stiffness of the system. In turn, equation 5 is based on equation (3a). Following the displacement step in equation (4), experimentally measured values of the restoring force become available and replace equation (5). Equation (2) may then be resolved for Δv_{n+1} using constant average velocity to yield a corrected term based on the measured, rather than the predicted, value of $\Delta \bar{r}_{n+1}$.

The algorithm creates moderate to strong numerical damping depending on the strict interpretation and computation of d_{n+1}. This is due to the fact that the measured restoring force used when resolving equation (2) originates from the predicted displacement, equation (4), rather than the applicable constant average velocity as used in equation (3b). These are not equivalent, as the predicted restoring force is based on equation (3a), which does not contain the required implicit term. When compared to an incremental, integrated form of the implicit Newmark method:

$$\Delta s_{n+1} = (\Delta t) d_n + (\Delta t)^2\left(\frac{1}{2} - \beta\right) v_n + (\Delta t)^2 \beta v_{n+1} \tag{6a}$$

$$\Delta s_{n+1} = (\Delta t) d_n + (\Delta t)^2\left(\frac{1}{2} - \beta\right) v_n + (\Delta t)^2 \beta (v_n + \Delta v_{n+1}) \tag{6b}$$

$$\Delta s_{n+1} = (\Delta t) d_n + (\Delta t)^2 \frac{1}{2} v_n + (\Delta t)^2 \beta \Delta v_{n+1} \tag{6c}$$

The procedure yields a predicted time-integral of restoring force as:

$$\Delta \bar{r}_{n+1} = (\Delta t) r_n + \frac{k}{2}(\Delta t)^2 v_n + k(\Delta t)^2 \beta \Delta v_{n+1} \tag{7}$$

The last term of equation (7) appears to be implicit, thus preventing the possibility of using it as an approximation in the prediction of the displacement. However, the implicit term, Δv_{n+1}, is already being solved for and is represented by the bracketed terms of equation (4). Therefore, inclusion of this seemingly implicit term renders a more accurate prediction for the displacement as:

$$\Delta d_{n+1} = \Delta t v_n + \frac{\frac{\Delta t}{2}\left(m + \frac{\Delta t}{2}c\right)^{-1}\left(\Delta \bar{f}_{n+1} - c\Delta t v_n - \Delta t r_n - \frac{k'}{2}(\Delta t)^2 v_n\right)}{1 + \left(m + \frac{\Delta t}{2}c\right)^{-1} k' \beta (\Delta t)^2} \tag{8}$$

where β in the constant average acceleration takes on the value of 1/4. Again, the implicit tangent stiffness, k^t may be replaced by k_o, the initial stiffness of the system. In many circumstances though, a better approximation to the tangent stiffness may be found by utilising the relationship between the restoring force and displacement increments over the previous time-step. For the SDOF case, the stiffness over the previous time-step may be expressed through the scalar operation:

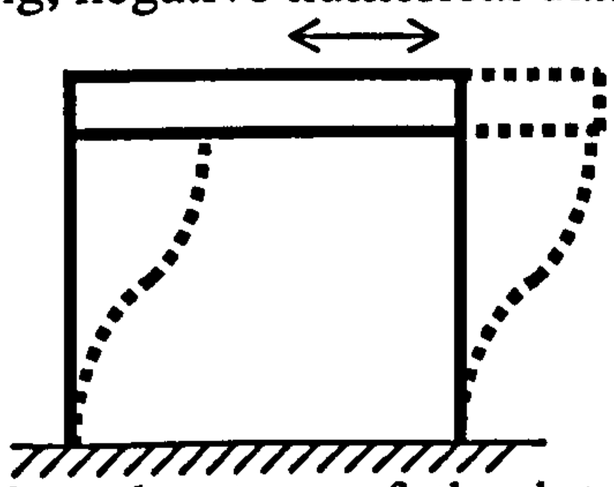
$$k_n^t x_n - k_{n-1}^t x_{n-1} = r_n - r_{n-1} \approx k_n^t \left(x_n - x_{n-1}\right)$$

$$k_n^t \approx \frac{\Delta r_n}{\Delta x_n} \tag{9}$$

This expression represents a much improved approximation to the tangent stiffness and may be implemented quite easily. Clearly, it is also be possible to estimate the stiffness for some MDOF structural systems by similar, yet extended procedures. Consistency between predicted and corrected displacements not only improves the accuracy of the algorithm, but also renders it unconditionally stable. This has been found through numerical simulations and inspection of the spectral radii of the amplification matrix. While theoretically impractical in pseudodynamic testing, the method facilitates solution through an explicit estimate of the tangent stiffness similarly to other implicit algorithms.

NUMERICAL EXAMPLE

A single storey sway frame system has been modelled by a single mass and sway spring of magnitudes 5400 kg and 3000 kN/m respectively, as shown in Figure 1. The natural sway frequency of the system, ω_n, is given by $\omega_n = \sqrt{(k/m)} = 23.57$rad/s. To display the advantages of this implicit, integrated method, the response of the sway frame to a single horizontal impulse has been shown. Two different time-step intervals have been used, Δt=0.03 sec. to display the numerical damping and Δt=0.086 sec. to display the stability properties. It is clear that the explicit method creates significant numerical damping while the implicit one preserves the energy with Δt=0.03 sec., Figure 2. When the time-step is increased to 0.086 sec. the stability limit for conditionally stable algorithms, $\Delta t_{max} = 2/\omega_n = 0.085$ is exceeded. This can clearly be seen as very strong, negative numerical damping in Figure 3.



M=5400 kg
K=3000 kN
ω_n=23.57 rad/s

Figure 1.

For evaluation of the advantage of the integral method with respect to capturing rapidly varying external excitation, the same SDOF system was subjected to the 1940 El Centro earthquake ground motion. The difference between using the integral form and the standard Newmark Explicit is not dramatic, but some improvement is present. As the linearisation error is of a random nature and limited magnitude, the effect would not be expected to be severe. For assessment of the consequence of applying the integral of the restoring force over the time-step rather than the linearisation based on the value at the start and end, and also of using an estimated tangent stiffness rather than the initial stiffness, a non-linear stiffness parameter is required. The same sway frame is used, but with the stiffness now being strain softening, defined by $k^t = k_o - 2\varphi d$,

where φ is a strain softening index. The effect of using the integrated, rather than linearised, restoring force integral has little effect in the numerical simulation. This is because in this simulation the linearisation will always underestimate the integral; during both loading and unloading. The error is thus cyclic and not cumulative. Under more realistic circumstances, the linearisation would underestimate during loading but overestimate during unloading. This effect would be artificially accumulate energy and may eventually result in instability. To fully evaluate this effect, practical experiments are required. The effect of using an estimated tangent stiffness results in a very slight damping rather than very slight negative damping, but owing to the fact that this is a second order term and of cyclic rather than cumulative nature, the error in both cases only represents damping of the order of 0.05%.

The implicit and explicit variations of this incremental, integrated algorithm have so far been evaluated in linear and non-linear numerical simulations, using the LabView modelling environment, which is designed for control and communication with hardware.

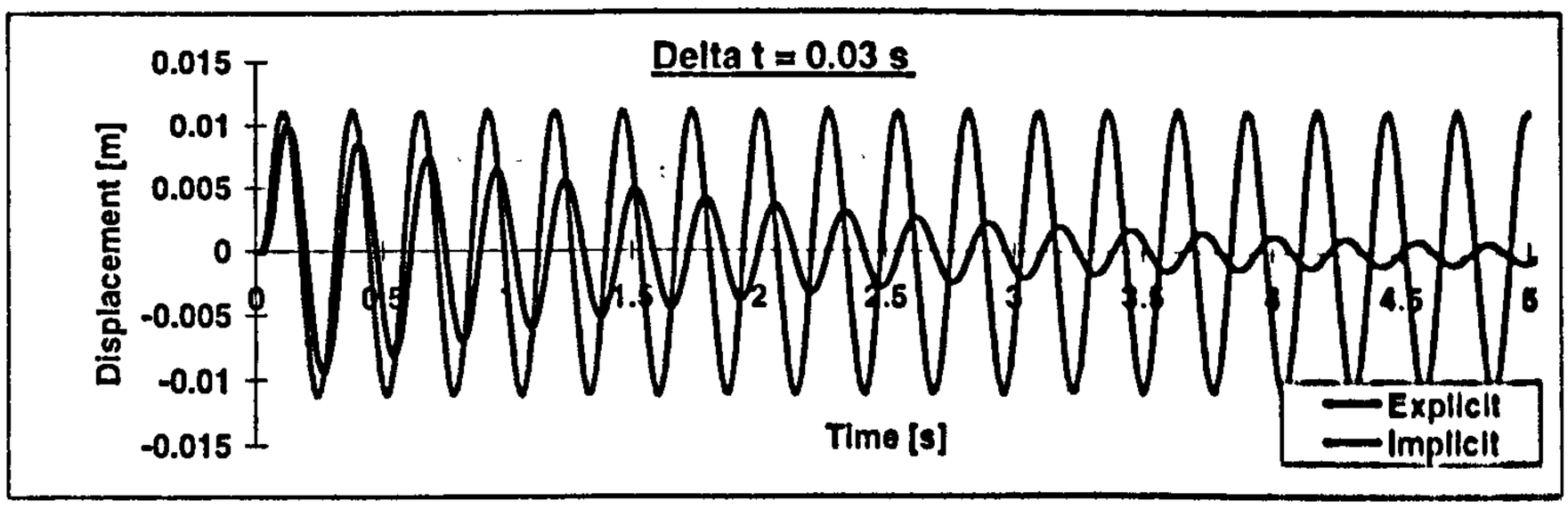


Figure 2.

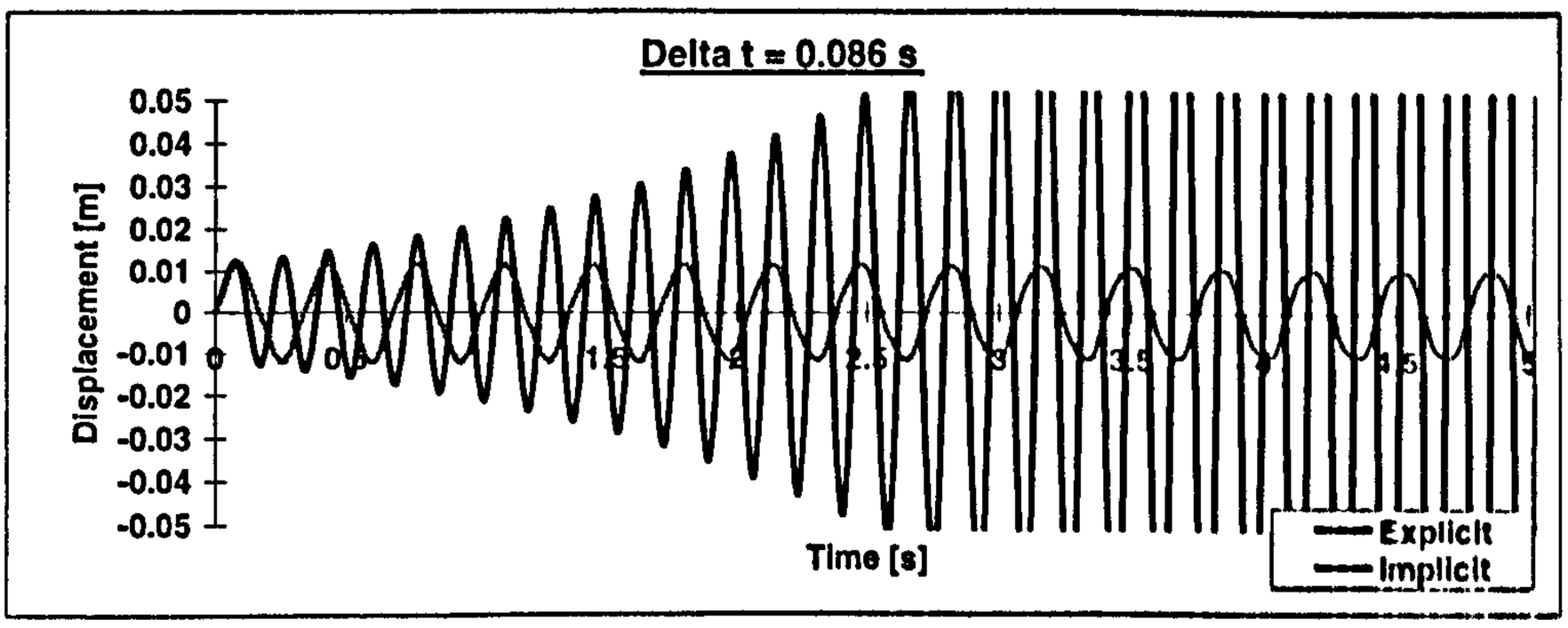


Figure 3.

CONCLUSIONS

The inclusion of an additional term in the displacement predictor in the integrated, explicit Newmark method has substantially increased the accuracy. The added term also renders the method implicit and unconditionally stable. Estimation of the required tangent stiffness has been improved based on the incremental force-displacement changes.

REFERENCES:

1. S.-Y. Chang, K.-C. Tsai and K.-C. Chen, 'Improved time integration for Pseudodynamic tests', *Earthquake Engng. Struct. Dyn.* **27**, 711-730 (1998)

Appendix F:

CMEM 2001 CONFERENCE PAPER

Assessment of algorithmic and control sensitivities in pseudodynamic testing

W. Algaard, N. Bićanić & A. Agar
Department of Civil Engineering, University of Glasgow, U.K.

Abstract

Software based implementation system for pseudodynamic testing realised within the LabView environment is discussed with an assessment of algorithmic and control sensitivities in the SDOF set-up. On the algorithmic level these include the effect of the type of time integration scheme and time step size, while the control effects are concerned with the method and rate of loading, controller capabilities and instrumentation accuracy. The sensitivities are illustrated in two model problems, an inelastic reinforced concrete column and an elastic steel column.

Introduction

Pseudodynamic testing is a combined computational/experimental technique for evaluating dynamic systems, originally proposed by Takanashi *et al.* in 1975 [1]. The method relies on modelling inertial and damping forces computationally, while the non-linear restoring forces are measured experimentally. This builds on the fact that inertia forces acting on a structure during motion may be correctly represented numerically, while the elastic restoring forces are too complex to account for with numerical models alone and need therefore be represented experimentally.

Components

Dynamic equilibrium equations for mass-spring-damper systems subjected to applied loads can generally be expressed as

$$M\frac{d^2x}{dt^2}+C\frac{dx}{dt}+r(x)=f \qquad (1)$$

where M and C are mass and viscous damping matrices and x, r and f are the displacement, restoring force and applied force vectors, respectively. The mass matrix can be assumed to remain constant throughout the test. The non-hysteretic part of the structural damping can reasonably be represented by a constant viscous damping matrix. The stiffness of the structure on the other hand, will display changes as damage accumulates, and the elastic restoring force may therefore be highly non-linear. To be able to capture the effect of this on the system response, the restoring force is modelled experimentally.

Implementation

The pseudodynamic test method uniquely utilises both computational and experimental terms to form the equation of motion (1). A time-stepping algorithm computes a displacement step, which is subsequently imposed on the structure by means of computer controlled servo-hydraulic actuators, figure 1. Once the structure has been deformed, the resulting restoring forces are measured. This can be done either during a hold period where the actuator remains stationary, or continually enabling smooth deformation of the structure. Based on the restoring force and the current damping and applied forces, the resulting acceleration may be computed, and the new displacement step can then be calculated. Sensitivities in pseudodynamic implementation are concerned with the speed and accuracy obtained experimentally as well as limitations on the algorithmic level.

Pseudodynamic test set-up

Experimental set-up

The experimental part of a pseudodynamic test set-up resembles that required for quasistatic testing. A SDOF facility at Glasgow has been built up to enable pseudodynamic testing of structural components comprising a stiff reaction rig onto which a horizontally orientated actuator is mounted. The actuator is capable of delivering a force of ± 50 kN and has a stroke of ± 50 mm. Displacement measurements are taken through an internal LVDT, while force is measured either directly through a load cell at the end of the actuator or indirectly through a pressure transducer within the pump system. A Moog-type servo valve, also remotely located, controls hydraulic pressure supply to the actuator. A software controller, running on a desktop PC, controls the valve. Communication between the PC and the instrumentation takes place through a high-speed communication card. The required active channels for a SDOF pseudodynamic test controlled locally consist of two input channels, force and displacement signals, and one output channel carrying the valve signal. A number of passive measurements may also be communicated through the same card. As all instrumentation is analogue, this card also carries out the D/A and A/D conversion.

Computational set-up

An unconventional approach has been elected to control, implement and execute the pseudodynamic tests. While hardware controllers and several computers are typically employed to handle execution and data logging, the system at Glasgow is built up of a single PC with a high-speed communication card. Not only does the same computer conduct the entire running of the test, but all the computational components are included in the same environment. These range from the time integration algorithm and equilibrium calculations, through the implementation and execution system with data logging to the actual actuator controller unit. In fact, the whole computational side of the set-up is created as a single piece of code in the National Instruments' LabView 5 environment. This entails a fully integrated implementation system entirely on a software level.

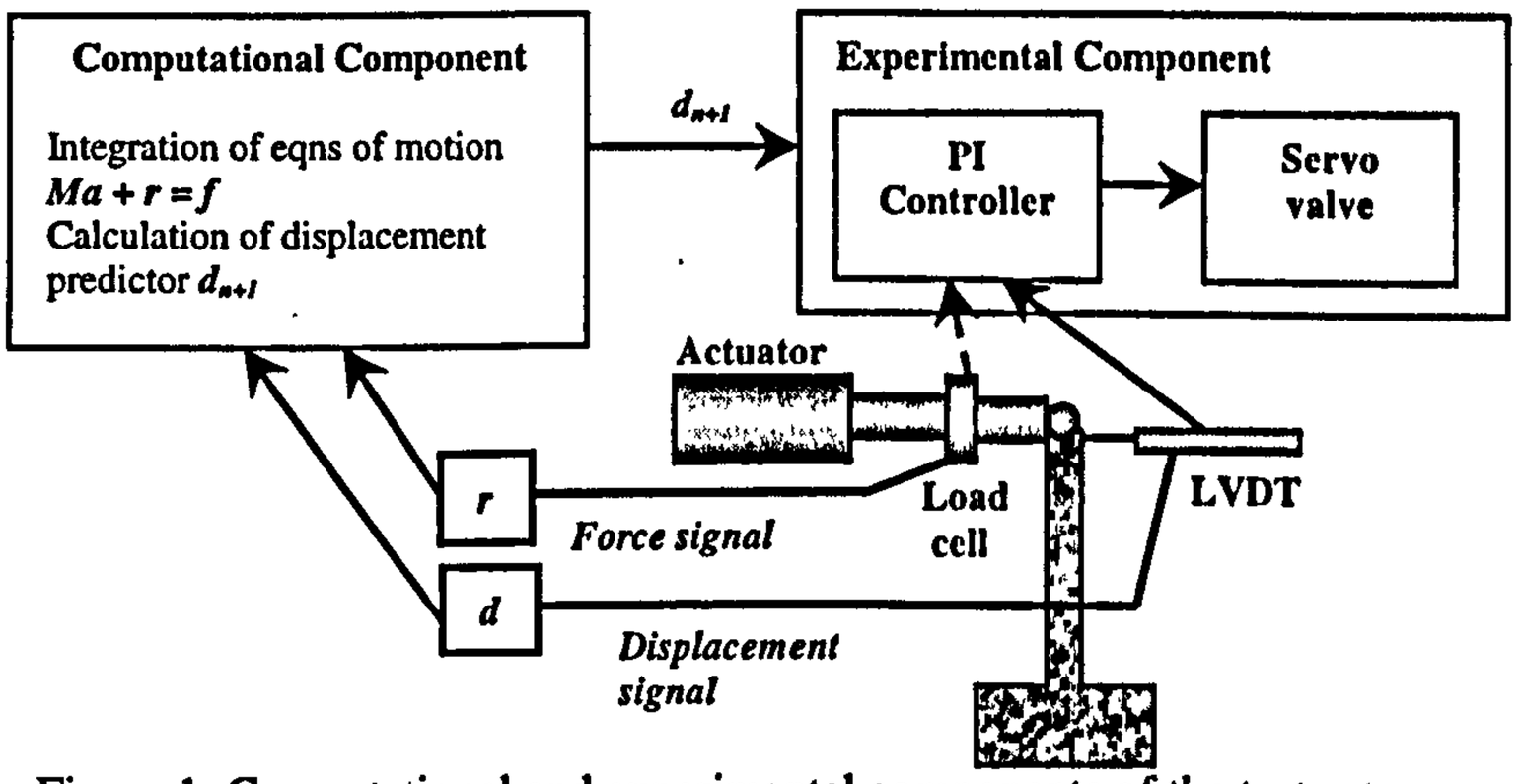


Figure 1: Computational and experimental components of the test set-up.

Implementation

A semi-continuous implementation system has been adopted implying the actuator motion is not interrupted by prescribed hold-periods. The two major parts of the implementation loop, time integration calculations and control iterations, are however not carried out concurrently. Whenever the control algorithm concludes, active control of the actuator is lost until a new target displacement is calculated in the next displacement step. The method is therefore not fully continuous. All required calculations are carried out in an amount of time comparable to that required for each iteration in the control algorithm. The implementation method thus relies on switching between time integration and control without delay, which is achieved by integrating the two algorithms fully by coding them into the same program [2].

Sensitivity study

Considerable work has been carried out to study the error propagation effects in pseudodynamic tests, e.g. Combescure *et al.* [3], but little or no research has

been carried out to investigate implementation sensitivities in general. These range from those concerned with time integration (type of, and details within the time stepping algorithm, as well as time step size) to those concerned with the experimental implementation (e.g. rate and method of loading, continuous or step-wise, controller capabilities, accuracy obtained and details within measurements).

The relationships between: *time step size*, *speed of implementation*, *discrete step accuracy* and *final system response* are considered. Two different time stepping algorithms are employed to evaluate how the above relationships may be affected by algorithmic differences in both time integration and control.

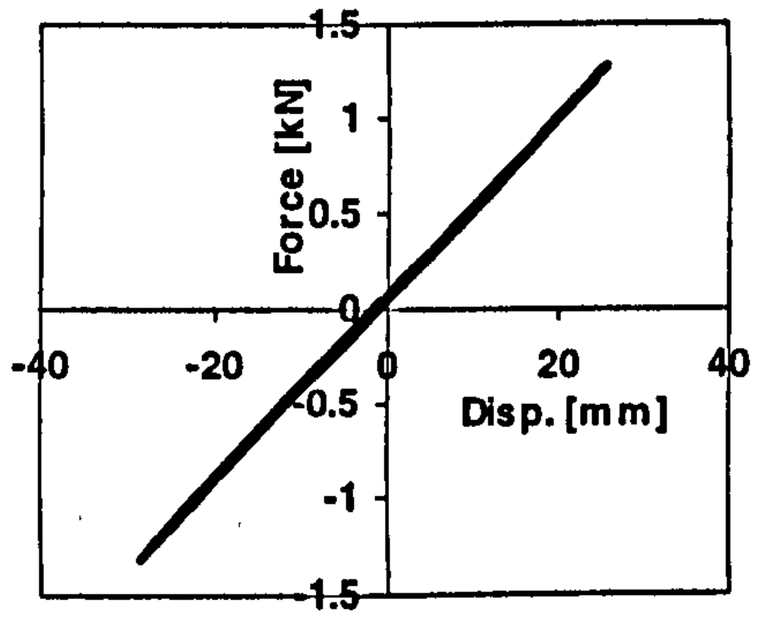


Figure 2a: Force-displacement loops for steel column.

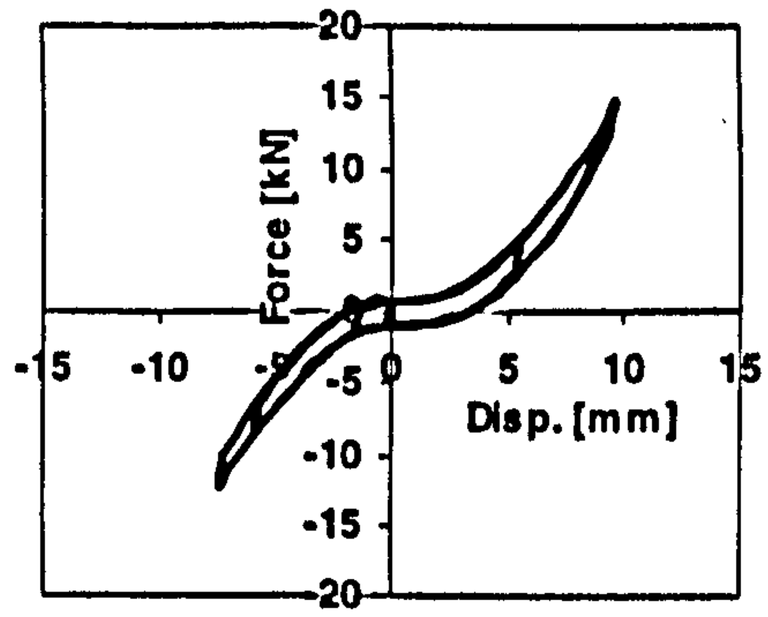


Figure 2b: Force-displacement loops for concrete column

Test programme

In order to evaluate the relationships discussed above, repeated pseudodynamic tests on two different dynamic systems have been carried out. These comprise a *reinforced concrete stub column* and a *slender steel column*, both with lumped virtual masses on top, yielding an inverted pendulum system. The masses have in both cases been tuned to create structures with natural frequencies with fundamental periods of around 1.2 seconds. Both structures were exposed to a scaled Port Hueneme accelerogram, and the first 4 seconds of the response were modelled using the central difference and the novel Newmark Implicit – Integral Form methods [4, 5]. Zero viscous damping is applied throughout. Typical force-displacement curves for the two columns subjected to a horizontal load at the top can be found in figures 2a and 2b.

Reinforced concrete stub column

The reinforced concrete column was designed to display a significant non-linear behaviour. Furthermore, the specimen was slightly damaged prior testing by imposing displacements exceeding those expected during pseudodynamic sensitivity tests. This was done to ensure repeatability of tests by avoiding further damage taking place. Maximum displacements of ± 10 mm were aimed for during these tests.

Time step sizes of 0.004, 0.04, 0.08 and 0.16 seconds were employed, requiring 1000, 100, 50 or 25 steps, respectively, to model the first 4 seconds of the response. Total testing time varied from 8 seconds to 2 minutes and 18 seconds.

Steel column

A slender steel column was designed for two reasons. Firstly because it would display a near perfectly linear force-displacement relationship providing a contrast to the non-linear reinforced concrete specimen. Secondly, due to its flexibility, the mass required to provide the desired frequency is small. This enables the true dynamic system to be created, opening for the possibility of a real reference solution, a so-called *snap-back test*, to be obtained.

The column comprised a 1600 mm long rectangular hollow steel section providing a second moment of area in the weak direction of 46.7 cm^4. The section was expected to remain elastic over the full stroke of the actuator (± 50 mm). In this case, time step sizes of 0.04, 0.08 and 0.16 seconds were selected, requiring 100, 50 or 25 steps. Total testing time here ranged from 9 to 50 seconds.

Ground motion

In order to investigate the effect on the response resulting from differences in the time step size, it is imperative that the structure is always subjected to identical loading, regardless of the step size. As the sampling period of the applied accelerogram is 0.004s and step sizes vary from 0.004s to 0.16s, truncation techniques are required for all step sizes different from that of the sampling rate.

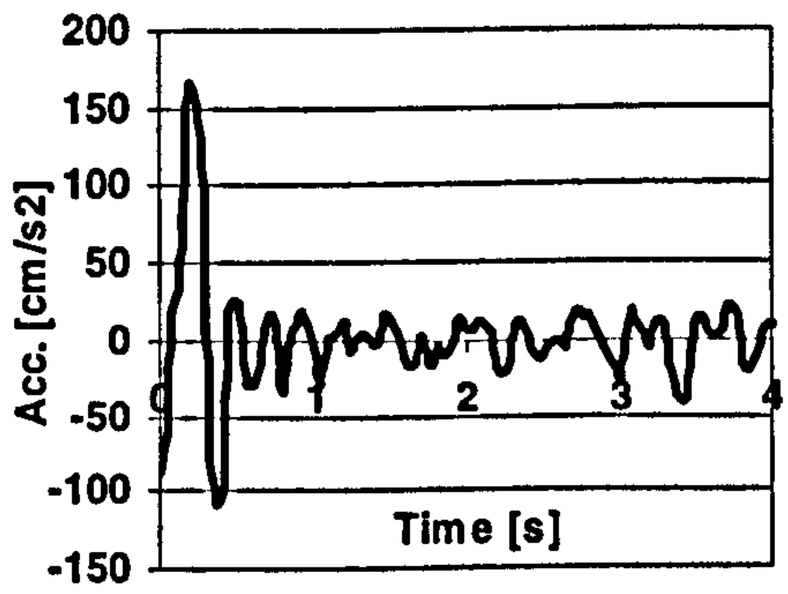


Figure 3a: Port Hueneme NS accelerogram.

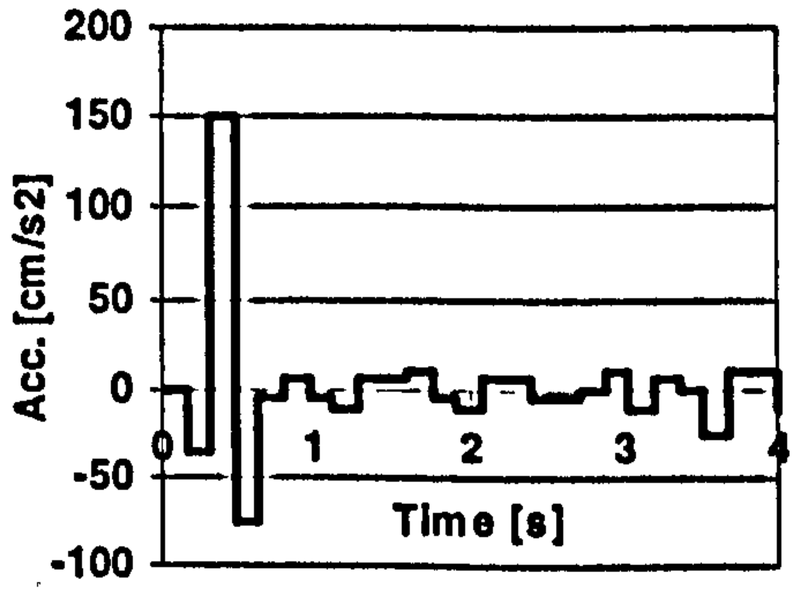


Figure 3b: Artificial accelerogram, sampling period 0.16s.

The truncation technique performs a different function depending on the time stepping algorithm employed. While the conventional algorithms include a single force value per time step, the integral form algorithms apply the time integral of the forces acting over the time step. This integral can be calculated numerically by taking account of the full sampling rate of the accelerogram. However, for direct comparison purposes, keeping the loading identical is preferable. To make the loading universal for time steps up to 0.16s, a synthetic

accelerogram with a sampling rate of only 0.16s was created. When employing smaller time steps, the same values are simply repeated. Figures 3a and 3b display the Port Hueneme NS and the artificially generated accelerograms.

Results

Speed – accuracy – time step size relationship

Figures 4a and 4b indicate the prevailing relationship between the discrete step accuracy, implementation time and time step size. These results are all obtained through tests on the reinforced concrete column using the Newmark Implicit – Integral form algorithm. Considering first figure 4a, which shows the relationship between test duration and maximum step error, i.e. the largest discrepancy between targeted and achieved positions, for each time step size. It is clear that increased test durations, i.e. slower tests, maintain better accuracy. Furthermore, it shows that increasing time step sizes also enable superior accuracy. It is also noted that the largest time step size is very accurate even for the shortest test durations, and that with the smallest time step this level of accuracy cannot be achieved even with considerable test durations.

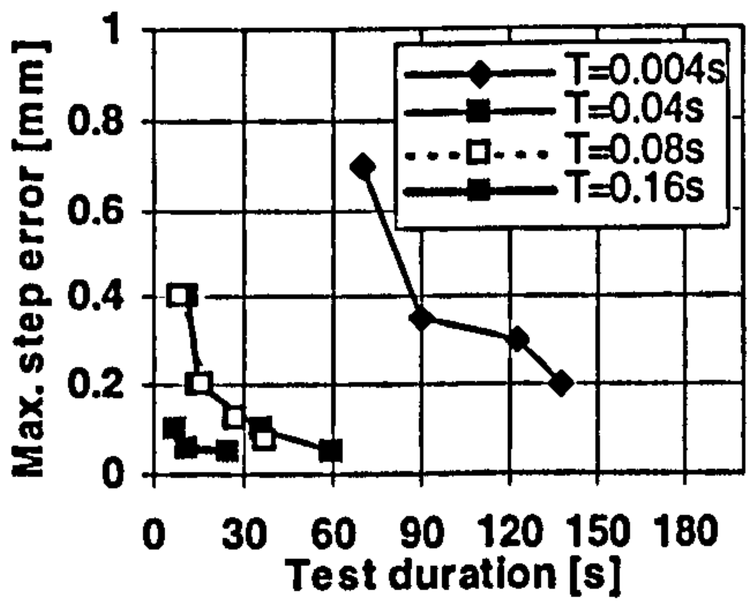


Figure 4a: Test duration vs. maximum discrete step error .

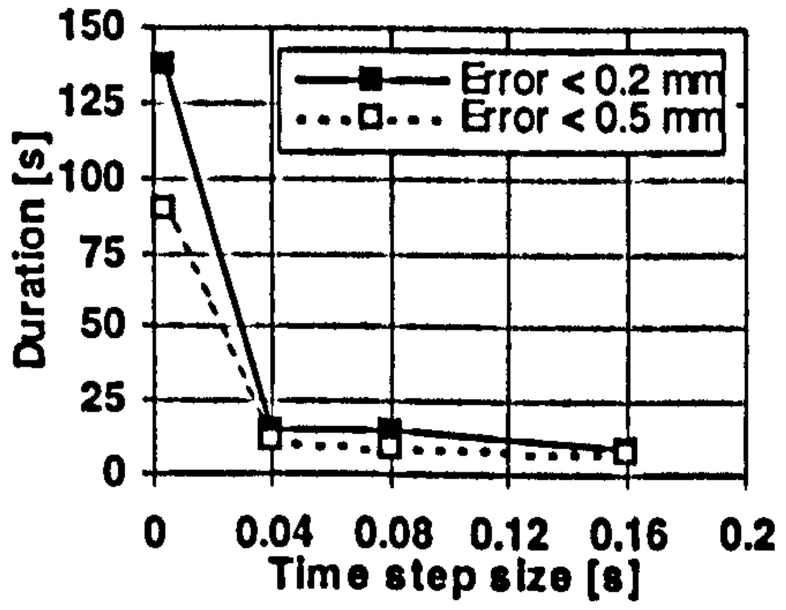


Figure 4b: Time step size vs. test duration.

Figure 4b gives an indication to the testing time required to maintain a minimum accuracy of 0.2 and 0.5mm for the range of time steps. Again it shows that the smallest time step, 0.004s, requires considerable more time than the larger ones. The differences between the 0.04, 0.08 and 0.16s steps are not considerable, but still display the general trend that the larger time steps require less overall testing time to maintain a specified accuracy.

Speed – response relationship

There appears to be a relationship where increased implementation speeds result in a less damped/negatively-damped response in some cases. The effect is not general however and emerges mostly during particularly fast implementation. Figure 5a displays clearly how the reduced durations lead to growing amplitudes

in the case of the 0.08s step size, while the same cannot be found in figure 5b using the 0.16s step size.

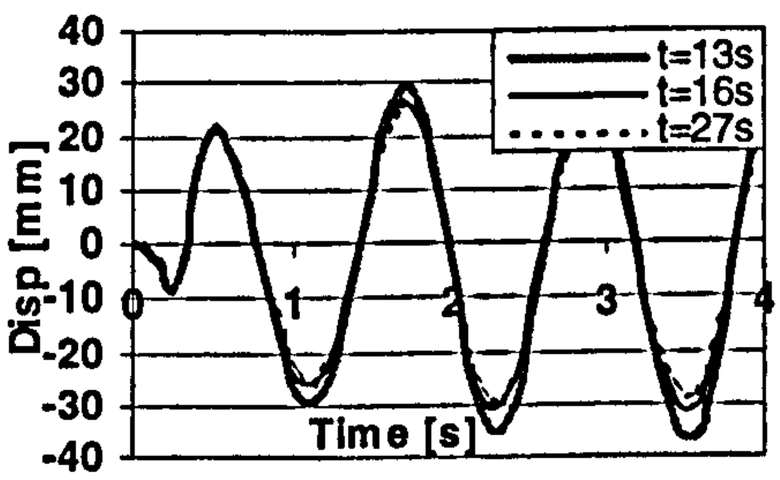


Figure 5a: Response obtained using 0.08s time steps.

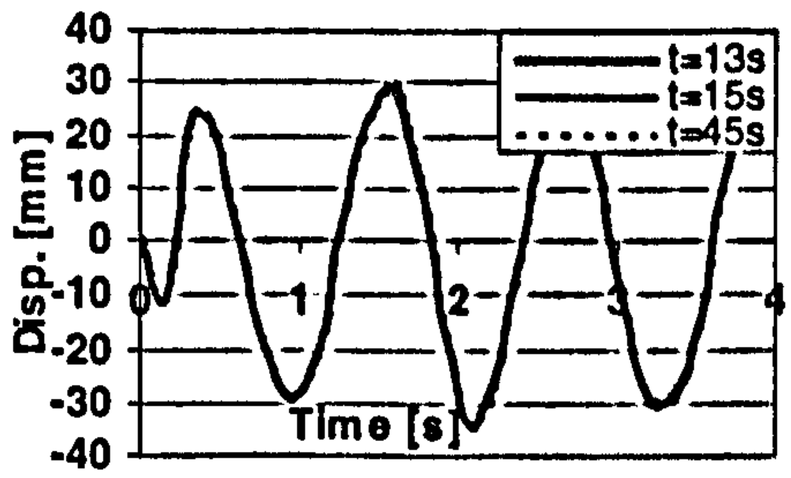


Figure 5b: Response obtained using 0.16s time steps.

Response – time integration relationship

Displaying the system response created using different time step sizes and time integration algorithms should reveal any effects on the response caused the variations in the time stepping schemes. System responses of the reinforced concrete column and the elastic steel column to the artificial accelerogram using step sizes of 0.04, 0.08 and 0.16s with the Newmark Implicit – Integral Form and central difference algorithms are displayed below.

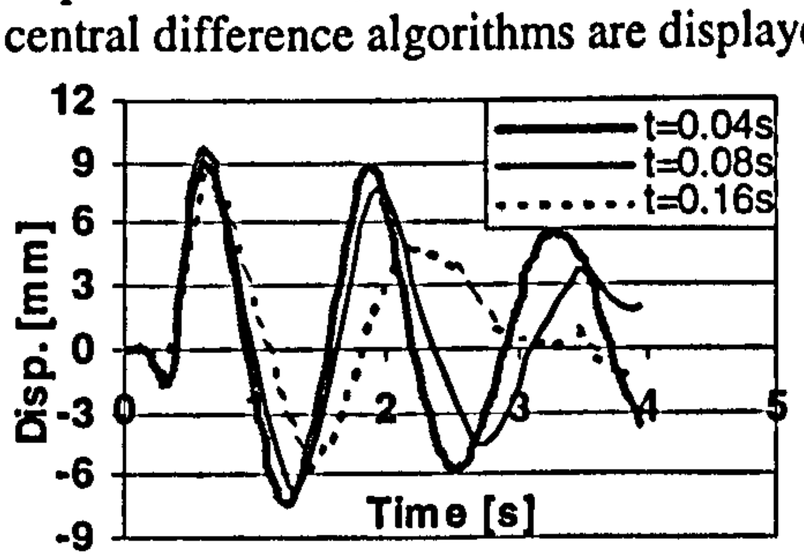


Figure 6a: Response obtained using integral form method.

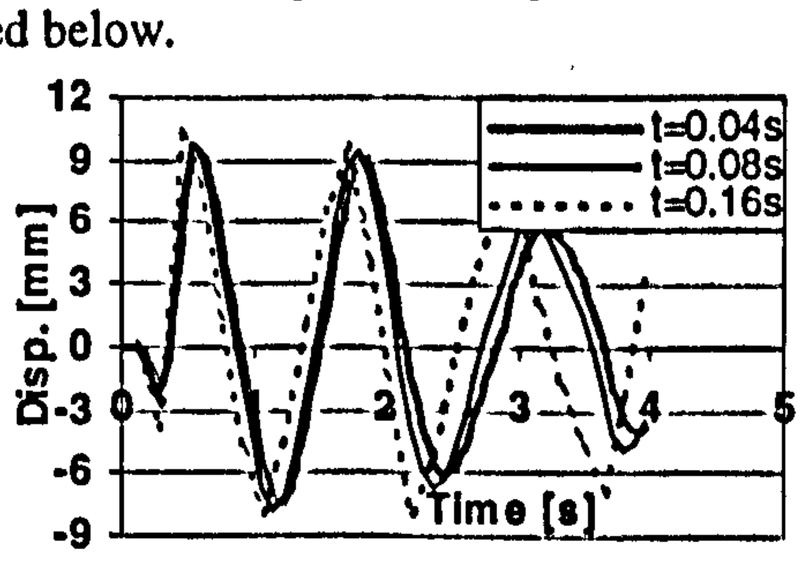


Figure 6b: Response obtained using cent. diff. method

Considering first the tests on the reinforced concrete specimen, figures 6a and 6b, it immediately becomes apparent that large differences in the response result both from the choice of the time step size and the integration algorithm. Assuming the shortest time step generates the most accurate response, increasing the step size with the central difference method results in a period shortening, while for the integral form it results in period elongation. Additionally, the central difference method displays evidence of amplitude amplification, while the integral form method displays evidence of amplitude decay. Similar system responses were obtained for the steel column under the same conditions, figures 7a and 7b. The integral form method still displays period elongation and the central difference method period shortening. The integral form also appears to generate more damping.

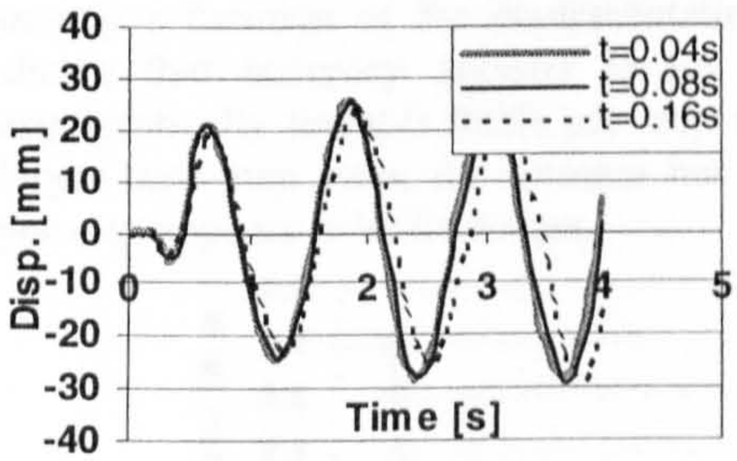


Figure 7a: Response of steel column using integral form method.

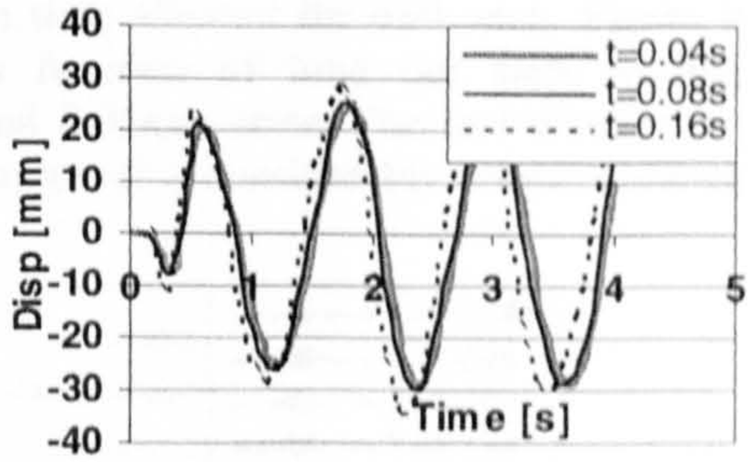


Figure 7b: Response of steel column using cent. diff. method.

Numerical simulations may be carried out on the steel column system as the stiffness is near perfectly linear. Using the measured stiffness, including the same mass and exposing the sample to the same ground motion as in the pseudodynamic tests, the response was obtained numerically using the same time integration methods. The response obtained can be seen in figures 8a and 8b using the integral form and central difference methods respectively.

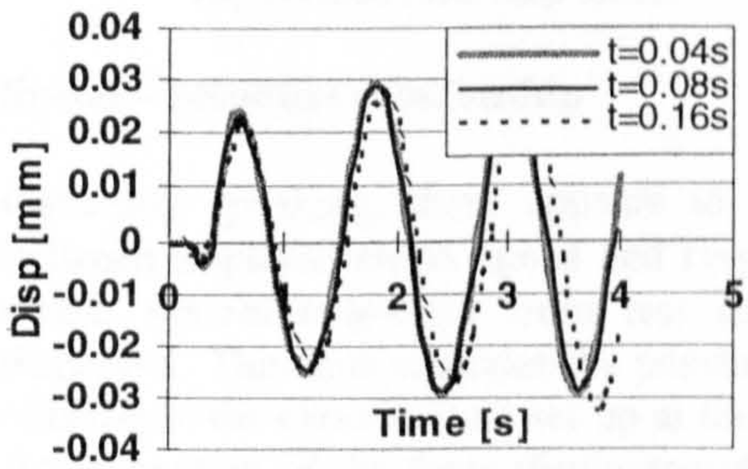


Figure 8a: Numerical simulation using integral form method.

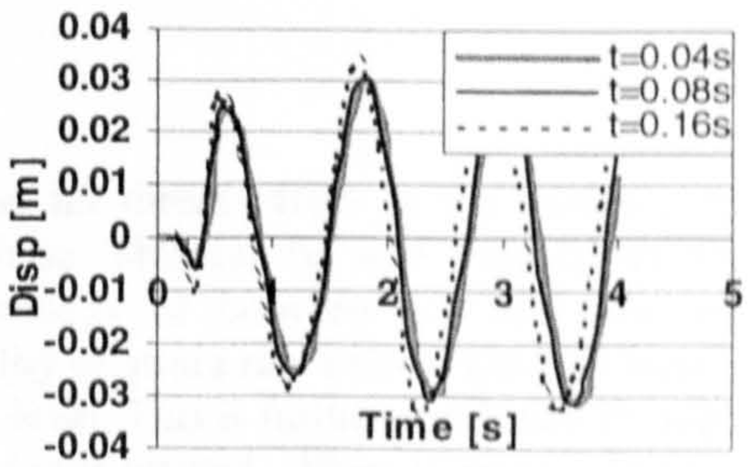


Figure 8b: Numerical simulation using cent. diff. method.

Analysis

Speed – accuracy – time step size relationship

The trend displayed in figure 4a suggests that increasing test durations and time step sizes improve accuracy. As longer test durations provide more time per time step, more time is available for control. This allows more control iterations and lower actuator speeds, so it is evident that improved accuracy may be obtained. Similarly, increased time step sizes imply more time available for each step, yielding the same effect. As an example, using the 0.08s step size can maintain a minimum accuracy of 0.2mm when running the test in 15s while the 0.004s step size requires 135s for a similar accuracy. However, 15s allows 0.3s per step using 0.08s steps, while 135s allows only 0.135s per step using 0.004s steps. Similarly, the very fast and accurate test using 0.16s steps, 0.07mm maximum error for a 9s duration, does in fact allow 0.36s per step, while comparable accuracy using the 0.04s steps is obtained by providing 36s test duration, or, again, 0.36s per step. It appears therefore that the discrete step accuracy is

mainly a function of the implementation time allowed for each step. Figure 9 shows that accuracy appears to be a function of time per step, tending asymptotically towards 0.05s per step and 0.05mm error. The fact that for the larger time step sizes, the actuator has to travel a considerably longer distance per step appears to be irrelevant.

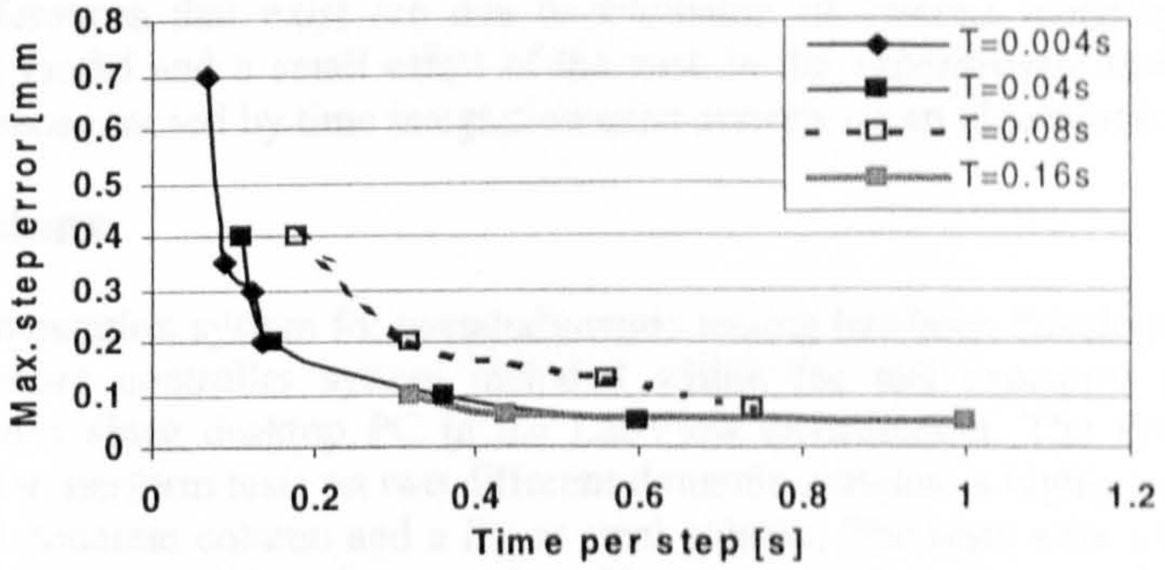


Figure 9: Relationship between time allocated per step and maximum step error for various time step sizes.

Speed – response relationship

Generally speaking, there appears to be no direct effect in the relationship between implementation speed and response. This can be seen from figure 5b, where system responses from test durations of between 11s and 45s are displayed. This thus excludes the possibility of strain rate effects and real inertia existing in the experimental set-up at this level. This is further confirmed through the inspection of the force-displacement loops created. When in some instances the response appears to be affected by the implementation speed, this is in fact due to the loss of control accuracy through the fast tests. Although the displacement errors do not carry onto the next step, the error in the force due to the incorrect positioning clearly does. Thus, as long as the displacement error is limited, say to 0.2mm, the response remains unaffected.

Response – time integration relationship

The response of the reinforced concrete column to the ground motion is highly influenced by the time stepping details. With increasing time step sizes, the difference between the two integration methods increases. The central difference method displays period shortening, while the Newmark implicit – Integral form displays period elongation and significant damping, as the time step sizes are increased from 0.04s. The damping is believed to originate from insufficient accuracy in the force measurement through the pressure transducer.

In order to get a better understanding of the effects on the response, the tests on the steel column were carried out. These revealed similar period elongation when using the integral form algorithm and shortening when using the central difference method. Evidence of more damping with the integral form method

existed, but through further numerical tests covering longer test time, it was confirmed that both methods were energy stable for linear systems. The numerical tests also provided reference solutions to the pseudodynamic ones. Response created pseudodynamically is near identical with the numerically generated one, for all time step sizes, comparing figures 7a/8a and 7b/8b. The small differences that exist are due to exclusion of viscous damping in the numerical model and a small offset of the zero in the experimental tests, hence the differences caused by time integration exist entirely on an algorithmic level.

Conclusions

An implementation system for pseudodynamic testing has been developed based on a software controller system included within the test execution program running on a singe desktop PC in the LabView environment. The system has been used to perform tests on two different dynamic systems: a highly non-linear reinforced concrete column and a linear steel column. The tests were conducted to evaluate the sensitivity of the results with respect to the method and speed of implementation. It has been concluded that utilisation of smaller time steps require longer overall testing time, as a minimum amount of time is required for accurate implementation of each displacement step, and that the implementation speed is not affected by the time integration scheme. On the other hand it has been shown that these affect the response, however that this is exclusively due to algorithmic effects at least in the linear case. Implementation speeds are not seen to affect the response other than through the loss of accuracy in the fastest test.

References

1: K. Takanashi, K. Udagawa, M. Seki, T. Okada and H. Tanaka, 'Nonlinear earthquake response analysis of structures by a computer-actuator on-line system', *Bull. earthquake resistant struct. res. centre (Tokyo)* issue 8, (1975)

2: W. Algaard, 'Algorithmic and control implementation in pseudodynamic testing' *Ph.D. Thesis*, Department of Civil Engineering, University of Glasgow, Scotland, to be submitted 10.2001

3: D. Combescure & P. Pegon, 'α-Operator Splitting time integration technique for pseudodynamic testing. Error propagation analysis', *Soil Dynamics and Earthquake Engineering,* **16**, 427-443, (1997)

4: W. Algaard, A. Agar and N. Bicanic, 'Enhanced Integral Form of the Newmark Time Stepping Scheme for Pseudodynamic Testing', *Int. Journal of Engineering Computations* (subm.)

5: S-Y. Chang, K-C. Tsai and K-C. Chen, 'Improved Time Integration for Pseudodynamic Tests', *Earthquake Engng Struct Dyn.*, **27**, 711-730, (1998)

Appendix G:

ENGINEERING COMPUTATIONS PAPER, VOL 18, 2001

Enhanced integral form of the Newmark time stepping scheme for pseudodynamic testing

William Algaard, Alan Agar and Nenad Bicanic
Department of Civil Engineering, University of Glasgow, Scotland, UK

Keywords *Testing, Time step control, Algorithms*

Abstract *A novel integral form time-integration algorithm for pseudodynamic testing is proposed, based on the Newmark implicit algorithm. The scheme builds on the recently proposed integral form of the Newmark explicit algorithm which exhibits improved abilities to handle rapidly varying loads and stiffness properties during pseudodynamic testing, but displays some numerical damping and conditional stability. The enhancement is based on the inclusion of an additional term in the displacement predictor, which not only renders the algorithm more consistent, but it eliminates numerical damping and makes the algorithm unconditionally stable.*

Introduction

Pseudodynamic testing is a combined computational/experimental technique for evaluating dynamic systems, originally proposed by Takanashi *et al.* (1975). The method relies on modelling inertial and damping forces computationally, while the non-linear restoring forces are measured experimentally.

Dynamic equilibrium equations can generally be expressed as:

$$M\frac{d^2x}{dt^2} + C\frac{dx}{dt} + r(x) = f \tag{1}$$

where M and C are mass and viscous damping matrices and x, r and f are the displacement, restoring force and applied force vectors, respectively. In what follows, it will be assumed that $r(x)$ is the only source of non-linearity, which can be obtained accurately enough through experimental measurements.

The pseudodynamic test method uniquely utilises both computational and experimental terms to form the equation of motion (1). The response is obtained by discretising time and calculating the response in a step-by-step manner. A time stepping algorithm computes a displacement step, which is subsequently imposed on the structure by means of computer controlled servo-hydraulic actuators, Figure 1. Once the structure has been deformed, the resulting restoring forces are measured. Based on these and the current damping and applied forces, the resulting new acceleration may be calculated. A new displacement step can then be calculated and the next step has thus commenced.

Engineering Computations
Vol. 18 No. 3/4, 2001, pp. 676-689

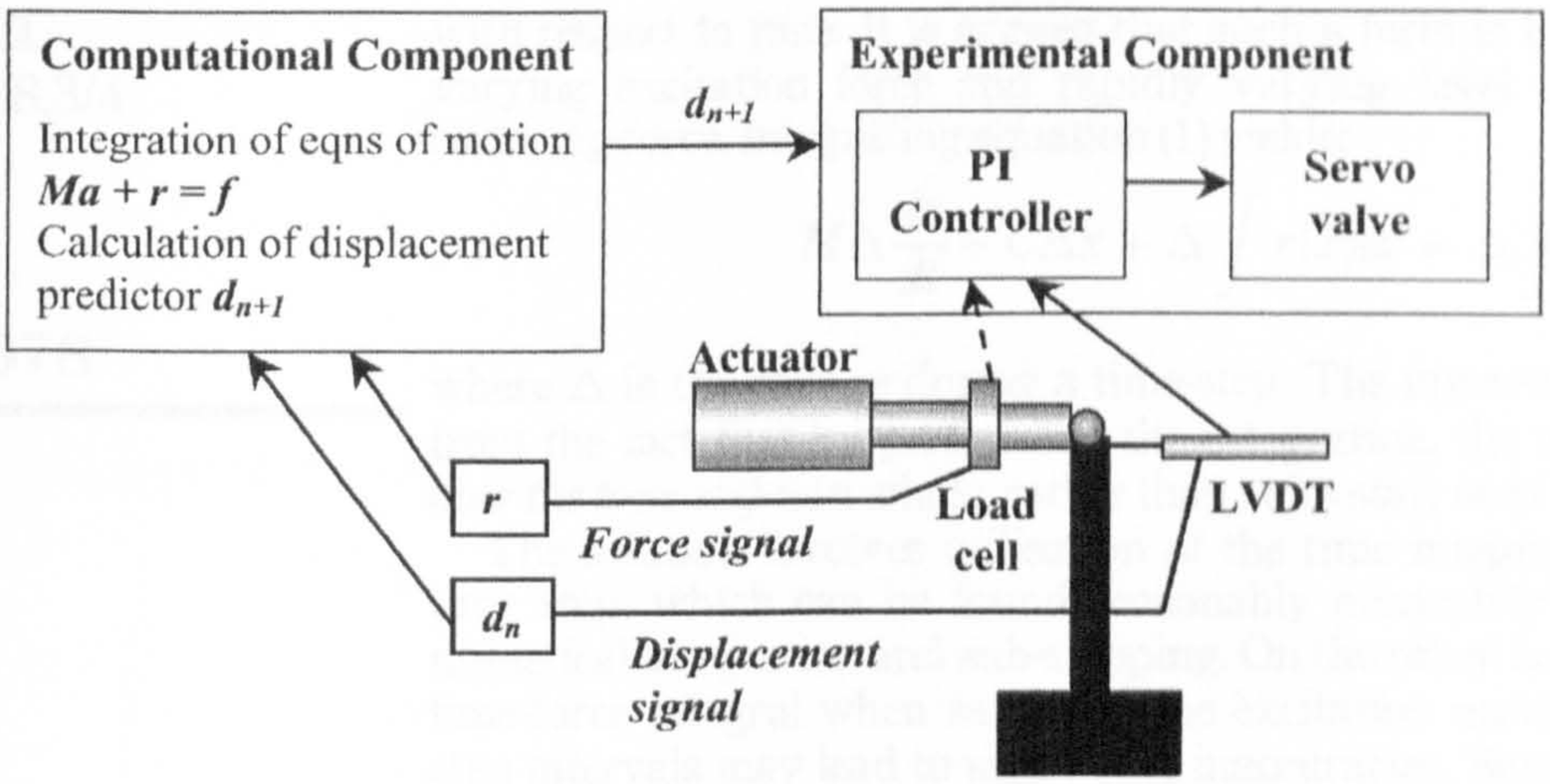


Figure 1. Computational and experimental components of the pseudodynamic test set-up

In comparison to shaking table testing there are some important differences. As the pseudodynamic testing is carried out in a step-by-step fashion, it is clear that it is unrealistic to be able to progress the test in real time. Furthermore, as inertial effects are modelled computationally, such forces need not and should not exist in the physical model. The time scale of a typical test is therefore expanded by one to three orders of magnitude, which has both beneficial and adverse effects. The fact that the structure is displaced slowly (and can even be stopped) provides a good opportunity for inspection and any detailed readings to be taken; however the strain rate effects on material response are neglected.

Several time stepping algorithms (Appendix) have been proposed for application in pseudodynamic testing (Bursi and Shing, 1996; Combescure and Pegon, 1997; Chang, 1997). The majority of these are explicit due to the fact that the non-linear structural restoring forces at the end of any time step are unknown and displacement iterations in pseudodynamic tests are undesirable as these might result in partial unloading (Shing and Manivannan, 1990). Although implicit methods have the advantage of being unconditionally stable, the duration of the time steps still has to be limited for accuracy purposes, due to rapid changes in both loading and stiffness. However, the so-called *integral form* of the Newmark explicit method, proposed by Chang *et al.* (1998), relies on integrating the second-order equation of motion (1) once with respect to time and it is argued that this method exhibits improved abilities to model rapidly varying loads and stiffness. Implementation of the integral form into the pseudodynamic testing framework is however complicated by the fact that certain stiffness related terms become implicit. The possibility of enhancing Chang's formulation of the integral form by modifying some of its aspects will be considered further.

Newmark explicit – integral form

The integral form of the Newmark explicit was initially suggested by Chang *et al.* (1998), by integrating the equation of motion (1) in its incremental form once

with respect to time. It is argued that such a form is better suited for rapidly varying excitation force and rapidly varying level of non-linearity of the restoring force. Integrating equation (1) yields:

$$M\Delta\frac{dx}{dt} + C\Delta x + \Delta\int r(x)dt = \Delta\int f dt \tag{2}$$

where Δ is the change during a time-step. The improved accuracy originates from the fact that by performing the integration, the equilibrium is satisfied *over the time step* as a whole, rather than at its start or at its end.

The solution involves utilisation of the time integral of the force for each time step, which can be found reasonably accurately through some simple numerical integration and sub-stepping. On the other hand, linearisation of the time-force integral when sampling the excitation accelerogram at large time step intervals may lead to significant inaccuracies. Similarly, for the restoring force, a linearised relationship between the start and the end values may very poorly represent the actual variation of the force over the time step (Figure 2). Clearly, by linearising between the start and the end points underestimates the restoring force during loading and overestimates it during unloading – this evidently adds energy to the system and could potentially lead to an instability. While the error in the linearisation of the excitation force is of a random nature, it should be noted that the error associated with restoring force is systematic and cumulative.

The applicability and the advantages of implementing the integral form method into pseudodynamic testing has been thoroughly investigated in Chang *et al.* (1998). As mentioned, the theory builds on integrating the incremental equations of the explicit format of the Newmark method $\beta = 0$ in the Newmark implicit relation:

$$d_{n+1} = d_n + \Delta t v_n + \left(\frac{1}{2} - \beta\right)(\Delta t)^2 a_n + \beta(\Delta t)^2 a_{n+1}$$

i.e.

$$d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}(\Delta t)^2 a_n$$

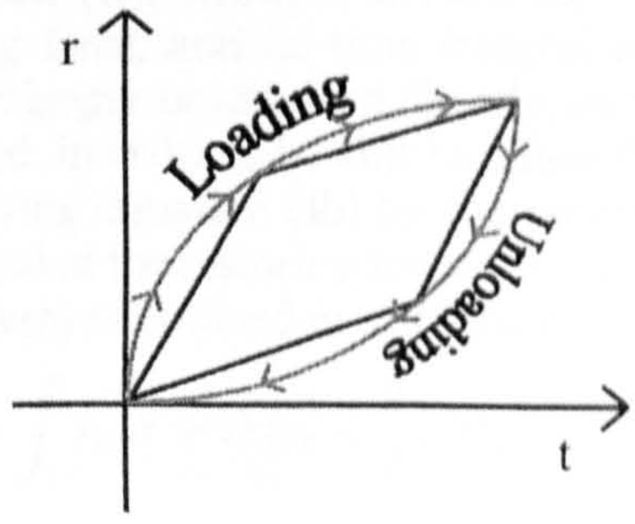


Figure 2.
Cumulative error as a result of linearisation of the restoring force

The Newmark explicit equations:

$$M\Delta a_{n+1} + C\Delta v_{n+1} + \Delta r_{n+1} = \Delta f_{n+1}$$

$$d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}(\Delta t)^2 a_n \tag{3}$$

$$v_{n+1} = v_n + \frac{1}{2}\Delta t(a_n + a_{n+1})$$

are integrated once with respect to time, which leads to the following equations:

$$M\Delta v_{n+1} + C\Delta d_{n+1} + \Delta \int r_{n+1} dt = \Delta \int f_{n+1} dt \tag{4a}$$

$$\int d_{n+1} dt = \int d_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n \tag{4b}$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t(v_n + v_{n+1}) \tag{4c}$$

where Δt is the time step duration, d and v the displacement and velocity, respectively, and Δ indicates the change over one time step. Whereas, in the usual Newmark explicit format, equations (3), the equations are solved for the change in acceleration, the equations of motion in the integral form, equations (4), are now solved for the *change in velocity*. Additionally, the integral form has an expression for the time-integral of displacement instead of the displacement predictor Δd_{n+1} in the usual form. More importantly, the term in the integral form no longer represents an explicit prediction that may be used as an initial displacement step in pseudodynamics. The displacement step is now an implicit function of v_n and v_{n+1} and can be found from equation (4c), which in turn requires the solution of equation (4a) to obtain the velocity at the end of the time step, v_{n+1}. In effect, the action of integrating the set of equations has rendered the method *implicit* in the sense that the predictor displacement cannot be deduced directly any more. The integral form algorithm also requires an assessment of the integral of the restoring force before the displacement predictor can be calculated. Such an estimate enables the solution for Δv_{n+1} to be found which in turn produces an explicit predictor for the displacement, equation (4c), which is needed for the pseudodynamic implementation. The restoring force, and its time integral are non-linear functions of displacement and can no longer be obtained directly, as no predictor displacement step exists to be imposed. In order to be able to utilise the algorithm, Chang *et al.* (1998) suggest multiplying equation (4b) by the tangent stiffness and an explicit expression of the integral of the restoring force at $t = t_{n+1}$ may be found (here expressed for an SDOF system) as outlined in equation (5).

$$\int r_{n+1} dt = \int r_n dt + \Delta t k d_n + \frac{k}{2}(\Delta t)^2 v_n = \int r_n dt + \Delta t r_n + \frac{k}{2}(\Delta t)^2 v_n \tag{5}$$

The physical interpretation of the above expression can be seen from the graph of restoring force vs. time, Figure 3, where $\int r_n dt$ indicates the restoring force-time area at a given time t while the sum of the two remaining terms represents the projected trapezoidal area assuming that a constant velocity exists until the end of the step.

Such a procedure tentatively assumes that the tangent stiffness is known or may be obtained somehow, which will normally not be the case in pseudodynamic testing (Chang *et al.*, 1998). Only for the SDOF and for certain simple MDOF structures may the stiffness matrix be computed from experimental data, and then only once a time step has been completed. To overcome the problem of the unknown tangent stiffness matrix, Chang *et al.* (1998) suggest replacing it with the initial stiffness term. The error involved is not large as the tangent stiffness is required only in the second order term on the right hand side of equation (5) (Chang *et al.*, 1998). In any case, once an expression for the integral of the restoring force exists, Chang *et al.* (1998) suggest a solution procedure where they solve for Δv_{n+1} from equation (4a) by substituting equation (4c) for d_{n+1}. When considering a SDOF system, the velocity change be can expressed as:

$$\Delta v_{n+1} = \left(m + \frac{\Delta t}{2}c\right)^{-1}\left(\Delta \int f_{n+1}dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right) \quad (6)$$

where k_0 is the initial stiffness in place of the tangent stiffness term. By substituting this result back into equation (4c), a prediction for the change in displacement can finally be expressed as follows:

$$\Delta d_{n+1} = \Delta t v_n + \frac{\Delta t}{2}\left(m + \frac{\Delta t}{2}c\right)^{-1} \left(\Delta \int f_{n+1}dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right) \quad (7)$$

which, when added onto the previous displacement value, furnishes an explicit displacement predictor to be applied in pseudodynamic tests similarly to any other displacement predictor. However, the method now departs from the traditional procedure in pseudodynamic testing. While the predictor displacement step is being imposed on the structure, the induced restoring force is continuously measured and its corresponding time integral is evaluated numerically. At the end of the step, this will represent an experimentally

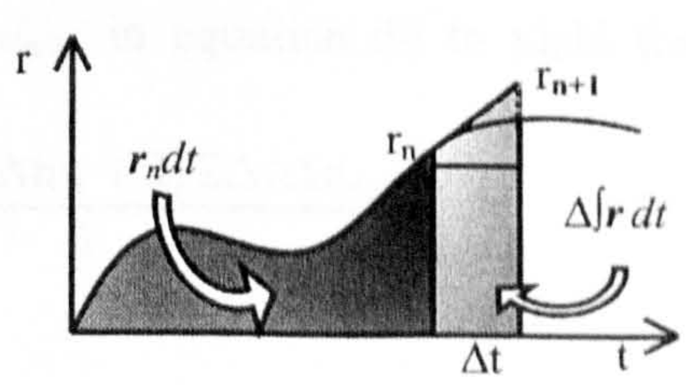


Figure 3.
Approximation of $\Delta r_{n+1} dt$

evaluated change in the time integral of the restoring force, the same term as the one which was earlier estimated in equation (5). In general, due to the material non-linearity, the restoring force will not follow the linear extrapolation as estimated and the change in the time integral of this restoring force will in reality be smaller than estimated. For this reason, Chang's algorithm then recalculates Δv_{n+1}, based on the measured time integral of the restoring force. At this stage, Δd_{n+1} may or may not be recalculated, based on the updated Δv_{n+1} in equation (4c). Irrespective of whether the displacement increment Δd_{n+1} is then recalculated, there is a numerical damping present, which may be negative or positive depending on which reference values for the restoring force are used at the beginning of the displacement increment. The effect will be present irrespective of whether the tangential or initial stiffness term is used in equation (5).

Newmark implicit – integral form

There is an inconsistency in the integral form of the Newmark explicit algorithm, when recalculating Δv_{n+1} is based on the updated $\Delta r_{n+1} dt$ (Algaard *et al.*, 2000). Δv_{n+1} indeed has to be recalculated; otherwise the information about the experimentally measured restoring forces is never taken into account. The predictor Δd_{n+1} is based on an estimation of Δv_{n+1}, which in turn builds on an approximation of the restoring force as expressed in equation (5). Once the integral of the restoring force has been obtained, the integrated equation of motion, equation (4a), can be applied.

Expanding equation (4c) yields:

$$\begin{aligned} d_{n+1} &= d_n + \frac{1}{2}\Delta t(v_n + v_{n+1}) = d_n + \frac{1}{2}\Delta t(v_n + v_n + \Delta v_{n+1}) \\ &= d_n + \Delta t v_n + \frac{1}{2}\Delta t \Delta v_{n+1} \end{aligned} \tag{8}$$

Furthermore, equation (4a) assumes the integral of the restoring force over that time step, $\Delta \int r_{n+1} dt$, to be determined by computing the time integral of the restoring force over the time step, as indicated in Figure 3. Assuming linear stiffness for simplicity, the exact expression for $\Delta \int r_{n+1} dt$ will be as follows:

$$\Delta \int r_{n+1} dt = \int r_{n+1} dt - \int r_n dt = \Delta t k \left(\frac{d_n + d_{n+1}}{2} \right) \tag{9}$$

Equation (8) can now be substituted for d_{n+1} in equation (9) to yield the following:

$$\Delta \int r_{n+1} dt = \Delta t k \left(\frac{d_n + d_n + \Delta t v_n + 1/2 \Delta t \Delta v_{n+1}}{2} \right) \tag{10}$$

which can be further manipulated into:

$$\Delta \int r_{n+1} dt = \Delta t k d_n + \frac{1}{2}\Delta t^2 k v_n + \frac{1}{4}\Delta t^2 k \Delta v_{n+1} \tag{11}$$

By comparing equation (11) with equation (5) it is clear that equation (11) now contains one additional term, $1/4\Delta t^2 k \Delta v_{n+1}$. This term corresponds to the term which is omitted in the standard implicit Newmark algorithm to render it explicit (i.e. $\beta = 0$), and the omission of this term is the cause of the numerical damping invariably present in the integral form of the algorithm. However, omitting the equivalent term in the integral form of the method does not render the method explicit – in fact its omission has no bearing on the nature of the algorithm.

The integral form of the Newmark algorithm has been made explicit through an estimation of the time integral of the restoring force, which enables the calculation of Δv_{n+1} followed by Δd_{n+1}. At this point it becomes clear that there is no reason why the seemingly implicit additional term in equation (11) cannot be included in the estimation of the time integral of restoring force, as the implicit variable is the actual unknown the expression is trying to represent. The situation is clarified through the following argument, where equation (11) has been substituted into equation (4a) and solved for Δv_{n+1} to yield an alternative expression for equation (6).

$$\Delta v_{n+1} = \left(m + \frac{\Delta t}{2}c\right)^{-1} \left(\Delta \int f_{n+1} dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n - \frac{1}{4}\Delta t^2 k_0 \Delta v_{n+1}\right) \tag{12}$$

The unknown, Δv_{n+1}, is present on both sides of the equation, but through further rearrangement:

$$\Delta v_{n+1} + \left(m + \frac{\Delta t}{2}c\right)^{-1} \frac{1}{4}\Delta t^2 k_0 \Delta v_{n+1}$$

$$= \left(m + \frac{\Delta t}{2}c\right)^{-1} \left(\Delta \int f_{n+1} dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)$$

$$\left(1 + \left(m + \frac{\Delta t}{2}c\right)^{-1} \frac{1}{4}\Delta t^2 k_0\right) \Delta v_{n+1}$$

$$= \left(m + \frac{\Delta t}{2}c\right)^{-1} \left(\Delta \int f_{n+1} dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)$$

an explicit expression for the velocity can finally be found.

$$\Delta v_{n+1} = \frac{\left(m + \frac{\Delta t}{2}c\right)^{-1}\left(\Delta \int f_{n+1}dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)}{1 + \left(m + \frac{\Delta t}{2}c\right)^{-1}\frac{1}{4}\Delta t^2 k_0} \quad (13)$$

Newmark time stepping scheme

This equation may now be substituted directly into equation (8) and rearranged to obtain a new explicit expression for the displacement predictor Δd_{n+1}.

$$\Delta d_{n+1} = \Delta t v_n + \frac{\frac{\Delta t}{2}\left(m + \frac{\Delta t}{2}c\right)^{-1}\left(\Delta \int f_{n+1}dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)}{1 + \left(m + \frac{\Delta t}{2}c\right)^{-1}\frac{1}{4}\Delta t^2 k_0} \quad (14)$$

By using equation (14) rather than equation (7) as the displacement predictor, the time stepping algorithm has become implicit. To be exact, the algorithm still requires a value of the tangent stiffness (k_0 terms in equation (14)), however this is also the case with the earlier explicit version of the algorithm and other implicit algorithms (Combescure and Pegon, 1997). The same term is now merely present in one additional place, and as with the explicit version, the initial stiffness, or an estimate of the tangent stiffness, can be used.

Principal differences between the two algorithms are summarised in Table I.

Stability and dissipation properties

The stability properties of an integration algorithm are studied by considering spectral properties of its recursive amplification matrix. Considering the state vector of the system at time $t = t_n$, the integral form of the numerical time integrator algorithms yields:

$$X_n = \begin{bmatrix} \int d_n dt \\ \Delta t d_n \\ \Delta t^2 v_n \end{bmatrix} \quad (15)$$

Chang's integral form Newmark explicit algorithm		
Predictor d_{n+1}	$\Delta t v_n + \frac{\Delta t}{2}\left(m + \frac{\Delta t}{2}c\right)^{-1}\left(\Delta \int f_{n+1}dt - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)$	
$\int d_{n+1}dt$	$\int d_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$	
Proposed integral form Newmark implicit algorithm		
Predictor d_{n+1}	$\Delta t v_n + \frac{\frac{\Delta t}{2}\left(m + \frac{\Delta t}{2}c\right)^{-1}\left(\Delta \int f_{n+1} - c\Delta t v_n - \Delta t r_n - \frac{k_0}{2}(\Delta t)^2 v_n\right)}{1 + \left(m + \frac{\Delta t}{2}c\right)^{-1}\frac{1}{4}\Delta t^2 k_0}$	
$\int d_{n+1}dt$	$\int d_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n + \frac{1}{4}(\Delta t)^2 \Delta v_{n+1}$	

Table I. Principal differences between the algorithms

For stability purposes, one can ignore the external load vector and damping forces, thus the recurrent relationship between the state vector at $t = t_n$ and $t = t_{n+1}$ can be expressed as (Bathe and Wilson, 1976):

$$X_{n+1} = [A]X_n \tag{16}$$

where $[A]$ is the recursive amplification matrix.

Stability of the Newmark explicit – integral form algorithm

Considering first the explicit, integral form algorithm, equations (4) can be expressed in terms of the variables of the state vector as:

$$\int d_{n+1}dt = \int d_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \frac{1}{2}\Delta t(v_n + v_{n+1}) \tag{17}$$

$$\Delta v_{n+1} = M^{-1}\left[-\int_n^{n+1} r_{n+1}dt\right]$$

Depending on the precise implementation of the algorithm, i.e. whether d_{n+1} is updated following the recalculation of v_{n+1} or not, the exact expressions for d_{n+1} and $r_{n+1}dt$ will differ.

Assuming initially that d_{n+1} is not recalculated, the predicted d_{n+1} remains and the term $r_{n+1}dt$ will be a function of the restoring force both at the start and at the end of the predicted step, yielding the second of equations (18). d_{n+1} will thus no longer be represented by the implicit expression in equation (17), rather by a simplification of equation (7) containing only the terms relevant for stability analyses. $r_{n+1}dt$ may be defined in terms of d_{n+1} or by the expression for the prediction step, however this will in the end lead to the same amplification matrix. Assuming linear stiffness for simplicity, $r_{n+1}dt$ may be expressed as $\frac{\Delta tk}{2}(d_n + d_{n+1})$, yielding the third equation of equations (18).

$$\int r_{n+1}dt = \int r_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$d_{n+1} = d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right) \tag{18}$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta tk}{2}(d_n + d_{n+1})\right)$$

Multiplying the second and third equation by Δt and Δt^2, respectively, and substituting Ω^2 for $\Delta t^2 k/m$, yields equations (19).

$$\int r_{n+1}dt = \int r_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n - \frac{1}{2}\Omega^2 \Delta t d_n - \frac{1}{4}\Omega^2 (\Delta t)^2 v_n \tag{19}$$

$$\Delta t^2 v_{n+1} = \Delta t^2 v_n - \Omega^2 \frac{\Delta t d_n}{2} - \Omega^2 \frac{\Delta t d_{n+1}}{2}$$

After sorting terms at $t = t_{n+1}$ and $t = t_n$ and expressing them in matrix form (Géradin and Rixen, 1994), the amplification matrix is obtained as:

$$A = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1 - \frac{\Omega^2}{2} & 1 - \frac{\Omega^2}{4} \\ 0 & -\Omega^2 + \frac{\Omega^4}{4} & 1 - \frac{\Omega^2}{2} + \frac{\Omega^4}{8} \end{bmatrix} \tag{20}$$

which clearly differs from the normal Newmark explicit matrix (Shing and Mahin, 1987). Stability of an algorithm is ensured when the spectral radius of the amplification matrix does not exceed unity (Golley and Amer, 1999). In the above matrix, one eigenvalue will be equal to unity, while the other two will form a pair of complex conjugates. Corresponding moduli have been plotted below as a function of Ω in Figure 4. Both the expected stability limit of 2.0 and the existence of noticeable numerical damping can be seen for the values of Ω exceeding 0.5.

If however the d_{n+1} is recalculated once the corrected v_{n+1} has been found, the situation is somewhat different. Equations (18) will be altered such that the displacement predictor is no longer present in the definition of d_{n+1}, but remains in the expression for v_{n+1}. This yields equations (21), which can be represented by the amplification matrix shown in equation (22), obtained the same way as equation (20).

$$\int r_{n+1}dt = \int r_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n$$

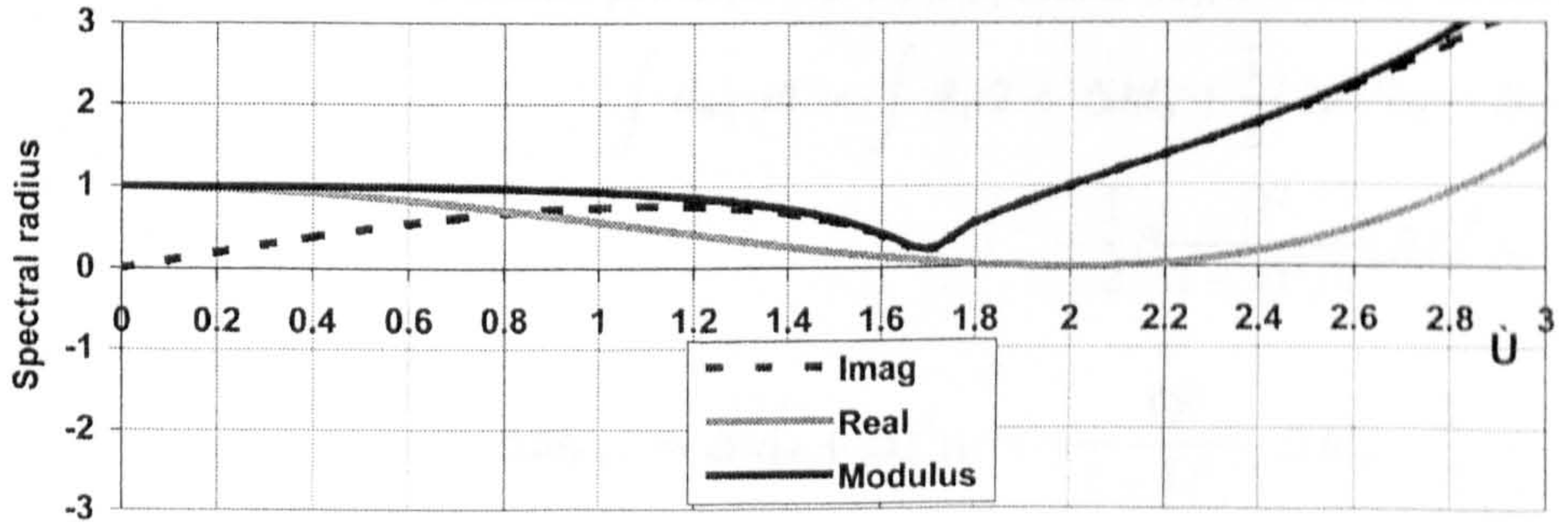


Figure 4.
Spectral radius vs. of Newmark explicit – integral form

$$d_{n+1} = d_n + \frac{1}{2}\Delta t v_n + \frac{1}{2}\Delta t v_{n+1}$$

$$v_{n+1} = v_n + M^{-1}\left(-\frac{\Delta t k}{2}\left(2d_n + \Delta t v_n + \frac{1}{2}\Delta t M^{-1}\left(-\Delta t k d_n - \frac{k}{2}(\Delta t)^2 v_n\right)\right)\right) \tag{21}$$

$$A = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1 - \frac{\Omega^2}{2} + \frac{\Omega^4}{8} & 1 - \frac{\Omega^2}{4} + \frac{\Omega^4}{16} \\ 0 & -\Omega^2 + \frac{\Omega^4}{4} & 1 - \frac{\Omega^2}{2} + \frac{\Omega^4}{8} \end{bmatrix} \tag{22}$$

This algorithm exhibits similar stability and damping characteristics as the standard Newmark explicit; perfect energy conservation up to the stability limit of 2.0. However, as the algorithm stands, it cannot be directly implemented into a pseudodynamic test. This is because r_n is in fact unknown at the start of the time step. The reason for this is that d_n was recalculated after the completion of the previous step, and the restoring force caused by it is thus unknown. The correct procedure would require the recalculated d_{n+1} to be imposed separately and the corresponding restoring force re-measured. Such a procedure would lead to a double step implementation, but employing iterations in an algorithm that is still only conditionally stable seems *inappropriate*. The method is instead implemented using the restoring force as measured at the end of the predictor step, before any update. Using this alternative method results in numerical damping and a reduced stability limit, but the appropriate amplification matrix for this algorithm cannot be formed.

Stability of the Newmark implicit – integral form algorithm

The effects of using the implicit version of the algorithm with the modified displacement predictor are substantial. Not only does the method avoid the numerical damping associated with the Newmark explicit – integral form algorithm, but owing to the fact that the algorithm is now genuinely implicit, it also becomes unconditionally stable. This was initially noted through numerical experiments, but can also be confirmed analytically. By considering the expression for the time integral of displacement, the displacement and the velocity and using a similar procedure as with equation (19), it leads to the following equations:

$$\int d_{n+1}dt = \int d_n dt + \Delta t d_n + \frac{1}{2}(\Delta t)^2 v_n - \beta\frac{\Omega^2}{1+\Omega^2/4}\Delta t d_n - \frac{1}{2}\beta\frac{\Omega^2}{1+\Omega^2/4}(\Delta t)^2 v_n$$

$$\Delta t d_{n+1} = \Delta t d_n + \Delta t^2 v_n - \gamma\frac{\Omega^2}{1+\Omega^2/4}\Delta t d_n - \frac{1}{2}\gamma\frac{\Omega^2}{1+\Omega^2/4}(\Delta t)^2 v_n \tag{23}$$

$$\Omega^2 \int d_{n+1}dt + (\Delta t)^2 v_{n+1} = \Omega^2 \int d_n dt + (\Delta t)^2 v_n$$

where β and γ are the parameters normally present in the Newmark algorithms, which typically take on the values of 0.25 and 0.5, respectively. Expressing the above equations in a matrix form yields again the recursive amplification matrix of the integration operator (Géradin and Rixen, 1994):

$$A = \begin{bmatrix} 1 & 1-\beta\frac{\Omega^2}{1+\Omega^2/4} & \frac{1}{2}-\frac{1}{2}\beta\frac{\Omega^2}{1+\Omega^2/4} \\ 0 & 1-\gamma\frac{\Omega^2}{1+\Omega^2/4} & 1-\frac{1}{2}\gamma\frac{\Omega^2}{1+\Omega^2/4} \\ 0 & -\Omega^2\left(1-\beta\frac{\Omega^2}{1+\Omega^2/4}\right) & 1-\Omega^2\left(\frac{1}{2}-\frac{1}{2}\beta\frac{\Omega^2}{1+\Omega^2/4}\right) \end{bmatrix} \quad (24)$$

The complex expression for the eigenvalues of [A] can be simplified to

$$(1-\lambda)\left(\lambda^2 + \left\{\left(-2+\frac{1}{2}\Omega^2\right)\left(1-\frac{1}{4}\xi^2\right)\right\}\lambda + 1 + \frac{\Omega^2}{2} - \frac{\Omega^2\xi^2}{8} - \frac{1}{2}\xi^2\right) = 0 \quad (25)$$

where λ are the eigenvalues and ξ^2 is expressed as

$$\xi^2 = \frac{\Omega^2}{1+\frac{1}{4}\Omega^2} \quad (26)$$

Eliminating $\lambda_1 = 1$ leaves the remaining second order equation:

$$\lambda_{2,3} = \frac{\left(-2+\frac{\xi^2}{2}+\frac{\Omega^2}{2}-\frac{\Omega^2\xi^2}{8}\right) \pm \sqrt{-4\Omega^2+\frac{1}{4}\xi^4+\frac{1}{4}\Omega^4+\frac{3\xi^2\Omega^2}{2}-\frac{\Omega^4\xi^2}{8}-\frac{\Omega^2\xi^4}{8}+\frac{\Omega^4\xi^4}{64}}}{2}$$

By plotting the real and imaginary parts of the solution and computing the moduli, it can be seen that the moduli for $\lambda_{2,3}$ are also equal unity for all Ω, Figure 5. This clearly yields the overall solution that $\rho(A) = 1$ for all Δt, which implies unconditional stability and perfect energy conservation.

The modifications carried out on the Newmark explicit – integral form have shown to eliminate the amplitude error of the algorithm and also improve the

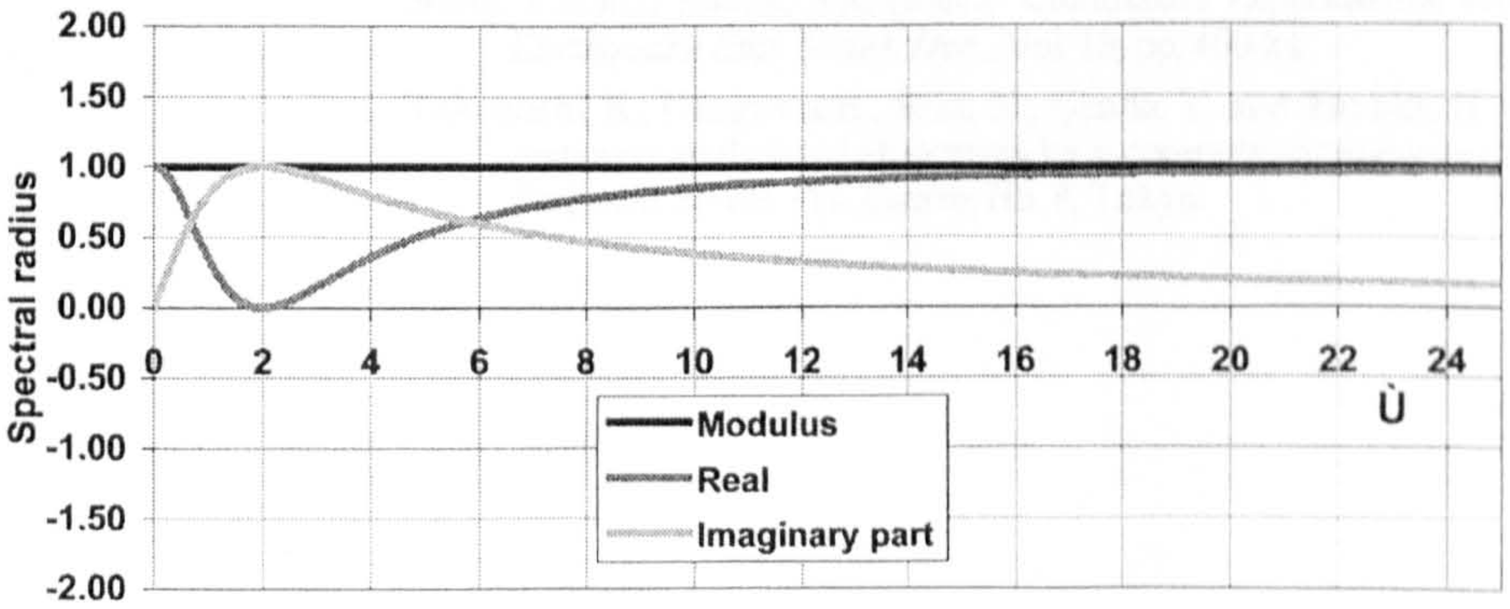


Figure 5.
Spectral radius vs. Newmark implicit – integral form

stability properties such that it is now unconditionally stable. As the algorithm is now implicit, the name Newmark implicit – integral form seems appropriate.

Conclusions

By including an additional term in the estimate for the restoring force in the Newmark explicit – integral form time-integration algorithm, a new displacement predictor has been proposed, which is consistent with the constant average velocity the algorithm assumes. This addition eliminates numerical damping which is otherwise present. Furthermore, the modification renders the algorithm unconditionally stable. It now successfully combines advantages of the integral form in handling rapidly varying loads and stiffness degradation with the unlimited time step sizes associated with implicit methods.

References

Algaard, W., W. Agar, W. and Bicanic, N. (2000), "Computational aspects of pseudodynamic testing", *Proc. 8th Conference of the Association for Computational Mechanics in Engineering (ACME 2000)*, Greenwich, April, pp. 12-15.

Bathe, K.J. and Wilson, E.D. (1976), *Numerical Methods in Finite Element Analysis*, Prentice-Hall, Englewood Cliffs, NJ.

Bursi, O.S. and Shing, P.B. (1996), "Evaluation of some implicit time-stepping algorithms for pseudodynamic tests", *Earthquake Eng. Struct. Dyn.*, Vol. 25, pp. 333-55.

Combescure, D. and Pegon, P. (1997), "Operator splitting time integration technique for pseudodynamic testing", *Soil Dynamics and Earthquake Engineering*, Vol. 16, pp. 427-43.

Chang, S.Y. (1997), "Improved numerical dissipation for explicit methods in pseudodynamic tests", *Earthquake Eng. Struct. Dyn.*, Vol. 26, pp. 917-29.

Chang, S.Y., Tsai, K.-C. and Chen, K.-C. (1998), "Improved time integration for pseudodynamic tests", *Earthquake Eng. Struct. Dyn.*, Vol. 27, pp. 711-30.

Géradin, M. and Rixen, D. (1994), *Mechanical Vibrations, Theory and Application to Structural Dynamics*, Masson, Paris.

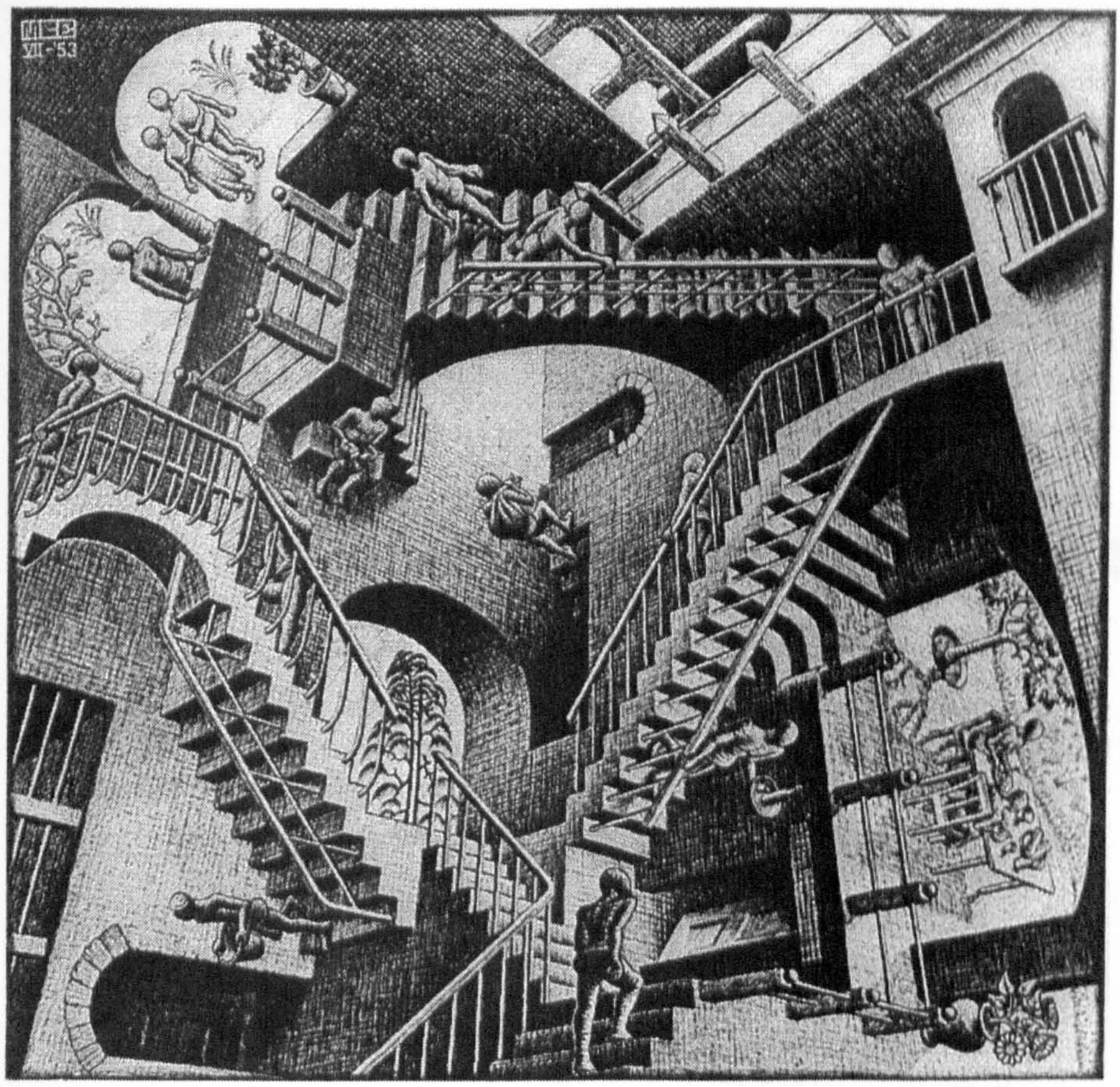
Golley, B.W. and Amer, M. (1999), "An unconditionally stable time-stepping procedure with algorithmic damping: a weighted integral approach using two general weight functions", *Earthquake Eng. Struct. Dyn.*, Vol. 28, pp. 1345-60.

Shing, P.B. and Manivannan, T. (1990), "On the accuracy of an implicit algorithm for pseudodynamic tests", *Earthquake Eng. Struct. Dyn.*, Vol. 19, pp. 631-51.

Shing, P.S. and Mahin, S.A. (1987), "Cumulative experimental errors in pseudodynamic tests", *Earthquake Eng. Struct. Dyn.*, Vol. 15, pp. 409-24.

Takanashi, K., Udagawa, K., Seki, M., Okada, T. and Tanaka, H. (1975), "Nonlinear earthquake response analysis of structures by a computer-actuator on-line system", *Bull. Earthquake Resistant Struct. Res. Centre*, No. 8, Tokyo.

Appendix



Any discussion on time stepping schemes and recursive time integration operators is for the third author a reminder of happy times and endless discussions with Ernie Hinton on the stability and dispersion characteristics of time integrators and above all about the common admiration and bewilderment with recursive drawings by the Dutch graphicus Maurits Cornelis Escher. This drawing entitled "Relativiteit" (litho 1953) is reproduced from the book "De tovoerspiegel van M. C. Escher" by Bruno Ernst, Meulenhoff, Amsterdam 1976.

Figure A1.
"Relativiteit" by Maurits Cornelis Escher

Appendix H:

ACME 2001 CONFERENCE PAPER

Algorithmic Improvements of Pseudodynamic testing

W. Algaard, N. Bićanić & A. Agar
Department of Civil Engineering, University of Glasgow, Glasgow, G12 8LT, U.K.

Abstract

Software based implementation system for pseudodynamic testing, incorporating the novel Newmark Implicit – Integral form time stepping algorithm, is realised within the LabView environment. The system is discussed with an assessment of algorithmic and control sensitivities of the SDOF set-up. On the algorithmic level these include the effect of the type of time integration scheme and the time step size, while the control effects are concerned with the method and the rate of loading, controller capabilities and instrumentation accuracy. The sensitivities are illustrated with two model problems, an inelastic reinforced concrete column and an elastic steel column.

Introduction

Pseudodynamic testing is a combined computational/experimental technique for evaluating dynamic systems. The method relies on modelling inertial and damping forces computationally, while the non-linear restoring forces are measured experimentally. Dynamic equilibrium equations for mass-spring-damper systems subjected to applied loads can generally be expressed as

$$M\frac{d^2x}{dt^2}+C\frac{dx}{dt}+r(x)=f \tag{1}$$

where M and C are mass and viscous damping matrices and x, r and f are the displacement, restoring force and applied force vectors, respectively. The pseudodynamic test method uniquely utilises both computational and experimental terms to form the equation of motion (1). A time-stepping algorithm computes a displacement step, which is subsequently imposed on the structure by means of computer controlled servo-hydraulic actuators. Once the structure has been deformed, the resulting restoring forces are measured. This can be done either during a hold period where the actuator remains stationary, or continually enabling smooth deformation of the structure. Based on the restoring force and the current damping and applied forces, the resulting acceleration may be computed, and the new displacement step can then be calculated. Sensitivities in pseudodynamic implementation are concerned with the speed and accuracy obtained experimentally as well as limitations on the algorithmic level.

Computational set-up

An unconventional approach has been elected to control, implement and execute the pseudodynamic tests. While hardware controllers and several computers are typically employed to handle execution and data logging, the system at Glasgow comprises a single PC with a high-speed communication card. Not only does the same computer conduct the entire running of the test, but all the computational components are included in the same environment. These range from the time integration algorithm and equilibrium calculations, through the implementation and execution system with data logging to the actual actuator controller unit. In fact, the whole computational side of the set-up is created as a single piece of code in the National Instruments' LabView 5

environment. This entails a fully integrated implementation system entirely on a software level.

Execution structure

Initially, the system properties of the tested dynamic structure have to be defined. These include the mass, viscous damping and, under some circumstances, stiffness properties, followed by selection of the desired time stepping algorithm and the time step size. Finally, the external loading on the structure is selected in terms of a ground motion accelerogram with a scale factor. Preliminary calculations then take place before all required data is passed into the selected implementation algorithm.

The implementation loop, also referred to as the outer loop, largely consists of a time stepping algorithm and a control algorithm. The time stepping algorithm essentially computes a change in displacement, velocity and acceleration based on the equilibrium equation with the applied force and measured restoring forces acting. This information is then passed forward into the next time step. However, prior to this, the computed displacement step, forming a target displacement, is sent into the control algorithm situated within the outer loop. The control algorithm, appearing as an inner loop, then communicates with the hardware and generates signals sending the actuator towards the target. The inner loop will then iterate until the displacement has been achieved. At this point, a measure of the restoring force is taken, and all variables are logged to the local hard drive before a new time step is commenced. This process is then repeated for the specified test duration or until it is stopped.

Implementation

A semi-continuous implementation system has been adopted implying the actuator motion is not interrupted by prescribed hold-periods. The two major parts of the implementation loop, time integration calculations and control iterations, are however not carried out concurrently. Whenever the control algorithm concludes, active control of the actuator is lost until a new target displacement is calculated in the next displacement step. The method is therefore not fully continuous. All required calculations are carried out in an amount of time comparable to that required for each iteration in the control algorithm. The implementation method thus relies on switching between time integration and control without delay, which is achieved by integrating the two algorithms fully by coding them into the same program [1].

Control algorithm

The purpose written control algorithm has as primary function to generate signals for the servo valve, ensuring that the actuator moves towards the target displacement commanded by the time integration algorithm. As the actuator is under displacement control, the required active channels for a SDOF pseudodynamic test controlled locally consist of two input channels, displacement and force signals, and one output channel carrying the valve signal.

The controller employs both proportional and integral type control modes. These have been especially adapted for the hydraulic system available. Adjusting the setting of these also determines the actuator speed. While the proportional part of the signal is generated as a multiple of the difference between the target and measured position, the integral part consists of a linear signal related to the integral of the displacement error. The recently proposed Integral Form time integration algorithms [1, 3, 4] require the

time integral of the restoring force to be calculated over each time step. This is in order to take account of any intra-step variations in the stiffness of the structure, which can be significant during damage accumulation in sizeable time steps. To obtain the maximum possible accuracy, force readings should be taken during each control iteration, and to minimize delays between each time step, the time integral of such should be computed as the process continues. To achieve this, the measurements and computations are coded directly into the controller. The situations is further complicated by the fact that the actuator velocity profile generally differs significantly from the constant one assumed by time integration algorithms. To take account of this, equivalent time points on the assumed displacement curve have to be found and used for the purpose of computing the time integral [1].

Sensitivity study

Considerable work has been carried out to study the error propagation effects in pseudodynamic tests, e.g. Combescure *et al.* [2], but little or no research has been carried out to investigate implementation sensitivities in general. These range from those concerned with time integration (type of, and details within the time stepping algorithm, as well as time step size) to those concerned with the experimental implementation (e.g. rate and method of loading, continuous or step-wise, controller capabilities, accuracy obtained and details within measurements).

The relationships between: *time step size*, *speed of implementation*, *discrete step accuracy* and *final system response* are considered. Two different time stepping algorithms are employed to evaluate how the above relationships may be affected by algorithmic differences in both time integration and control.

Test programme

In order to evaluate the relationships discussed above, repeated pseudodynamic tests on two different dynamic systems have been carried out. These comprise a *reinforced concrete stub column* and a *slender steel column*, both with lumped virtual masses on top, yielding an inverted pendulum system. The masses have in both cases been tuned to create structures with natural frequencies with fundamental periods of around 1.2 seconds. Both structures were exposed to a scaled Port Hueneme accelerogram, and the first 4 seconds of the response were modelled using the central difference and the novel Newmark Implicit – Integral Form methods [1, 3, 4]. Zero viscous damping is applied throughout. Typical force-displacement curves for the two columns subjected to a horizontal load at the top can be found in figures 1a and 1b.

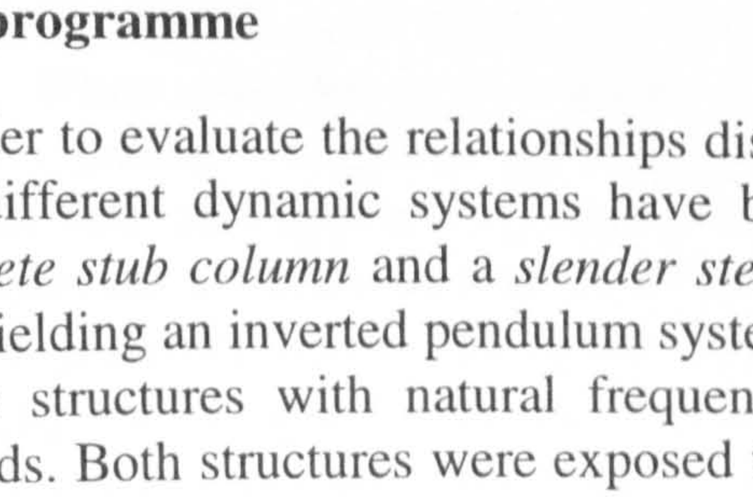


Figure 1a: Force-displacement loops for steel column.

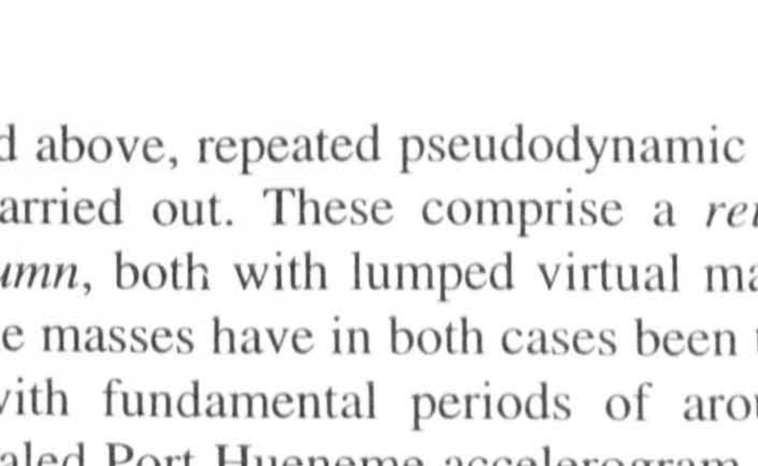


Figure 1b: Force-displacement loops for concrete column

Results: Response – Time integration relationship

Displaying the system response created using different time step sizes and time integration algorithms should reveal any effects on the response caused the variations in the time stepping schemes. System responses of the reinforced concrete column and the elastic steel column to the artificial accelerogram using step sizes of 0.04, 0.08 and 0.16s with the Newmark Implicit – Integral Form and central difference algorithms are displayed below.

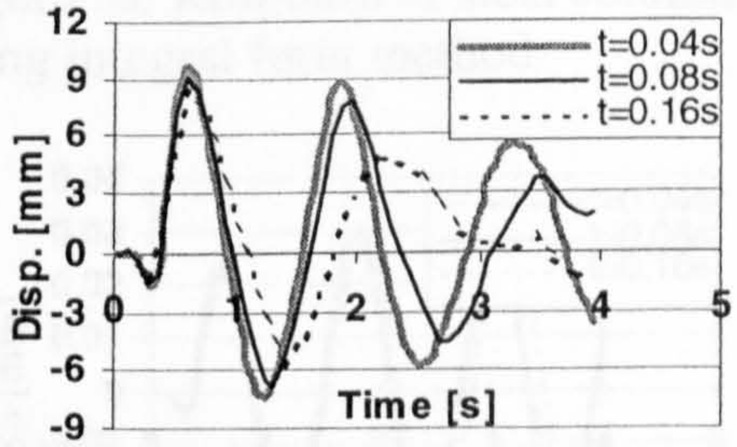


Figure 2a: Response obtained using integral form method.

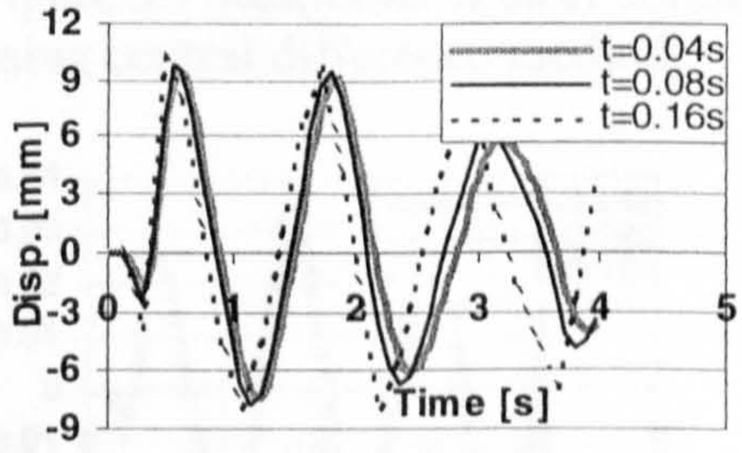


Figure 2b: Response obtained using central difference method.

Considering first the tests on the reinforced concrete specimen, figures 2a and 2b, it immediately becomes apparent that large differences in the response result both from the choice of the time step size and the integration algorithm. Assuming the shortest time step generates the most accurate response, increasing the step size with the central difference method results in a period shortening, while for the integral form it results in period elongation. Additionally, the central difference method displays evidence of amplitude amplification, while the integral form method displays evidence of amplitude decay. When considering the results from steel column, the integral form method still displays period elongation and the central difference method period shortening. The integral form also appears to generate more damping, figures 3a and 3b.

Numerical simulations may be carried out on the steel column system as the stiffness is near perfectly linear. Using the measured stiffness, including the same mass and exposing the sample to the same ground motion as in the pseudodynamic tests, the response was obtained numerically using the same time integration methods. The response obtained can be seen in figures 4a and 4b using the integral form and central difference methods respectively.

Analysis: Response – Time integration relationship

The tests on both the concrete and steel columns revealed period elongation when using the integral form algorithm and shortening when using the central difference method. Evidence of more damping with the integral form method existed, but through further numerical tests covering longer test time, it was confirmed that both methods were energy stable for linear systems, as expected from analytical analyses. The numerical tests also provided reference solutions to the pseudodynamic tests on the steel column. The response from this is near identical with the numerically generated one, for all time step sizes, comparing figures 3a/4a and 3b/4b. The small differences that exist are due to exclusion of viscous damping in the numerical model and a small offset of the zero in the experimental tests, hence the differences caused by time integration exist entirely on an algorithmic level.

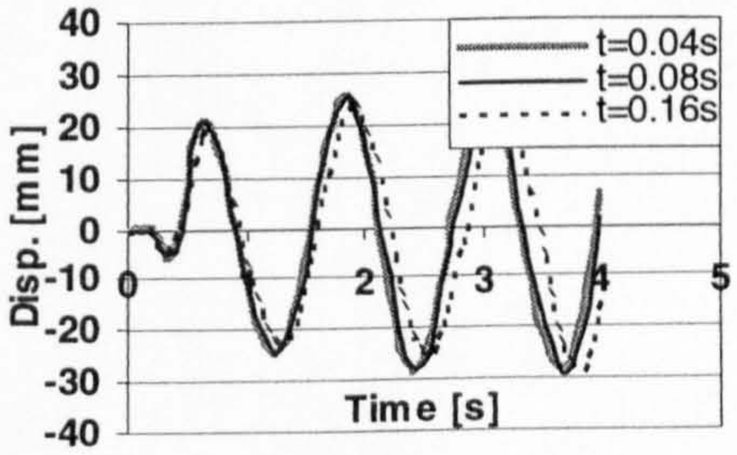


Figure 3a: Response of steel column using integral form method.

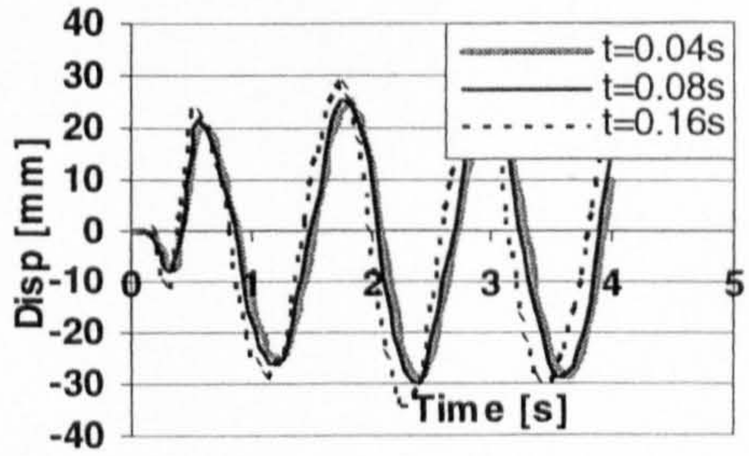


Figure 3b: Response of steel column using central difference method.

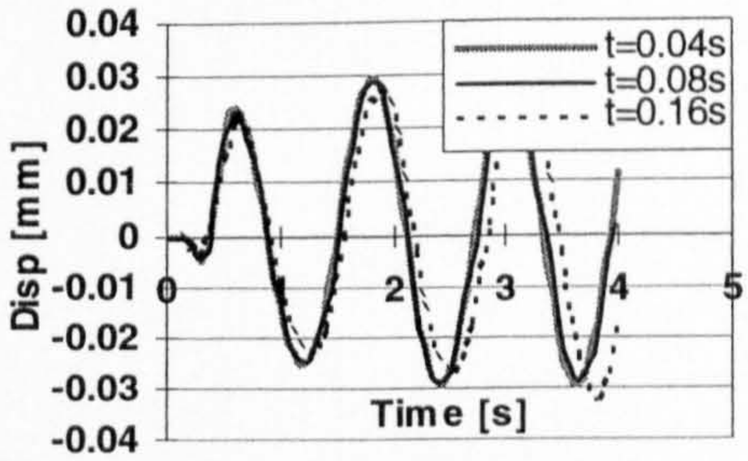


Figure 4a: Numerical simulation using integral form method.

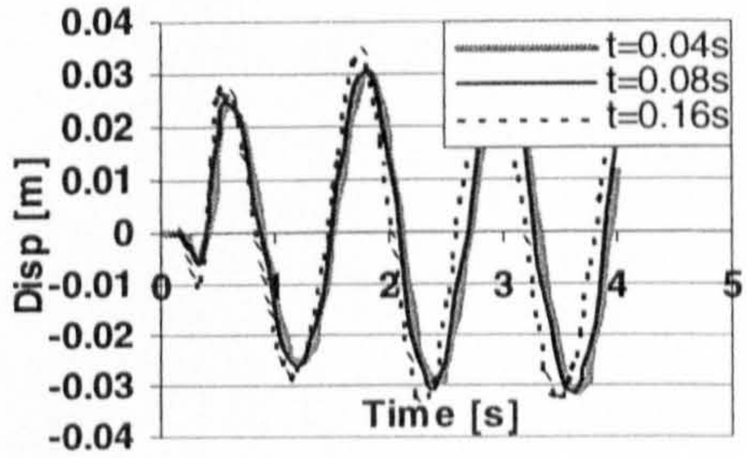


Figure 4b: Numerical simulation using central difference method.

Conclusions

An implementation system for pseudodynamic testing has been developed based on a software controller system included within the test execution program running on a singe desktop PC in the LabView environment. The system has been used to perform tests on two different dynamic systems: a highly non-linear reinforced concrete column and a linear steel column. The tests were conducted to evaluate the sensitivity of the results with respect to the implementation. It has been concluded that the time integration scheme, and time step size affect the response, however that this is exclusively due to algorithmic effects at least in the linear case.

References

1: W. Algaard, 'Algorithmic and control implementation in pseudodynamic testing' *Ph.D. Thesis*, Department of Civil Engineering, University of Glasgow, Scotland, to be submitted 10.2001

2: D. Combescure & P. Pegon, 'α-Operator Splitting time integration technique for pseudodynamic testing. Error propagation analysis', *Soil Dynamics and Earthquake Engineering*, **16**, 427-443, (1997)

3: W. Algaard, A. Agar and N. Bicanic, 'Enhanced Integral Form of the Newmark Time Stepping Scheme for Pseudodynamic Testing', *Int. Journal of Engineering Computations* (subm.)

4: S-Y. Chang, K-C. Tsai and K-C. Chen, 'Improved Time Integration for Pseudodynamic Tests', *Earthquake Engng Struct Dyn.*, **27**, 711-730, (1998)